
Field Service Developer Guide

Version 66.0, Spring '26



CONTENTS

Chapter 1: Field Service Developer Guide	1
Get Ready to Develop with Field Service	2
API End-of-Life Policy	2
Field Service Data Objects	2
Field Service Core Data Model	3
Field Service Inventory Management Data Model	7
Field Service Preventive Maintenance Data Model	9
Field Service Product Service Campaign Data Model	11
Field Service Warranty Management Data Model	11
Field Service Pricing Data Model	12
Field Service Object References	13
Field Service REST API	498
Field Service Flow	498
Field Service Mobile Settings	503
Service Report Template	506
Appointment Bundling REST APIs	512
Field Service Metadata API	544
FieldServiceSettings	544
Skill	552
TimeSheetTemplate	554
Field Service Tooling API	556
CleanRule	556
TimeSheetTemplate	561
Field Service Apex Namespace	565
FSL Apex Namespace	565
Field Service Custom Triggers	621
Field Service Code Examples	634
Create a Service Report with Apex	635
Generate Work Orders on Maintenance Plans with Apex	637
Code Examples: Dispatcher Console Custom Actions	638
Create Service Appointment Lists in the Dispatcher Console	641
Field Service Mobile App	642
Get Started with Lightning Web Components in the Field Service Mobile App	643
Build Custom Lightning Web Components for Service Documents	657
Configure Deep Linking for the Field Service Mobile App	664
Add Lightning Web Components for Plug-Ins to the Field Service Mobile App	671
Index	682

CHAPTER 1 Field Service Developer Guide

In this chapter ...

- [Get Ready to Develop with Field Service](#)
- [API End-of-Life Policy](#)
- [Field Service Data Objects](#)
- [Field Service REST API](#)
- [Field Service Metadata API](#)
- [Field Service Tooling API](#)
- [Field Service Apex Namespace](#)
- [Field Service Custom Triggers](#)
- [Field Service Code Examples](#)
- [Field Service Mobile App](#)

Field Service is a powerful, highly customizable toolbox of features that you can use to set up and manage a multiplatform, mobile service operation. This guide offers reference information and code examples to help you understand the structure of Field Service features and customize them with code.

This guide includes:

- Object diagrams that illustrate the relationships between field service data objects
- Reference information for field service objects in the REST, Metadata, and Tooling APIs
- Apex reference information for the Field Service managed package
- Apex code examples that you can use to complete common field service tasks

If you're new to field service, we recommend checking out [Get on the Road with Field Service](#) in Trailhead. Then, find curated links to documentation, blogs, Trailhead modules, and other resources through the [Field Service learning map \(English only\)](#)

Happy coding!

SEE ALSO:

[Install the Field Service Managed Package](#)
[Field Service Mobile App](#)

Get Ready to Develop with Field Service

To prepare to work programmatically with field service features, verify that Field Service is enabled in your org.

1. From Setup, enter *Field Service Settings* in the Quick Find box, then select **Field Service Settings**.
2. Verify that Field Service is enabled.
3. Click **Save**.

You now have access to Salesforce's standard field service objects, which are listed in this guide. However, that's just the beginning! To get your feet wet before you start working programmatically with Field Service, it's a good idea to walk through the setup tasks outlined in [Set Up Field Service](#).

API End-of-Life Policy

Salesforce is committed to supporting each API version for a minimum of 3 years from the date of first release. To improve the quality and performance of the API, versions that are over 3 years old sometimes are no longer supported.

Salesforce notifies customers who use an API version scheduled for deprecation at least 1 year before support for the version ends.

-  **Note:** Version 20.0 of REST API and SOAP API have now been deprecated and are no longer supported. You can continue to access this legacy API version until Summer '22 is released, at which point this legacy version will be retired and will become unavailable. For more information, see this Knowledge Article: [Salesforce Platform API Versions 7.0 through 20.0 Retirement](#).
-  **Note:** Versions 21.0 through 30.0 of REST API and SOAP API will be deprecated in the Summer '22 release. For more information, see this Knowledge Article: [Salesforce Platform API Versions 21.0 through 30.0 Retirement](#).

Field Service Data Objects

Field Service is based on a suite of standard and custom Salesforce objects that relate to each other in different ways. These objects also serve as the foundation of the Field Service managed package and mobile app.

The following object relationship diagrams and reference information are your field service object dictionary. Use them to guide your implementation decisions. You can create, retrieve, update, or delete field service records and objects, through our SOAP and REST APIs. You can query these objects using SOQL (Salesforce Object Query Language).

IN THIS SECTION:

[Field Service Core Data Model](#)

Use Field Service's core objects to complete essential field service tasks such as managing work orders, defining your service territories, and tracking your workforce.

[Field Service Inventory Management Data Model](#)

Use Field Service's inventory objects to track the storage, request, consumption, return, and retirement of items in your inventory.

[Field Service Preventive Maintenance Data Model](#)

Manage periodic maintenance with the help of maintenance plans, which define the maintenance schedule for specific assets. Maintenance plans typically reflect the terms in a customer's service contract or entitlements.

[Field Service Product Service Campaign Data Model](#)

Use product service campaign to record the actions to address situations such as product recalls, manual firmware upgrades, safety or compliance audits, or end-of-life communications. Assets affected are associated with a campaign using product service campaign items. Campaign and campaign items can then be associated with work orders and return orders as needed to complete the work.

[Field Service Warranty Management Data Model](#)

Use warranty items to record details of the labor, parts, and expenses, along with any exchange options, that are provided to rectify issues with products sold or installed. Create standard warranties for products and product families and, for products you install, record details of additional or extended warranties along with exclusions and void terms.

[Field Service Pricing Data Model](#)

Link work orders to products or assets in your org to track product pricing and work being performed on your customers' installed products.

[Field Service Object References](#)

Object reference for standard and custom Salesforce objects used in Field Service.

SEE ALSO:

[Salesforce SOAP API Developer Guide](#)

[Salesforce REST API Developer Guide](#)

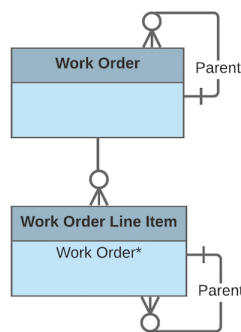
[Salesforce SOQL and SOSL Reference](#)

Field Service Core Data Model

Use Field Service's core objects to complete essential field service tasks such as managing work orders, defining your service territories, and tracking your workforce.

 **Note:** Asterisks in the following diagrams signify required fields.

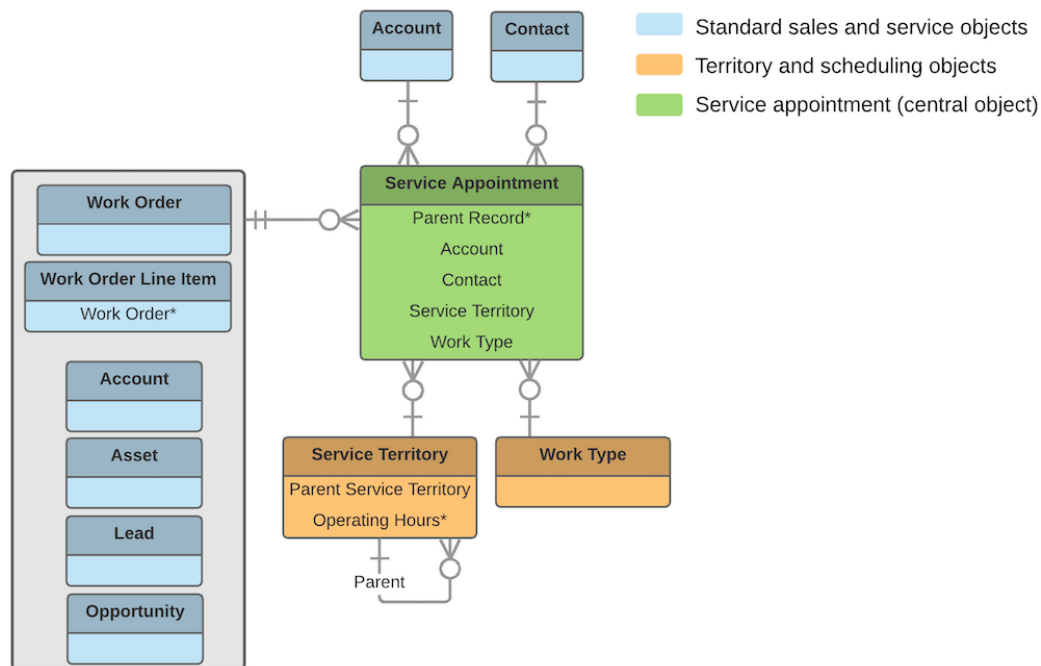
Work orders represent work to be completed for your customers, and are central to field service operations in Salesforce. To divide the work further for billing purposes or to track subtasks, add work order line items, which are child records of work orders.



Work orders offer a great deal of flexibility. They can be associated with many types of records, including:

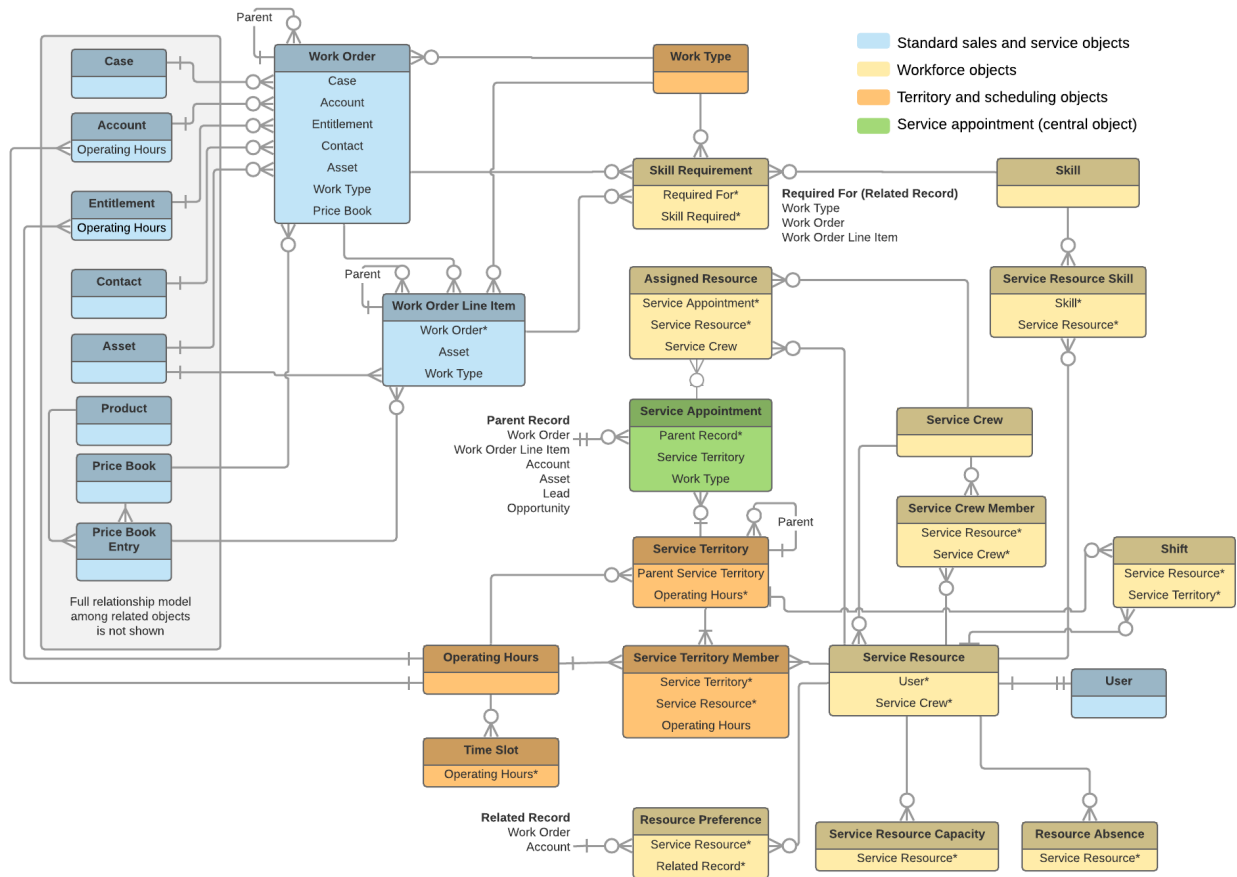
- *Assets*, to track work performed on a specific asset
- *Cases*, to indicate that the work is being performed as part of a customer case
- *Accounts* and *contacts*, representing the customer
- *Entitlements* and *service contracts*, to indicate that the work is being done to fulfill a service-level agreement

While work orders describe the work to be performed, *service appointments* represent the visits your team makes to the field to perform the work. They include scheduling settings such as an arrival window, scheduled start and end times, and appointment duration. Every service appointment has a parent record. The parent record is typically a work order or work order line item, but you can also add child service appointments to accounts, assets, leads, or opportunities to track related visits. A record can have multiple child service appointments; for example, a work order may have two service appointments if two visits were needed to complete the work.



If your team often performs the same tasks for multiple customers, create *work types* to standardize your field service work. Work types are templates that can be applied to work orders and work order line items. On a work type, you can define the duration of the work and add *skill requirements* to indicate the level of expertise needed to complete the work. You can also opt to auto-create a child service appointment on any record that uses the work type.

Work orders and service appointments are supported by a variety of objects that control when and where the work occurs, the nature of the work, and who performs it.



Who Performs the Work

Members of your mobile workforce are represented in Salesforce as *service resources*. A service resource represents an individual technician who can be assigned to a service appointment. You can also create *service crews*, which are groups of service resources with complementary skills and experience that can be assigned to appointments as a unit.

To assign a service resource to a service appointment, create an *assigned resource* record. Assigned resources contain lookups to a service resource and a service appointment. To assign a service crew to an appointment, first create a representative service resource record with a resource type of Crew. Then, create an assigned resource record that looks up to the Crew service resource.

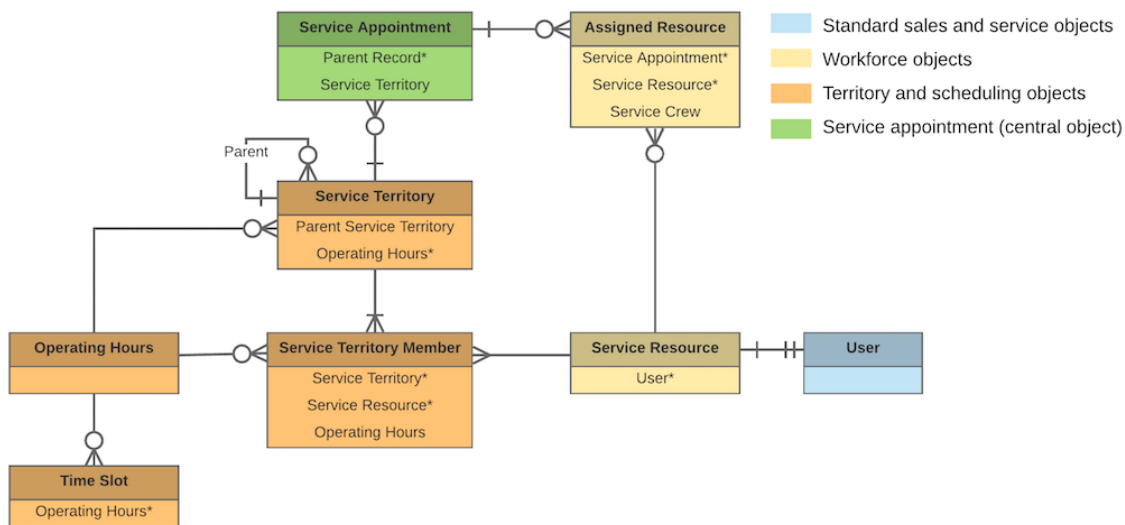
Service resources come with several objects that let you define their skills and availability:

- *Service resource skills* represent a service resource's certifications or levels of expertise.
- *Resource capacity* records track the hourly or job-based capacity of contractors.
- *Resource absences* represent time when a service resource needs to miss work.
- *Resource preferences* designate specific service resources as preferred, required, or excluded on a work order or account.

Where the Work Occurs

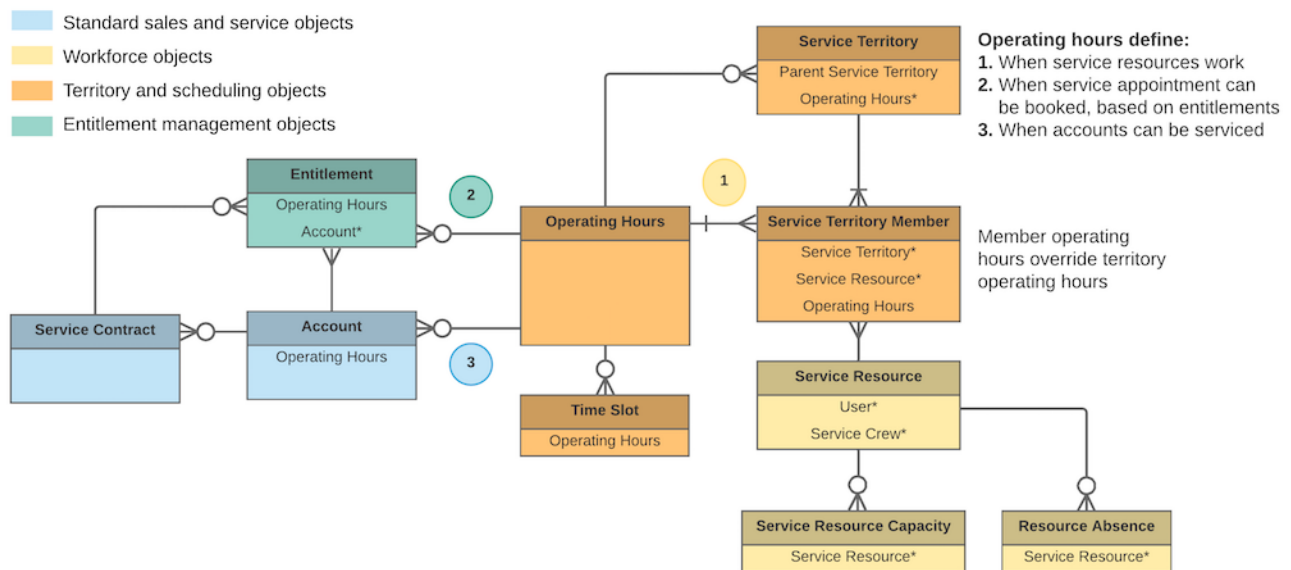
Service territories are the places where your team can perform field service work and are a way to organize your service resources. They typically represent geographic territories such as cities or counties but may also represent functional divisions like sales versus service.

A *work order* can be associated with one service territory. *Service resources* are assigned to one or more service territories as *service territory members* to indicate that they are available to work in the territory.



When the Work Occurs

Operating hours indicate when your team can perform field service work. They can be assigned to accounts, service territories, and service territory members. To add detail to operating hours, create *time slots*, which represent the hours of operation in a particular day.



If a customer *entitlement* includes terms regarding when the customer has access to field service, you can track these hours on the entitlement using the **Operating Hours** field (API name: `SvcApptBookingWindowsId`). For example, if Customer A is entitled to service Monday through Friday from 8 AM to noon, but Customer B is entitled to 24/7 service, you can create operating hours for each customer and assign them to the related entitlement.

SEE ALSO:

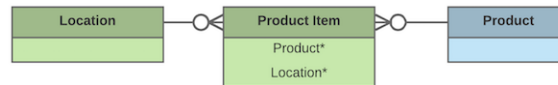
[Guidelines for Creating Operating Hours for Field Service](#)

Field Service Inventory Management Data Model

Use Field Service's inventory objects to track the storage, request, consumption, return, and retirement of items in your inventory.

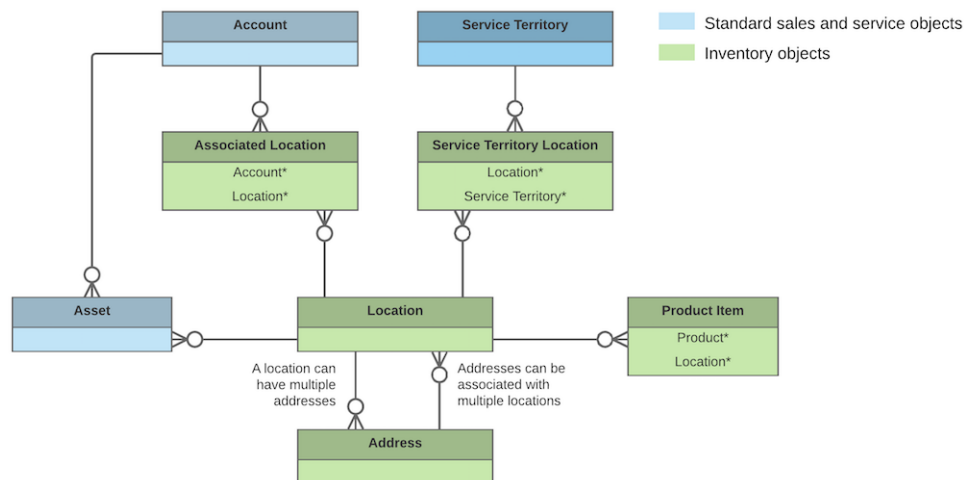
 **Note:** Asterisks in the following diagrams signify required fields.

Inventory management starts with *product items*, which represent the stock of a particular product at a particular location. Each product item is associated with a *product* and a *location* in Salesforce. For example, if you have 50 hammers stored in your Warehouse A location and 200 stored in Warehouse B, create one product item for each location. Product items list a quantity at the location that is updated automatically when inventory is transferred or consumed.



If the Inventory Location option is selected on a location, it means that inventory can be stored there. Product items can be associated only with inventory locations.

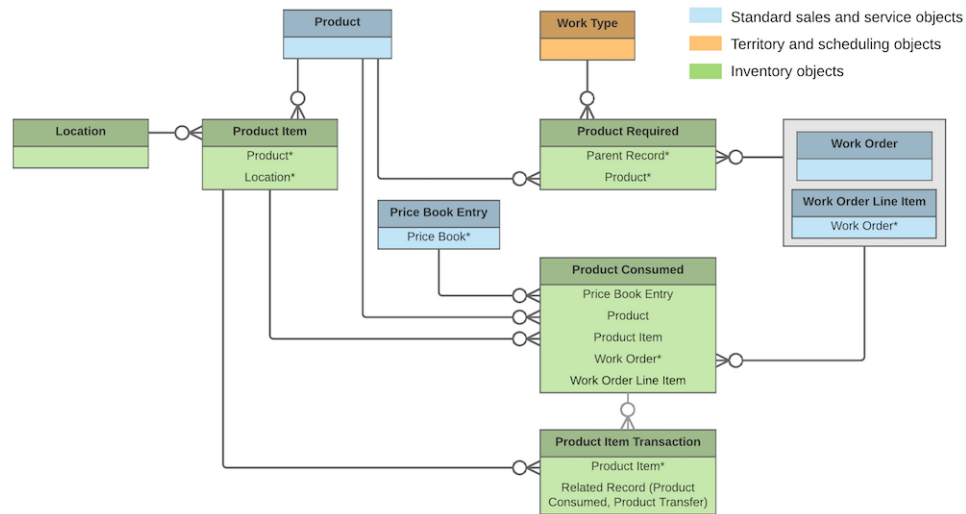
You can link a location to multiple *accounts* and *service territories*. For example, if a location is a shopping mall, you can choose to associate it with every account that operates a store in the mall. You can also create *addresses* for a location, such as a mailing and home address. And to keep track of customer sites, create *associated locations*, which contain lookups to an *account* and a *location*.



If a particular product is needed to complete a field service job, add *products required* to ensure that the assigned service resources arrive prepared. Products required can be child records of *work orders*, *work order line items*, and *work types*. Work order and work order line items inherit their work type's products required.

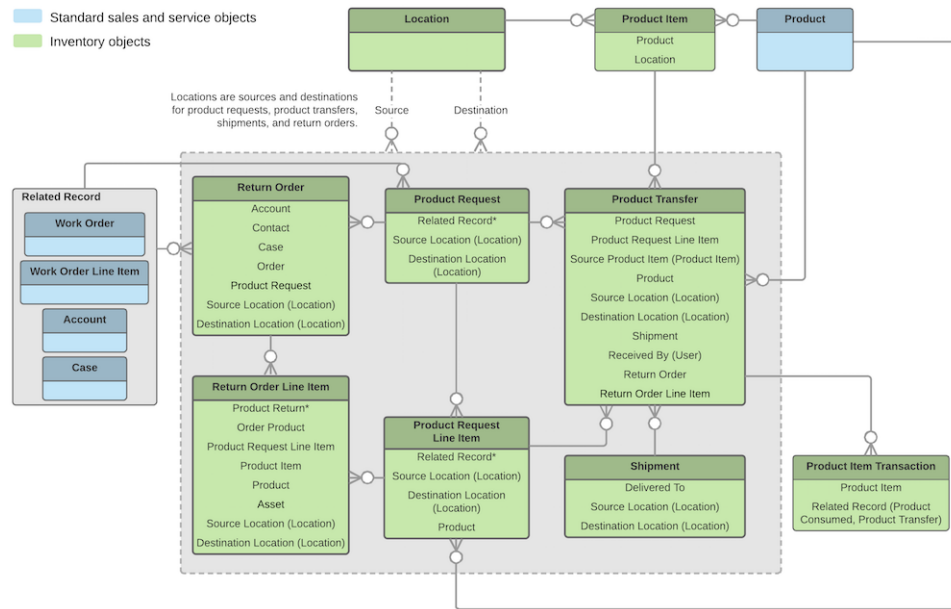
When a product is consumed during the completion of a work order, track its consumption by creating a *product consumed* record. You can add products consumed to work orders or work order line items. Track product consumption at the line item level if you want to know which products were used for each line item's tasks.

The way you use products consumed depends on how closely you want to track the state of your inventory in Salesforce. If you want to track the entire life cycle of items in your inventory, including their storage, transfer, and consumption, link your product consumed records to product items. This approach ensures that your inventory numbers update automatically to reflect the consumption of products from your inventory. If you want to track product consumption only, however, specify a **Price Book Entry** on each product consumed record and leave the **Product Item** field blank.



The movement of items into and out of your inventory and between locations is tracked using these objects:

- *Product requests* are orders for products, which you might create when stock is running low.
- *Product request line items* are subdivisions of a product request.
- *Product transfers* track the movement of product items between inventory locations.
- *Shipments* represent the shipment of product items between locations.
- *Product item transactions* describe actions performed on a product item. They're auto-generated records that help you track when inventory is replenished, consumed, or adjusted.
- *Return orders* track the return of a product item due to damage, order errors, or other reasons.
- *Return order line items* are subdivisions of a return order.



SEE ALSO:

- [Set Up Your Field Service Inventory](#)
- [Guidelines for Transferring Inventory](#)
- [Guidelines for Consuming Inventory](#)
- [Common Inventory Management Tasks](#)

Field Service Preventive Maintenance Data Model

Manage periodic maintenance with the help of maintenance plans, which define the maintenance schedule for specific assets. Maintenance plans typically reflect the terms in a customer's service contract or entitlements.

Note: Asterisks in the following diagram signify required fields.

A *maintenance plan* can be linked to an *account* and *contact*—representing the customer—and, if the customer has one, to a *service contract*.

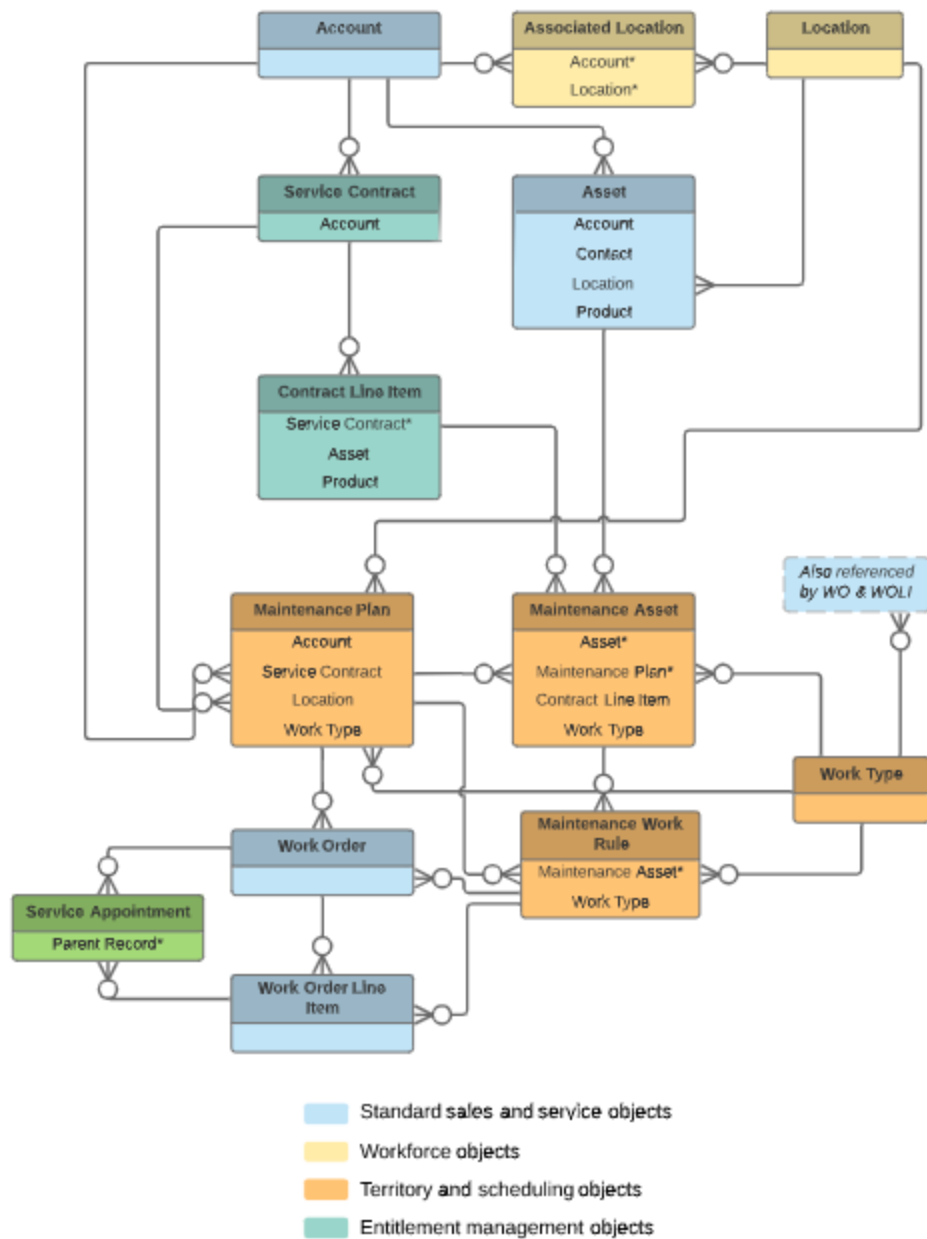
A maintenance plan can cover multiple *assets*. For example, you can create a maintenance plan to track the monthly maintenance of 20 laser printers installed in a customer's office building. The assets covered by a maintenance plan are represented as child records of the maintenance plan called *maintenance assets*. For more complex reoccurring maintenance, you can define *maintenance work rules* for most assets and maintenance plans. With this feature you can, for example, set up a schedule for minor monthly maintenance and a major yearly service. You can also link a maintenance plan to a *location* to indicate where the assets are installed. For example, link a plan to a location of the Site type that represents the office building where the printers are installed.

After you create a maintenance plan or maintenance work rules, it's time to generate *work orders* for the planned maintenance visits. Maintenance plans come with a Generate Work Orders quick action, which can also be called with Apex code. Your maintenance plan settings determine how many work orders and work order line items are generated at once and what their settings are. You can opt to generate one work order per maintenance asset for each visit, or a parent work order for each visit with one work order line item per maintenance asset.

Maintenance plans, maintenance assets, and maintenance work rules can be associated with a *work type*:

- If you specify a work type on a maintenance plan, the plan's work orders use that work type.

- If you specify a work type on a maintenance asset, generated work orders that are associated with the maintenance asset use the maintenance asset's work type.
- If you specify a work type on a maintenance work rule, generated work orders that are associated with the maintenance work rule use the maintenance work rule's work type.



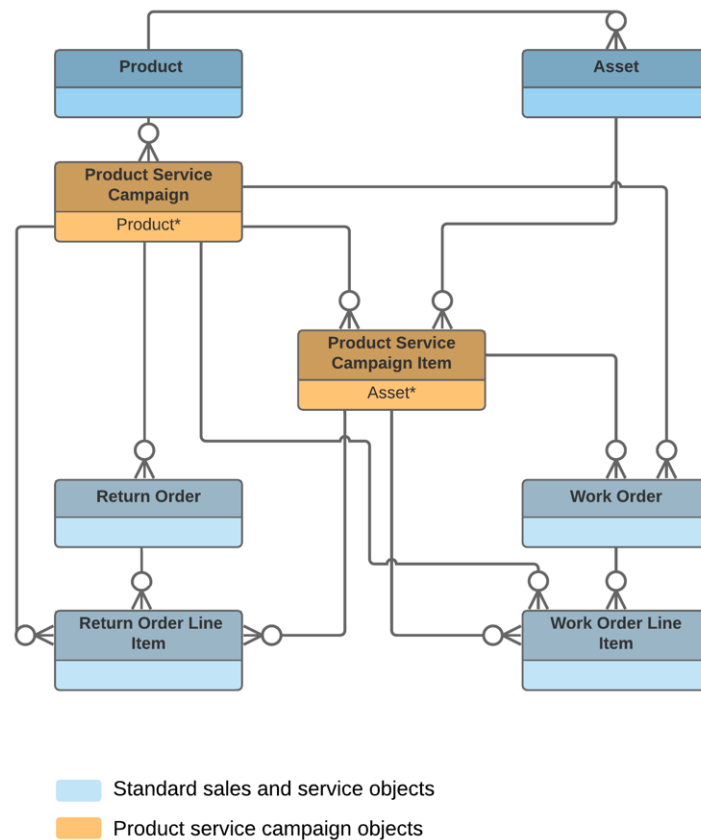
SEE ALSO:

[Generate Work Orders on Maintenance Plans with Apex](#)

Field Service Product Service Campaign Data Model

Use product service campaign to record the actions to address situations such as product recalls, manual firmware upgrades, safety or compliance audits, or end-of-life communications. Assets affected are associated with a campaign using product service campaign items. Campaign and campaign items can then be associated with work orders and return orders as needed to complete the work.

 **Note:** Asterisks mean these fields are required.

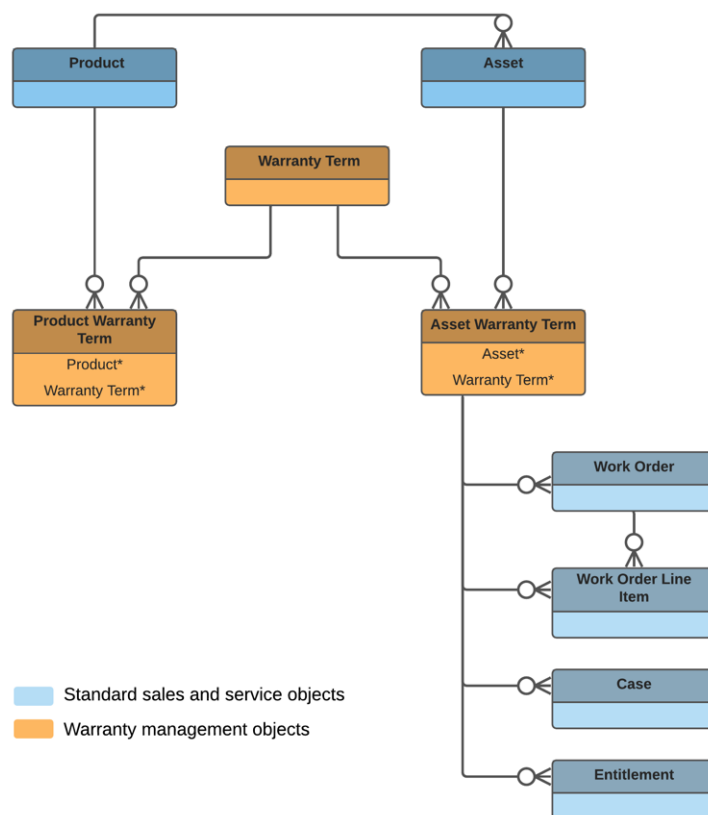


Field Service Warranty Management Data Model

Use warranty items to record details of the labor, parts, and expenses, along with any exchange options, that are provided to rectify issues with products sold or installed. Create standard warranties for products and product families and, for products you install, record details of additional or extended warranties along with exclusions and void terms.

 **Note:** Asterisks in the following diagram signify required fields.

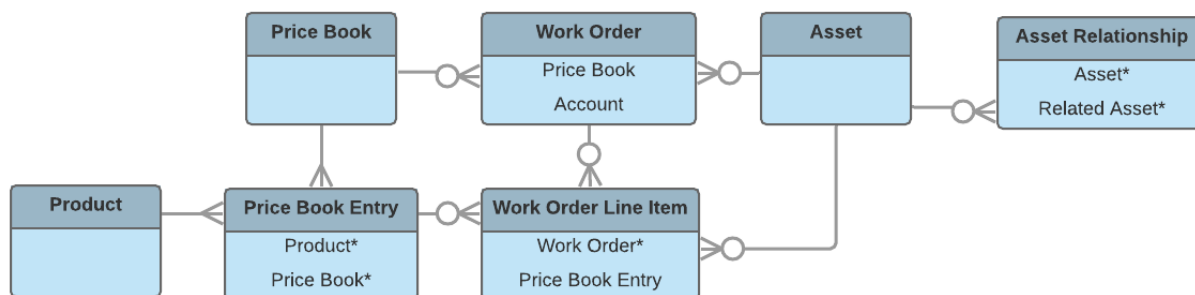
A *warranty term* is linked to a *product warranty term* to define the standard warranty offered for a product or product family. When a product is installed, *asset warranty term* details are created from the standard warranty. An asset warranty term can be associated with a work order, work order line item, case, or entitlement to track actions related to the fulfillment of the warranty term provisions.



Field Service Pricing Data Model

Link work orders to products or assets in your org to track product pricing and work being performed on your customers' installed products.

 **Note:** Asterisks in the following diagram signify required fields.



If you've set up a product catalog in Salesforce to track the goods and services your business offers, you can associate items in your price books with work orders and their line items, similar to the way you can associate products with opportunities or orders. If you specify a *price book* on a *work order*, this allows you to link each *work order line item* to a *price book entry* (product) from the price book. List price, discount, and quantity are defined at the line-item level.

For example, if you create a work order for a solar panel installation, select a price book in the Price Book lookup field on the work order. Then, use the Price Book Entry lookup field on its work order line items to select goods or services listed in your price book, such as Site Assessment, Solar Panel, and Inverter. A quick glance at a completed work order's line items shows you which products from your product catalog were sold as part of the work order.

After a product is purchased and installed for a customer, it is typically tracked as an *asset* in Salesforce. The Asset lookup field on work orders and work order line items allows you to track work being performed on a specific asset. It also makes it possible to view a history of all work completed on the asset.

If an asset is replaced or upgraded, the relationship between the old and new asset is tracked in an *asset relationship* record. An asset relationship lists a start and end time—for instance, if the replacement asset is being leased—and a relationship type, which must be defined by the admin.

SEE ALSO:

[Work Order Pricing Guidelines for Field Service](#)

[Equal Asset Relationships](#)

Field Service Object References

Object reference for standard and custom Salesforce objects used in Field Service.

IN THIS SECTION:

[Field Service Standard and Custom Objects](#)

A list of standard Salesforce objects and Salesforce-managed custom objects used in Field Service.

[Field Service Custom Fields on Standard Objects](#)

A list of custom fields on standard Salesforce objects installed with the Field Service managed package.

[Supplementary Field Service Objects](#)

A list of Field Service objects that support history tracking or sharing.

Field Service Standard and Custom Objects

A list of standard Salesforce objects and Salesforce-managed custom objects used in Field Service.

The following list links to reference information for the standard objects and Salesforce-managed custom objects. Some objects are specific to Field Service, while others are used across a variety of Salesforce features.



Note: Most objects are available only if Field Service is enabled. Objects not tied to Field Service enablement are shown with an asterisk (*).

IN THIS SECTION:

[Address](#)

Represents a mailing, billing, or home address.

[ApptBundleAggrDurDnscale](#)

Sums the duration of the bundle members, reduced by a predefined percentage. This object is available in API version 54.0 and later.

[ApptBundleAggrPolicy](#)

Policy that defines how the property values of the bundle members are aggregated and assigned to the bundle. This object is available in API version 54.0 and later.

[ApptBundleConfig](#)

Represents the general parameters that define the behavior of the bundle. This object is available in API version 54.0 and later.

[ApptBundlePolicy](#)

Policy that defines how the bundling of service appointments should be handled. This object is available in API version 54.0 and later.

[ApptBundlePolicySvcTerr](#)

Represents a link between the BundlePolicy and the ServiceTerritory. This object is available in API version 54.0 and later.

[ApptBundlePropagatePolicy](#)

Policy that defines which property values are inherited from the bundle to the bundle members or are assigned as constant values in the bundle members. This object is available in API version 55.0 and later.

[ApptBundleRestrictPolicy](#)

Policy that defines the restrictions that are considered while forming a bundle. This object is available in API version 54.0 and later.

[ApptBundleSortPolicy](#)

Policy that defines the properties by which the bundle members are sorted within the bundle. Can also be used in the automatic mode for determining the order of the automatic selection of bundle members. This object is available in API version 54.0 and later.

[AppExtension](#)

Represents a connection between the Field Service mobile app and another app, typically for passing record data to the Salesforce mobile app or other apps. This object is available in API version 41.0 and later.

[Asset*](#)

Represents an item of commercial value, such as a product sold by your company or a competitor, that a customer has purchased.

[AssetAccountParticipant](#)

Represents a junction between the Asset and Account objects describing the association between a participating account and an asset. This object is available in API version 56.0 and later.

[AssetAttribute](#)

Stores asset attributes to track and analyze asset conditions to improve their uptime. This object is available in API version 57.0 and later.

[AssetContactParticipant](#)

Represents a junction between the Asset and Contact objects describing the association between a participating contact and an asset. This object is available in API version 56.0 and later.

[AssetDowntimePeriod*](#)

Represents a period during which an asset is not able to perform as expected. Downtime periods include planned activities, such as maintenance, and unplanned events, such as mechanical breakdown. This object is available in API version 49.0 and later.

[AssetRelationship*](#)

Represents a non-hierarchical relationship between assets due to an asset modification; for example, a replacement, upgrade, or other circumstance. In Revenue Lifecycle Management, this object represents an asset or assets grouped in a bundle or set. This object is available in API version 41.0 and later.

[AssetWarranty](#)

Defines the warranty terms applicable to an asset along with any exclusions and extensions. This object is available in API version 50.0 and later.

[AssignedResource](#)

Represents a service resource who is assigned to a service appointment in Field Service and Lightning Scheduler. Assigned resources appear in the Assigned Resources related list on service appointments. This object is available in API version 38.0 and later.

[AssociatedLocation](#)

Represents a link between an account and a location in Field Service. You can associate multiple accounts with one location. For example, a shopping center location may have multiple customer accounts.

[AttributeDefinition](#)

Represents a product, asset, or object attribute, for example, a hardware specification or software detail. This object is available in API version 57.0 and later.

[AttributePicklist](#)

Represents a custom picklist for an asset attribute. This object is available in API version 57.0 and later.

[AttributePicklistValue](#)

Represents the values of an asset attribute picklist. This object is available in API version 57.0 and later.

[ContractLineItem](#)*

Represents a product covered by a service contract (customer support agreement). This object is available in API version 18.0 and later.

[ContractLineOutcome](#)

Represents information on a contract line outcome's captured data and other related parameters that are used when capturing data. This object is available in API version 58.0 and later.

[ContractLineOutcomeData](#)

Represents the contract line outcome's captured data. It stores the data that was captured between the contract line outcome's start date and end date. This object is available in API version 58.0 and later.

[DigitalSignature](#)

Represents a signature captured on a service report in field service.

[Entitlement](#)*

Represents the customer support an account or contact is eligible to receive. This object is available in API version 18.0 and later. Entitlements may be based on an asset, product, or service contract.

[EntityMilestone](#)*

Represents a required step in a customer support process on a work order. The Salesforce user interface uses the term "object milestone." This object is available in API version 37.0 and later.

[Expense](#)

Represents an expense linked to a work order. Service resource technicians can log expenses, such as tools or travel costs. This object is available in API version 49.0 and later.

[ExpenseReport](#)

Represents a report that summarizes expenses. This object is available in API version 50.0 and later.

[ExpenseReportEntry](#)

Represents an entry in an expense report. This object is available in API version 50.0 and later.

[FieldServiceMobileSettings](#)

Represents a configuration of settings that control the Field Service iOS and Android mobile app experience. This object is available in API version 38.0 and later.

[FldSvcObjChg](#)

Represents a change made to one of a service appointment's tracked fields. This object is available in API version 63.0 and later.

[FldSvcObjChgDtl](#)

Represents the details of a change made to one of a service appointment's tracked fields. This object is available in API version 63.0 and later.

[FSL__Time_Dependency__c](#)

Represents a dependency between two service appointments. This object is used to define scheduling relationships between two appointments. It allows you to determine the order and timing in which dependent appointments should be scheduled.

[GeolocationBasedAction](#)

Represents a geolocation-based action, which is an action that's triggered when a user enters, exits, or is within the area of the associated object. Available in API version 61.0 and later.

[LinkedArticle](#)

Represents a knowledge article that is attached to a work order, work order line item, or work type. This object is available in API version 37.0 and later.

[Location](#)

Represents a warehouse, service vehicle, work site, or other element of the region where your team performs field service work. In API version 49.0 and later, you can associate activities with specific locations. Activities, such as the tasks and events related to a location, appear in the activities timeline when you view the location detail page. Also in API version 49.0 and later, Work.com users can view Employees as a related list on Location records. In API version 51.0 and later, this object is available for Omnichannel Inventory and represents physical locations where inventory is available for fulfilling orders.

[MaintenanceAsset](#)

Represents an asset covered by a maintenance plan in field service. Assets can be associated with multiple maintenance plans.

[MaintenancePlan](#)

Represents a preventive maintenance schedule for one or more assets in field service.

[MaintenanceWorkRule](#)

Represents the recurrence pattern for a maintenance record. This object is available in API version 49.0 and later.

[MobileSettingsAssignment](#)

Represents the assignment of a particular field service mobile settings configuration to a user profile. This object is available in API version 41.0 and later.

[OperatingHours](#)

Represents the hours in which a service territory, service resource, or account is available for work. OperatingHours is used by Field Service, Salesforce Scheduler, Salesforce Meetings, Sales Engagement, and Workforce Engagement. This object is available in API version 38.0 and later.

[OperatingHoursHoliday](#)

Represents the day or hours for which a service territory or service resource is unavailable in Field Service, Salesforce Scheduler, Salesforce Meetings, Sales Engagement, or Workforce Engagement. This object is available in API version 54.0 and later.

[Pricebook2*](#)

Represents a price book that contains the list of products that your org sells.

[Product2*](#)

Represents a product that your company sells.

[ProductConsumed](#)

Represents an item from your inventory that was used to complete a work order or work order line item in field service.

[ProductConsumedState](#)

Represents the status of an item from your inventory that was used to complete a work order or work order line item in Field Service. This object is available in API version 57.0 and later.

[ProductItem](#)

Represents the stock of a particular product at a particular location in field service, such as all bolts stored in your main warehouse.

[ProductItemTransaction](#)

Represents an action taken on a product item in field service. Product item transactions are auto-generated records that help you track when a product item is replenished, consumed, or adjusted.

[ProductRequest](#)

Represents an order for a part or parts in field service.

[ProductRequestLineItem](#)

Represents a request for a part in field service. Product request line items are components of product requests.

[ProductRequired](#)

Represents a product that is needed to complete a work order or work order line item in field service.

[ProductServiceCampaign](#)

Represents a set of activities to be performed on a product service campaign asset, such as a product recall for safety issues or product defects. This object is available in API version 51.0 and later.

[ProductServiceCampaignItem](#)

Represents a product service campaign's asset. This object is available in API version 51.0 and later.

[ProductServiceCampaignItemStatus](#)

Represents a status for a product service campaign item in field service. This object is available in API version 51.0 and later.

[ProductServiceCampaignStatus](#)

Represents a status for a product service campaign in field service. This object is available in API version 51.0 and later.

[ProductTransfer](#)

Represents the transfer of inventory between locations in field service.

[ProductWarrantyTerm](#)

Defines the relationship between a product or product family and warranty term. This object is available in API version 50.0 and later.

[RecordsetFilterCriteria](#)

Represents a set of filters that can be used to match service appointments or assets based on your criteria fields. For example, you can create recordset filter criteria so that only service appointments that satisfy the filter criteria are matched to the filtered shifts, and likewise only maintenance work rules that satisfy your criteria are matched to assets. This object is available in API version 50.0 and later. Assets and maintenance work rules are available in API version 52.0 and later.

[RecordsetFilterCriteriaRule](#)

Represents a rule using fields from the designated source object to create filters on the filtered, or target, object.

RecordsetFilterCriteriaRule is associated with the RecordsetFilterCriteria object. This object is available in API version 50.0 and later.

[RecordsetFiltrCritMonitor](#)

Monitors whether the value of an asset attribute is within the threshold of a recordset filter criteria (RFC). You can monitor one or more RFCs for an Asset. This object is available in API version 57.0 and later.

[ResourceAbsence](#)

Represents a time period in which a service resource is unavailable to work in Field Service, Salesforce Scheduler, or Workforce Engagement. This object is available in API version 38.0 and later.

[ResourcePreference](#)

Represents an account's preference for a specified service resource on field service work.

[ReturnOrder](#)

Represents the return or repair of inventory or products in Field Service, or the return of order products in Order Management. This object is available in API version 42.0 and later.

[ReturnOrderLineItem](#)

Represents a specific product that is returned or repaired as part of a return order in Field service, or a specific order item that is returned as part of a return order in Order Management. This object is available in API version 42.0 and later.

[SerializedProduct](#)

Records serial numbers for each individual product in an inventory. This object is available in API version 50.0 and later.

[SerializedProductTransaction](#)

Represents transactions performed on a serialized product. This object is available in API version 57.0 and later.

[ServiceAppointment](#)

Represents an appointment to complete work for a customer in Field Service, Lightning Scheduler, Intelligent Appointment Management, and Virtual Care. This object is available in API version 38.0 and later.

[ServiceAppointmentStatus](#)

Represents a possible status of a service appointment in field service.

[ServiceContract*](#)

Represents a customer support contract (business agreement). This object is available in API version 18.0 and later.

[ServiceCrew](#)

Represents a group of service resources who can be assigned to service appointments as a unit.

[ServiceCrewMember](#)

Represents a technician service resource that belongs to a service crew.

[ServiceReport](#)

Represents a report that summarizes a work order, work order line item, or service appointment.

[ServiceReportLayout](#)

Represents a service report template in field service.

[ServiceResource](#)

Represents a service technician or service crew in Field Service and Salesforce Scheduler, or an agent in Workforce Engagement. This object is available in API version 38.0 and later.

[ServiceResourceCapacity](#)

Represents the maximum number of scheduled hours or number of service appointments that a capacity-based service resource can complete within a specific time period. This object is available in API version 38.0 and later.

[ServiceResourceSkill](#)

Represents a skill that a service resource possesses in Field Service and Lightning Scheduler. This object is available in API version 38.0 and later.

[ServiceTerritory](#)

Represents a geographic or functional region in which work can be performed in Field Service, Salesforce Scheduler, or Workforce Engagement. This object is available in API version 38.0 and later.

[ServiceTerritoryLocation](#)

Represents a location associated with a particular service territory in field service.

[ServiceTerritoryMember](#)

Represents a service resource who can be assigned in a service territory in Field Service, Salesforce Scheduler, or Workforce Engagement. This object is available in API version 38.0 and later.

[Shift](#)

Represents a shift for service resource scheduling. Available in API versions 46.0 and later.

[ShiftPattern](#)

Represents a pattern of templates for creating shifts. This object is available in API version 51.0 and later.

[ShiftPatternEntry](#)

ShiftPatternEntry links a shift template to a shift pattern. This object is available in API version 51.0 and later.

[ShiftTemplate](#)

Represents a template for creating shifts. This object is available in API version 51.0 and later.

[Shipment](#)

Represents the transport of inventory in field service or a shipment of order items in Order Management.

[Skill*](#)

Represents a category or group of Chat users or service resources in Field Service or Workforce Engagement. This object is available in API version 24.0 and later.

[SkillRequirement](#)

Represents a skill that is required to complete a particular task in Field Service, Omni-Channel, Salesforce Scheduler, or Workforce Engagement. Skill requirements can be added to pending service routing objects in Omni-Channel. They can be added to work types, work orders, and work order line items in Field Service and Lightning Scheduler. And they can be added to job profiles in Workforce Engagement. This object is available in API version 38.0 and later. You also can add skill requirements to work items in Omni-Channel skills-based routing using API version 42.0 and later.

[TimeSheet](#)

Represents a schedule of a service resource's time in Field Service or Workforce Engagement. This object is available in API v47.0 and later.

[TimeSheetEntry](#)

Represents a span of time that a service resource spends on a field service task. This object is available in API version 47.0 and later.

[TimeSlot](#)

Represents a period of time on a specified day of the week during which work can be performed in Field Service, Salesforce Scheduler, or Workforce Engagement. Operating hours consist of one or more time slots. This object is available in API version 38.0 and later.

[TravelMode](#)

Represents a travel mode used for travel time calculations. The records include information about the type of transportation (such as Car or Walking), whether a vehicle can take toll roads, and whether a vehicle is transporting hazardous materials. This object is available in API version 54.0 and later.

[WarrantyTerm](#)

Represents warranty terms defining the labor, parts, and expenses covered, along with any exchange options, provided to rectify issues with products. This object is available in API version 50.0 and later.

[WorkCapacityAvailability](#)

Represents the available work capacity for a specific time and service territory. This object is available in API version 59.0 and later.

[WorkCapacityLimit](#)

Represents the capacity limit in a specific service territory for a workstream or for the whole service territory in a given period. This object is available in API version 59.0 and later.

[WorkCapacityUsage](#)

Represents the capacity usage in a specific service territory for a workstream or for the whole service territory in a given period. This object is available in API version 59.0 and later.

[WorkOrder*](#)

Represents field service work to be performed for a customer. This object is available in API version 36.0 and later.

[WorkOrderLineItem*](#)

Represents a subtask on a work order in field service. This object is available in API version 36.0 and later.

[WorkOrderLineItemStatus](#)

Represents a possible status of a work order line item in field service.

[WorkPlan](#)

Represents a work plan for a work order or work order line item. This object is available in API version 52.0 and later.

[WorkPlanSelectionRule](#)

Represents a rule that selects a work plan for a work order or work order line item. This object is available in API version 52.0 and later.

[WorkPlanTemplate](#)

Represents a template for a work plan. This object is available in API version 52.0 and later.

[WorkPlanTemplateEntry](#)

Represents an object that associates a work step template with a work plan template. This object is available in API version 52.0 and later.

[WorkOrderStatus](#)

Represents a possible status of a work order in field service.

[WorkStep](#)

Represents a work step in a work plan. This object is available in API version 52.0 and later.

[WorkStepStatus](#)

Represents a picklist for a status category on a work step. This object is available in API version 52.0 and later.

[WorkStepTemplate](#)

Represents a template for a work step. This object is available in API version 52.0 and later.

[WorkType](#)

Represents a type of work to be performed in Field Service and Lightning Scheduler. Work types are templates that can be applied to work order or work order line items. This object is available in API version 38.0 and later.

[WorkTypeGroup](#)

Represents a grouping of work types used to categorize types of appointments available in Lightning Scheduler, or to define scheduling limits in Field Service. This object is available in API version 45.0 and later.

[WorkTypeGroupMember](#)

Represents the relationship between a work type and the work type group it belongs to. This object is available in API version 45.0 and later.

Address

Represents a mailing, billing, or home address.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

The following access checks must be enabled:

- Industries Insurance
- Retail Execution
- Industries Visit
- Field Service
- Order Management
 - Perms: FulfillmentOrder, OrderSummary, AdvancedOrderManagement, OrderCCS
 - Prefs: OrdersEnabled, EnhancedCommerceOrders
- Public Sector
- Employee Experience
- Contact Tracing For Employees

You can create an address only when creating a location.

Fields

Field Name	Details
Address	Type address Properties Filter, Nillable Description The full address.
AddressType	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description Picklist of address types. The values are: • Mailing • Shipping • Billing • Home
City	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The address city.

Field Name	Details
Country	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The address country.
Description	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description A brief description of the address.
DrivingDirections	Type string Properties Create, Filter, Nillable, Sort, Update Description Directions to the address.
GeocodeAccuracy	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The level of accuracy of a location's geographical coordinates compared with its physical address. A geocoding service typically provides this value based on the address's latitude and longitude coordinates.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The most recent date on which a user referenced this record.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort

Field Name	Details
	Description The most recent date on which a user viewed this record.
Latitude	Type double Properties Create, Filter, Nillable, Sort, Update Description Used with <code>Longitude</code> to specify the precise geolocation of the address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places.
LocationType	Type picklist Properties Create, Defaulted on create, Filter, Filter, Group, Sort, Update Description Picklist of location types. The available values are: • Warehouse (default) • Site • Van • Plant
Longitude	Type double Properties Create, Filter, Nillable, Sort, Update Description Used with <code>Latitude</code> to specify the precise geolocation of the address. Acceptable values are numbers between –180 and 180 with up to 15 decimal places.
Name	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description Name of the address.
ParentId	Type reference

Field Name	Details
	Properties Create, Filter, Group, Sort Description A lookup field to the parent location. This is a relationship field. Relationship Name Parent Relationship Type Lookup Refers To Location
PostalCode	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The address postal code.
State	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The address state.
Street	Type textarea Properties Create, Filter, Group, Nillable, Sort, Update Description The address street.
TimeZone	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Picklist of available time zones.

Usage

 **Important:** “Address” in Salesforce can also refer to the Address compound field found on many standard objects. When referencing the Address object in your Apex code, always use `Schema.Address` instead of `Address` to prevent confusion with the standard Address compound field. If referencing both the address object and the Address field in the same snippet, you can differentiate between the two by using `System.Address` for the field and `Schema.Address` for the object.

Associated Object

This object has the following associated object. If the API version isn’t specified, they’re available in the same API versions as this object. Otherwise, they’re available in the specified API version and later.

AddressHistory (API version 62.0)

History is available for tracked fields of the object.

ApptBundleAggrDurDnscale

Sums the duration of the bundle members, reduced by a predefined percentage. This object is available in API version 54.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

- Field Service must be enabled.
- Bundling must be enabled in the Field Service Settings.
- The Field Service Admin, Field Service Bundle for Dispatcher, and Field Service Integration permission sets must be enabled.

Fields

Field	Details
BundleAggregationPolicyId	Type reference Properties Create, Filter, Group, Sort Description The ID of the parent appointment bundle aggregation policy. This is a relationship field. Relationship Name BundleAggregationPolicy Relationship Type Lookup Refers To ApptBundleAggrPolicy

Field	Details
FromBundleMemberNumber	Type int Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description The number of the first bundle member to which the downscale is applied.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last accessed this record, a record related to this record, or a list view.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last viewed this record or list view. If this value is null, the user might have only accessed this record or list view (<code>LastReferencedDate</code>) but not viewed it.
MaxReduction	Type int Properties Create, Filter, Group, Nillable, Sort, Update Description The maximum reduction that can be applied to a bundle member.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The name of the appointment bundle aggregation downscale policy.
PercentageOfReduction	Type int Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The percentage of duration reduction.
ToBundleMemberNumber	Type int
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The number of the last bundle member to which the downscale is applied.

ApptBundleAggrPolicy

Policy that defines how the property values of the bundle members are aggregated and assigned to the bundle. This object is available in API version 54.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

- Field Service must be enabled.
- Bundling must be enabled in the Field Service Settings.
- The Field Service Admin, Field Service Bundle for Dispatcher, and Field Service Integration permission sets must be enabled.

Fields

Field	Details
AggregationAction	Type picklist
	Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description The aggregation action to be performed. Possible values are: All default and custom Service Appointment fields.
AggregationFieldType	Type picklist
	Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details
	Description The target field type in the bundle to which the aggregation is directed. Possible values are: • Boolean • Date • Numeric • Picklist • Picklist-Multi • Skills • String
AggregationOrder	Type int Properties Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update Description The order the aggregation is triggered.
BundleFieldName	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Name of the target field in the bundle where the value is taken from the bundle member. Possible values are: All default and custom Service Appointment fields.
BundleMemberAddiFieldName	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Name of an additional source field that is connected to the initial source field in the bundle member from which the value is taken. Possible values are: All default and custom Service Appointment fields.
BundleMemberFieldName	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details
	Description Name of the source field in the bundle member from which the value is taken. Possible values are: All default and custom Service Appointment fields.
BundlePolicyId	Type reference Properties Create, Filter, Group, Sort Description ID of the parent bundle policy. This is a relationship field. Relationship Name BundlePolicy Relationship Type Lookup Refers To ApptBundlePolicy
ConstantValue	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The constant value that is used in the aggregation.
DateValue	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Represents how the date value will be determined. Possible values are: • End of Day • Now • Null • Start of Day
DoesAllowDuplicateStrings	Type boolean

Field	Details
	Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates if you want to allow the same string to appear more than once when using the 'Sum based on Bundle Members' action type.
DownscaleSortDirection	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Applies only if the Set Downscaled Duration action is set. The downscaling sorting direction of the bundle member service appointments, according to their duration. Possible values are: • Ascending • Descending
FilterCriteriaId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The active recordset filter criteria used for aggregating the bundle members. This is a relationship field. Relationship Name FilterCriteria Relationship Type Lookup Refers To RecordsetFilterCriteria
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last accessed this record, a record related to this record, or a list view.
LastViewedDate	Type dateTime

Field	Details
	Properties Filter, Nillable, Sort Description The timestamp when the current user last viewed this record or list view. If this value is null, the user might have only accessed this record or list view (<code>LastReferencedDate</code>) but not viewed it.
<code>MaxBundleDuration</code>	Type int Properties Create, Filter, Group, Nillable, Sort, Update Description The maximum bundle duration that can be accumulated from the bundle members (after downscaling).
<code>Name</code>	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The name of the appointment bundle aggregation policy.
<code>ShouldUpdateOnCreationOnly</code>	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates if you want to update the field in the bundle only when it is created.

ApptBundleConfig

Represents the general parameters that define the behavior of the bundle. This object is available in API version 54.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

- Field Service must be enabled.
- Bundling must be enabled in the Field Service Settings.
- The Field Service Admin, Field Service Bundle for Dispatcher, and Field Service Integration permission sets must be enabled.

Fields

Field	Details
AddToBundleStatuses	Type multipicklist Properties Create, Defaulted on create, Filter, Nillable, Update Description The statuses of service appointment that are allowed to be bundled. Possible values are: • Accepted • Canceled • Cannot Complete • Completed • Dispatched • In Progress • None • Rejected • Scheduled The default value is None.
BundleStatusesToPropagate	Type multipicklist Properties Create, Defaulted on create, Filter, Nillable, Update Description The bundle statuses that when updated are inherited by the bundle members. Possible values are: • Accepted • Canceled • Cannot Complete • Completed • Dispatched • In Progress • None • Rejected • Scheduled The default value is None.

Field	Details
CriteriaForAutoUnbundlingId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The criteria that causes a bundle service appointment to be unbundled. This is a relationship field. Relationship Name CriteriaForAutoUnbundling Relationship Type Lookup Refers To RecordsetFilterCriteria
DoesAddTravelTime	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description If the bundle members aren't in the same location, add travel time between them to the bundle's duration according to their sort order. The default value is false.
DoesDeleteEmptyBundles	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description If the bundle has no remaining bundle members, the bundle is deleted.
EmptyBundleStatus	Type picklist Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The status from the Canceled category that a bundle service appointment changes to if it has no remaining bundle members, but still appears in the appointment list. Possible values are determined by the org's statuses. The default value is None.
LastReferencedDate	Type dateTime

Field	Details
	Properties Filter, Nillable, Sort Description The timestamp when the current user last accessed this record, a record related to this record, or a list view.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last viewed this record or list view. If this value is null, the user might have only accessed this record or list view (<code>LastReferencedDate</code>) but not viewed it.
MemberStatusesNotToPropagate	Type multipicklist Properties Create, Defaulted on create, Filter, Nillable, Update Description The bundle member statuses that aren't overridden when the bundle's status is updated. Possible values are: - Accepted - Canceled - Cannot Complete - Completed - Dispatched - In Progress - None - Rejected - Scheduled The default value is None.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The name of the Appointment Bundle Config.

Field	Details
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description ID of the owner of this object. This is a polymorphic relationship field. Relationship Name Owner Relationship Type Lookup Refers To Group, User
RemoveFromBundleStatuses	Type multipicklist Properties Create, Defaulted on create, Filter, Nillable, Update Description The statuses of service appointments that are allowed to be removed from a bundle. Possible values are: - Accepted - Canceled - Cannot Complete - Completed - Dispatched - In Progress - None - Rejected - Scheduled The default value is None.
StatusOnRemovalFromBundle	Type picklist Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The status that a service appointment is given when it's removed from a bundle. Possible values are:

Field	Details
	- Accepted - Canceled - Cannot Complete - Completed - Dispatched - In Progress - None - Rejected - Scheduled The default value is None.
StatusesNotToUpdateOnUnbundle	Type multipicklist Properties Create, Defaulted on create, Filter, Nillable, Update Description The statuses that aren't updated when a bundle is unbundled. Possible values are: - Accepted - Canceled - Cannot Complete - Completed - Dispatched - In Progress - None - Rejected - Scheduled The default value is None.

ApptBundlePolicy

Policy that defines how the bundling of service appointments should be handled. This object is available in API version 54.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

- Field Service must be enabled.

- Bundling must be enabled in the Field Service Settings.
- The Field Service Admin, Field Service Bundle for Dispatcher, and Field Service Integration permission sets must be enabled.

Fields

Field	Details
BundleEndTimeFieldName	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description If IsTimeCalcByBundleDurationField is true, this field represents the name of the field used for entering the end time of the bundle.
BundleStartTimeFieldName	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description If IsTimeCalcByBundleDurationField is true, this field represents the name of the field used for entering the start time of the bundle.
CanAllowSchleDepndInBundle	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description This field is reserved for future use.
ConstantTimeValue	Type int Properties Create, Filter, Group, Nillable, Sort, Update Description If IsTimeCalcByBundleDurationField is true, this field represents the total time of the bundle as a preset constant value.
FilterCriteriaId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The active recordset filter criteria used for the bundle members. Only service appointments that meet the criteria can be bundled. This is a relationship field. Relationship Name FilterCriteria Relationship Type Lookup Refers To RecordsetFilterCriteria
IsAutomaticBundling	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates if the policy is relevant for automatic bundling.
IsManualBundling	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates if the policy is relevant for manual bundling. The default value is 'false'.
IsTimeCalcByBundleDurationFld	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates if the bundle's duration is validated. If true, the bundle's start time is subtracted from the bundle's end time. If the result is a negative value, it uses ConstantTimeValue as the bundle's duration. The default value is 'false'.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort

Field	Details
	Description The timestamp when the current user last accessed this record, a record related to this record, or a list view.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last viewed this record or list view. If this value is null, the user might have only accessed this record or list view (LastReferencedDate) but not viewed it.
LimitAmountOfBundleMembers	Type int Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description The maximum number of bundle members that can be included in a bundle.
LimitDurationOfBundle	Type int Properties Create, Filter, Group, Nillable, Sort, Update Description The maximum duration of a bundle.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description Name of the bundle policy.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description ID of the owner of this object. This is a polymorphic relationship field.

Field	Details
	Relationship Name Owner
	Relationship Type Lookup
	Refers To Group, User
Priority	Type int
	Properties Create, Filter, Group, idLookup, Sort, Update
	Description The priority level that this bundle policy should be given when the bundle policies are analyzed using the automatic mode.

ApptBundlePolicySvcTerr

Represents a link between the BundlePolicy and the ServiceTerritory. This object is available in API version 54.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

- Field Service must be enabled.
- Bundling must be enabled in the Field Service Settings.
- The Field Service Admin, Field Service Bundle for Dispatcher, and Field Service Integration permission sets must be enabled.

Fields

Field	Details
BundlePolicyId	Type reference
	Properties Create, Filter, Group, Sort
	Description The ID of the parent bundle policy. This is a relationship field.

Field	Details
	Relationship Name BundlePolicy Relationship Type Lookup Refers To ApptBundlePolicy
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last accessed this record, a record related to this record, or a list view.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last viewed this record or list view. If this value is null, the user might have only accessed this record or list view (<code>LastReferencedDate</code>) but not viewed it.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The name of the appointment bundle service territory.
ServiceTerritoryId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the service territory. This is a relationship field. Relationship Name ServiceTerritory

Field	Details
	Relationship Type Lookup
	Refers To ServiceTerritory

ApptBundlePropagatePolicy

Policy that defines which property values are inherited from the bundle to the bundle members or are assigned as constant values in the bundle members. This object is available in API version 55.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

- Field Service must be enabled.
- Bundling must be enabled in the Field Service Settings.
- The Field Service Admin, Field Service Bundle for Dispatcher, and Field Service Integration permission sets must be enabled.

Fields

Field	Details
AdditionalConstantValue	Type string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The additional constant value that is connected to the initial constant value to be added to the bundle members.
BundleFieldName	Type picklist
	Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description Name of the source field in the bundle from which the value is taken. Possible values are: All default and custom Service Appointment fields.
BundleMemberFieldName	Type picklist

Field	Details
	Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Name of the target field in the bundle member where the value is inherited from the bundle. Possible values are: All default and custom Service Appointment fields.
BundlePolicyId	Type reference Properties Create, Filter, Group, Sort Description ID of the parent bundle policy. This field is a relationship field. Relationship Name BundlePolicy Relationship Type Lookup Refers To ApptBundlePolicy
ConstantValue	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The constant value to be added to the bundle members.
DateValue	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Represents how the date value is determined. Possible values are: • End of Day • Now • Null • Start of Day

Field	Details
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last accessed this record, a record related to this record, or a list view.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last viewed this record or list view. If this value is null, the user might have only accessed this record or list view (LastReferencedDate) but not viewed it.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The name of the appointment bundle propagation policy.
ShouldAddConstantValue	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates if you want to enable adding a constant value to the bundle members.
ShouldUpdateOnAdd	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates if you want to enable updating the fields of the bundle members when they are added to the bundle.
ShouldUpdateOnRemove	Type boolean

Field	Details
	Properties Create, Defaulted on create, Filter, Group, Sort, Update
	Description Indicates if you want to enable updating the fields of the bundle members when they are removed from the bundle.
ShouldUpdateOnUnbundle	Type boolean
	Properties Create, Defaulted on create, Filter, Group, Sort, Update
	Description Indicates if you want to enable updating the fields of the bundle members when performing the Unbundle action.

ApptBundleRestrictPolicy

Policy that defines the restrictions that are considered while forming a bundle. This object is available in API version 54.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

- Field Service must be enabled.
- Bundling must be enabled in the Field Service Settings.
- The Field Service Admin, Field Service Bundle for Dispatcher, and Field Service Integration permission sets must be enabled.

Fields

Field	Details
BundlePolicyId	Type reference
	Properties Create, Filter, Group, Sort
	Description ID of the parent bundle policy. This is a relationship field.
	Relationship Name BundlePolicy

Field	Details
	Relationship Type Lookup Refers To ApptBundlePolicy
DoesAllowEmpty	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Allows a bundle member service appointment with an empty Restriction Field Name to be bundled.
DoesRestrictAutomaticMode	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates if you want to apply this restriction when using the automatic mode.
DoesRestrictManualMode	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates if you want to apply this restriction when using the manual mode.
IsRestrictByDateOnly	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates if you want the bundle to be restricted according to the calendar date only, ignoring the time of day.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last accessed this record, a record related to this record, or a list view.

Field	Details
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last viewed this record or list view. If this value is null, the user might have only accessed this record or list view (<code>LastReferencedDate</code>) but not viewed it.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The name of the appointment bundle restriction policy.
RestrictionFieldName	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Name of the field in the service appointment used for applying the restriction. Possible values are: All default and custom Service Appointment fields.

ApptBundleSortPolicy

Policy that defines the properties by which the bundle members are sorted within the bundle. Can also be used in the automatic mode for determining the order of the automatic selection of bundle members. This object is available in API version 54.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

- Field Service must be enabled.
- Bundling must be enabled in the Field Service Settings.
- The Field Service Admin, Field Service Bundle for Dispatcher, and Field Service Integration permission sets must be enabled.

Fields

Field	Details
BundlePolicyId	Type reference Properties Create, Filter, Group, Sort Description The ID of the parent bundle policy. This is a relationship field. Relationship Name BundlePolicy Relationship Type Lookup Refers To ApptBundlePolicy
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last accessed this record, a record related to this record, or a list view.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last viewed this record or list view. If this value is null, the user might have only accessed this record or list view (<code>LastReferencedDate</code>) but not viewed it.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description Name of the appointment bundle sort policy.
SortDirection	Type picklist

Field	Details
	Properties Create, Filter, Group, Restricted picklist, Sort, Update Description The order of the appointments in a bundle Possible values are: - Ascending - Descending
SortFieldName	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Name of the field in the service appointment used for sorting the bundle members. Possible values are: All default and custom Service Appointment fields.
SortOrder	Type int Properties Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update Description The order of fields used for sorting the bundle members.
SortType	Type picklist Properties Create, Filter, Group, Restricted picklist, Sort, Update Description The applied sort type for arranging the bundle. Sort for Automatic Bundling defines the order that automated bundling uses to examine the candidate service appointments to be bundled. Sort Within a Bundle defines the order of bundle members. It's also used when you unbundle to define the order that the service appointments are scheduled on the Gantt. Possible values are: - SortForAutomaticBundling—Sort For Automatic Bundling - SortWithinABundle—Sort Within a Bundle

AppExtension

Represents a connection between the Field Service mobile app and another app, typically for passing record data to the Salesforce mobile app or other apps. This object is available in API version 41.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field Name	Details
<code>AppExtensionLabel</code>	Type string Properties Create, Filter, Group, Sort, Update Description The label in the UI for the app extension.
<code>AppExtensionName</code>	Type string Properties Create, Filter, Group, Sort, Update Description The API name of the app extension.
<code>FieldServiceMobileSettingsId</code>	Type reference Properties Create, Filter, Group, Sort Description The ID of a set of field service mobile settings.
<code>InstallationUrl</code>	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The URL that takes the user to the app install location, such as the App Store or Google Play.
<code>LaunchValue</code>	Type string Properties Create, Filter, Group, Sort, Update

Field Name	Details
	Description A value directing the Field Service app to the appropriate app extension. The Launch Value can be a static URL or a dynamic value that you can represent with certain tokens. These tokens pass field information from the record that the user is currently viewing. The basic format for these tokens is based on the field names; for example: {!\$Name}.
ScopedToObjectTypes	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description Indicates the types of records from which the app extension can be activated. Scoping an app extension to an object lets users activate the app extension from records of the specified type. For example, to scope to both work orders and service appointments you would use the value <code>WorkOrder, ServiceAppointment</code>.
Type	Type picklist Properties Create, Filter, Group, Restricted picklist, Sort, Update Description A picklist of types of app extensions: iOS, Android, Flow, and Lightning Apps

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

AppExtensionChangeEvent

Change events are available for the object. Available in API version 55.0 and later.

Asset

Represents an item of commercial value, such as a product sold by your company or a competitor, that a customer has purchased.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
AccountId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description (Required) ID of the Account associated with this asset. Must be a valid account ID. Required if <code>ContactId</code> isn't specified. This field is a relationship field. Relationship Name Account Relationship Type Lookup Refers To Account
Address	Type address Properties Filter, Nillable Description Represents the physical address or geolocation of the asset.
AssetLevel	Type int Properties Filter, Group, Nillable, Sort Description The asset's position in an asset hierarchy. If the asset has no parent or child assets, its level is 1. Assets that belong to a hierarchy have a level of 1 for the root asset, 2 for the child assets of the root asset, 3 for their children, and so forth. On assets created before the introduction of this field, the asset level defaults to -1. After the asset record is updated, the asset level is calculated and automatically updated.
AssetProvidedById	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The account that provided the asset, typically a manufacturer. This field is a relationship field.

Field	Details
	Relationship Name AssetProvidedBy Relationship Type Lookup Refers To Account
AssetServicedById	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The account in charge of servicing the asset. This field is a relationship field. Relationship Name AssetServicedBy Relationship Type Lookup Refers To Account
AssetTypeId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The asset type associated with the asset. This field is a relationship field. This field is available in API version 62.0 and later for users with the Health Cloud Appointment Management permission set. Relationship Name AssetType Relationship Type Lookup Refers To AssetType
Availability	Type percent Properties Filter, Nillable, Sort

Field	Details
	Description The percentage of expected uptime where the asset was available for use.
Averagetimetorepair	Type double Properties Create, Filter, Nillable, Sort, Update Description Represents the number of hours it typically takes to repair an asset after a failure.
Averagetimebetweenfailure	Type double Properties Create, Filter, Nillable, Sort, Update Description Represents the number of hours that typically elapses before the asset is likely to fail again.
AverageUptimePerDay	Type double Properties Create, Filter, Nillable, Sort, Update Description The average number of hours per day the asset is expected to be available for use.
City	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The city detail for the address.
ConsequenceOfFailure	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update Description The business impact associated with the asset's failure. Using this field, you can address the asset's health and take action using Flows . To enable this field, use Object Manager to update the field availability. Make sure that the field is visible for field-level security and for page layout. To learn more, see What Determines Field Access . The picklist values aren't predefined in orgs created before Winter '22 that aren't Field Service enabled. This field is available in API version 53.0 and later.

Field	Details
	Possible values are: • Insignificant • Minor • Moderate • Major • Critical
ContactId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description Required if AccountId isn't specified. ID of the Contact associated with this asset. Must be a valid contact ID that has an account parent (but doesn't need to match the asset's AccountId). This field is a relationship field. Relationship Name Contact Relationship Type Lookup Refers To Contact
Country	Type String Properties Create, Filter, Group, Nillable, Sort, Update Description The country detail for the address.
CurrencyIsoCode	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Three-letter ISO 4217 currency code associated with the invoice. The default value is USD. This field is available in API version 55.0 and later. This field is available when CPQ Plus, Salesforce Billing, or Revenue Cloud is enabled.
CurrentAmount	Type currency

Field	Details
	Properties Filter, Nillable, Sort Description Reserved for future use. This field is available in API version 50.0 and later. This field is available when CPQ Plus, Salesforce Billing, or Revenue Cloud is enabled.
CurrentLifecycleEndDate	Type dateTime Properties Filter, Nillable, Sort Description Represents the end of the period shown as current. System-populated field inherited from the end date of the current asset state period. If that field is empty, as with an evergreen subscription, the Current Lifecycle End Date field is also empty. This field is available in API version 50.0 and later. This field is available when CPQ Plus, Salesforce Billing, or Revenue Cloud is enabled.
CurrentMrr	Type currency Properties Filter, Nillable, Sort Description The asset's monthly recurring revenue during the current asset state period. System-populated field inherited from the monthly recurring revenue on the current asset state period. If no asset state period is current, the value is 0. Label is Current Monthly Recurring Revenue. This field is available in API version 50.0 and later. This field is available when CPQ Plus, Salesforce Billing, or Revenue Cloud is enabled.
CurrentQuantity	Type double Properties Filter, Nillable, Sort Description The asset's quantity during the current asset state period. System-populated field inherited from the quantity on the current asset state period. If no asset state period is current, the value is 0. This field is available in API version 50.0 and later. This field is available when CPQ Plus, Salesforce Billing, or Revenue Cloud is enabled.
Description	Type textarea

Field	Details
	Properties Create, Nillable, Update Description Description of the asset.
DigitalAssetStatus	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update Description Status of digital tracking of the asset. The default picklist includes the following values: - On - Off - Warning - Error
ExternalIdentifier	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the matching record in an external system. This field is available in API version 49.0 and later.
GeocodeAccuracy	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Accuracy level of the geocode for the address.
HasLifecycleManagement	Type boolean Properties Defaulted on create, Filter, Group, Sort Description True if this asset is a lifecycle-managed asset, otherwise false. You can't switch an asset to a lifecycle-managed asset or the reverse. This field is system populated. The default value is <i>false</i>. This field is available in API version 50.0 and later. This field is available when CPQ Plus, Salesforce Billing, or Revenue Cloud is enabled.

Field	Details
<code>InstallDate</code>	Type date Properties Create, Filter, Group, Nillable, Sort, Update Description Date when the asset was installed.
<code>IsCompetitorProduct</code>	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether this Asset represents a product sold by a competitor (<code>true</code>) or not (<code>false</code>). The default value is <code>false</code>. Its UI label is Competitor Asset.
<code>IsInternal</code>	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates that the asset is produced or used internally (<code>true</code>) or not (<code>false</code>). The default value is <code>false</code>. Its UI label is Internal Asset.
<code>LastReferencedDate</code>	Type dateTime Properties Filter, Nillable, Sort Description The date and time that the asset was last modified. Its UI label is Last Modified Date.
<code>LastViewedDate</code>	Type dateTime Properties Filter, Nillable, Sort Description The date and time that the asset was last viewed.
<code>Latitude</code>	Type double Properties Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description Used with Longitude to specify the precise geolocation of the address.
LifecycleEndDate	Type dateTime Properties Filter, Nillable, Sort Description Represents the end of the asset's lifecycle. System-populated field inherited from the end date of the final asset state period. If that field is empty, as with an evergreen subscription, the lifecycle has no end date. This field is available in API version 50.0 and later. This field is available when CPQ Plus, Salesforce Billing, or Revenue Cloud is enabled.
LifecycleStartDate	Type dateTime Properties Filter, Nillable, Sort Description Represents the beginning of the asset's lifecycle. System-populated field inherited from the start date of the earliest asset state period. This field can't be edited. When a new asset action affects the start date of an asset state period, the period is deleted and a new one is generated. This field is available in API version 50.0 and later. This field is available when CPQ Plus, Salesforce Billing, or Revenue Cloud is enabled.
LocationId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The asset's location. Typically, this location is the place where the asset is stored, such as a warehouse or van. If you have access to the location entity, it doesn't necessarily mean you can access the location id field. To access the location, you must have <code>userHasLocation</code> user access.
Longitude	Type double Properties Create, Filter, Group, Nillable, Sort, Update Description Used with Latitude to specify the precise geolocation of the address.
ManufactureDate	Type date

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update Description The date when the asset was manufactured. This field is available from API version 49.0 and later.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description (Required) Name of the asset. Label is Asset Name.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The asset's owner. By default, the asset owner is the user who created the asset record. Its UI label is Asset Owner. This field is a relationship field. Relationship Name Owner Relationship Type Lookup Refers To User
ParentId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The asset's parent asset. Its UI label is Parent Asset. This field is a relationship field. Relationship Name Parent Relationship Type Lookup Refers To Asset

Field	Details
PostalCode	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The postal code for the address.
Price	Type currency Properties Create, Filter, Nillable, Sort, Update Description Price paid for this asset.
PricingSource	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Pricing source to use when amending or renewing an asset. Valid values are: • LastTransaction—Last Transaction • PriceBookListPrice—Price Book or List Price Available in API version 60.0 and later.
Product2Id	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description (Optional) ID of the Product2 associated with this asset. Must be a valid Product2 ID. Its UI label is Product. This field is a relationship field. Relationship Name Product2 Relationship Type Lookup Refers To Product2

Field	Details
ProductCode	Type string Properties Filter, Group, Nillable, Sort Description The product code of the related product.
ProductDescription	Type string Properties Filter, Sort, Nillable Description The product description of the related product.
ProductFamily	Type picklist Properties Filter, Group, Sort, Nillable Description The product family of the related product.
PurchaseDate	Type date Properties Create, Filter, Group, Nillable, Sort, Update Description Date on which this asset was purchased.
Quantity	Type double Properties Create, Filter, Nillable, Sort, Update Description Quantity purchased or installed. The Quantity field value isn't set by Customer Asset Lifecycle Management. Instead, you can populate the field as you need.
QuantityIncreasePricingType	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details
	Description Specify which pricing type to use when the quantity of this asset is increased. Its UI label is Pricing Type for Quantity Increase. This field is available in API version 56.0 and later. This field is available when Revenue Cloud is enabled. Possible values are: • <code>LastNegotiatedPrice</code>—Available in API version 58.0 and later. • <code>ListPrice</code>
<code>RecordTypeId</code>	Type reference Properties Create, Filter, Group, Sort, Update Description The unique identifier for the asset. This field is a relationship field. Relationship Name RecordType Relationship Type Lookup Refers To RecordType
<code>Reliability</code>	Type percent Properties Filter, Nillable, Sort Description The percentage of expected uptime where the asset wasn't subject to unplanned downtime.
<code>RenewalPricingType</code>	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The price used when renewing a subscription. Its UI label is Pricing Type for Renewal. This field is available in API version 55.0 and later. This field is available when Revenue Cloud is enabled. Possible values are: • <code>LastNegotiatedPrice</code> • <code>ListPrice</code>

Field	Details
RenewalTerm	Type double Properties Create, Filter, Nillable, Sort, Update Description With Renewal Term Unit, defines the default subscription term for renewal quotes. This field is available in API version 55.0 and later. This field is available when Revenue Cloud is enabled.
RenewalTermUnit	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The unit of time for a subscription term. This field is available in API version 55.0 and later. This field is available when Revenue Cloud is enabled. Possible values are: • <code>Annual</code>—Available in API version 58.0 and later. —UI label is <code>Years</code>. • <code>Months</code>
RootAssetId	Type reference Properties Filter, Group, Nillable, Sort Description (Read only) The top-level asset in an asset hierarchy. Depending on where an asset lies in the hierarchy, its root could be the same as its parent. Its UI label is Root Asset. This field is a relationship field. Relationship Name RootAsset Relationship Type Lookup Refers To Asset
SalesStoreId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description ID of the RetailStore or WebStore associated with this Asset. This field is a polymorphic relationship field.

Field	Details
	To access this field, your org must have a Salesforce Order Management license or a B2B Commerce License. This field is available in API v60.0 and later. Relationship Name SalesStore Relationship Type Lookup Refers To RetailStore, WebStore
SerialNumber	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description Serial number for this asset.
State	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The state detail for the address.
Status	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update Description Customizable picklist of values. The default picklist includes the following values: • Purchased • Shipped • Installed • Registered • Obsolete
StatusReason	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The explanation of the device status. This field is available from API version 49.0 and later. Possible values are: • Not Ready • Off • Offline • Online • Paused • Standby
StockKeepingUnit	Type string Properties Filter, Group, Nillable, Sort Description The SKU assigned to the related product.
Street	Type textarea Properties Create, Filter, Group, Nillable, Sort, Update Description The street detail for the address.
SumDowntime	Type double Properties Filter, Nillable, Sort Description Accumulated downtime (planned and unplanned), determined as follows: • When only <code>UptimeRecordStart</code> is set, the sum of all downtime from <code>UptimeRecordStart</code> • When <code>UptimeRecordStart</code> and <code>UptimeRecordEnd</code> are set, the sum of all downtime from <code>UptimeRecordStart</code> to <code>UptimeRecordEnd</code> Otherwise, downtime isn't accumulated.
SumUnplannedDowntime	Type double Properties Filter, Nillable, Sort

Field	Details
	Description Accumulated unplanned downtime, determined as follows: - When only <code>UptimeRecordStart</code> is set, the sum of all unplanned downtime from <code>UptimeRecordStart</code> - When <code>UptimeRecordStart</code> and <code>UptimeRecordEnd</code> are set, the sum of all unplanned downtime from <code>UptimeRecordStart</code> to <code>UptimeRecordEnd</code> Otherwise, unplanned downtime isn't accumulated.
<code>TotalLifecycleAmount</code>	Type currency Properties Filter, Nillable, Sort Description The total amount of revenue for the asset, including revenue from each stage in the asset lifecycle. This field is available when CPQ Plus, Salesforce Billing, or Revenue Cloud is enabled.
<code>UptimeRecordEnd</code>	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description The date until which <code>SumDowntime</code> and <code>SumUnplannedDowntime</code> are accumulated.
<code>UptimeRecordStart</code>	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description The date from which <code>SumDowntime</code> and <code>SumUnplannedDowntime</code> are accumulated.
<code>UsageEndDate</code>	Type date Properties Create, Filter, Group, Nillable, Sort, Update Description Date when usage for this asset ends or expires.
<code>Uuid</code>	Type string Properties Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The unique ID for the asset. This field is available in API version 49.0 and later.

Usage

Use this object to track products sold to customers. With asset tracking, a client application can quickly determine which products were previously sold or are currently installed at a specific account. You can also create hierarchies of up to 10,000 assets.

For example, suppose that your company wants to renew and upsell opportunities on products sold in the past. Similarly, your company can track competitive products in a customer environment where products can be replaced or swapped out.

Asset tracking is also useful for product support, providing detailed information to assist with product-specific support issues. For example, the `PurchaseDate` or `SerialNumber` can indicate whether a given product has certain maintenance requirements, including product recalls. Similarly, the `UsageEndDate` can indicate when the asset was removed from service or when a license or warranty expires.

If an application creates an `Asset` record, it must specify a `Name` and either an `AccountId`, `ContactId`, or both.

With REST API, use the `getRelatedListInfo` function to get information about related lists on the asset. Note that when requesting information about `PrimaryAssets`, the response is labeled `Related Assets`, and the response for `RelatedAssets` is labeled `Primary Assets`.

Associated Objects

This object has the following associated objects. If the API version isn't specified, those objects are available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

AssetChangeEvent (API version 44.0)

Change events are available for the object.

AssetFeed

Feed tracking is available for the object.

AssetHistory

History is available for tracked fields of the object.

AssetOwnerSharingRule

Sharing rules are available for the object.

AssetShare

Sharing is available for the object.

AssetAccountParticipant

Represents a junction between the `Asset` and `Account` objects describing the association between a participating account and an asset. This object is available in API version 56.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
AccountId	Type reference Properties Create, Filter, Group, Sort, Update Description The stakeholder account associated with the asset. This field is a relationship field. Relationship Name Account Relationship Type Lookup Refers To Account
AssetId	Type reference Properties Create, Filter, Group, Sort Description The asset associated with the account. This field is a relationship field. Relationship Name Asset Relationship Type Lookup Refers To Asset
EffectiveEndDate	Type date Properties Create, Filter, Group, Nillable, Sort, Update Description The date when the association between the stakeholder and the vehicle ended.
EffectiveStartDate	Type date Properties Create, Filter, Group, Sort, Update

Field	Details
	Description The date when the association between the stakeholder and the vehicle was initiated.
IsActive	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether the association between the stakeholder and the vehicle is active (true) or not (false). The default value is <code>false</code> .
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The most recent date on which a user referenced this record.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The most recent date on which a user viewed this record.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The name of the asset account participant.
RecordTypeId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The record type associated with the asset account participant. This field is a relationship field.

Field	Details
	Relationship Name RecordType Relationship Type Lookup Refers To RecordType
StakeholderRole	Type picklist Properties Create, Filter, Group, Restricted picklist, Sort, Update Description Specifies the role of the associated account. Possible values are: • Customer-Preferred Dealer • Nominated Dealer • Closest Dealer • Sales Dealer • Service Dealer • Customer • Driver • Financier • Owner
UsageType	Type picklist Properties Create, Filter, Group, Restricted picklist, Sort, Update Description Specifies the usage type of the asset account participant. Possible values are: • Automotive • Manufacturing • FieldServiceLightning—Field Service Lightning
VehicleId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The vehicle that's marked as an asset. This field is a relationship field.
	Relationship Name Vehicle
	Relationship Type Lookup
	Refers To Vehicle

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

[AssetAccountParticipantChangeEvent](#)

Change events are available for the object.

[AssetAccountParticipantFeed](#)

Feed tracking is available for the object.

[AssetAccountParticipantHistory](#)

History is available for tracked fields of the object.

AssetAttribute

Stores asset attributes to track and analyze asset conditions to improve their uptime. This object is available in API version 57.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `query()`, `update()`

Special Access Rules

Field Service must be enabled.

Fields

Field	Details
<code>AssetId</code>	Type reference
	Properties Create, Filter, Group, Sort
	Description The ID of the asset.

Field	Details
	This field is a relationship field. Relationship Name Asset Relationship Type Lookup Refers To Asset
AttributeDefinitionId	Type reference Properties Create, Filter, Group, Sort, Update Description The ID of the attribute definition for this asset attribute. This field is a relationship field. Relationship Name AttributeDefinition Relationship Type Lookup Refers To AttributeDefinition
AttributeName	Type string Properties Filter, Group, Sort Description The name given to the asset attribute in the UI by the user.
AttributePicklistValueId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the attribute picklist value if the attribute is a picklist type. This field is a relationship field. Relationship Name AttributePicklistValue Relationship Type Lookup

Field	Details
	Refers To AttributePicklistValue
AttributeValue	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description Stores the value of an asset attribute, for example 5-TB storage .
ExternalId	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description An auto-generated ID of the attribute record saved in an external system (for example an HBase database). This field is reserved and used for internal purpose.

Usage

Add asset descriptors to the AssetAttribute object instead of creating multiple custom attributes on an asset. This helps scale to a high asset volume in the system.

AssetContactParticipant

Represents a junction between the Asset and Contact objects describing the association between a participating contact and an asset. This object is available in API version 56.0 and later.

Supported Calls

```
create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(),
retrieve(), search(), undelete(), update(), upsert()
```

Fields

Field	Details
AssetId	Type reference Properties Create, Filter, Group, Sort Description The asset associated with the contact.

Field	Details
	This field is a relationship field. Relationship Name Asset Relationship Type Lookup Refers To Asset
ContactId	Type reference Properties Create, Filter, Group, Sort, Update Description The contact associated with the asset. This field is a relationship field. Relationship Name Contact Relationship Type Lookup Refers To Contact
EffectiveEndDate	Type date Properties Create, Filter, Group, Nillable, Sort, Update Description The date when the association between the stakeholder and the vehicle ended.
EffectiveStartDate	Type date Properties Create, Filter, Group, Sort, Update Description The date when the association between the stakeholder and the vehicle was initiated.
IsActive	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update

Field	Details
	Description Indicates whether the association between the stakeholder and the vehicle is active (true) or not (false). The default value is <code>false</code>.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The most recent date on which a user referenced this record.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The most recent date on which a user viewed this record.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The name of the asset contact participant.
StakeholderRole	Type picklist Properties Create, Filter, Group, Sort, Update Description Specifies the role of the associated contact.
UsageType	Type picklist Properties Create, Filter, Group, Restricted picklist, Sort, Update Description Specifies the usage type of the asset contact participant. Possible values are: Automotive

Field	Details
	• Manufacturing • <code>FieldServiceLightning</code>—Field Service Lightning
<code>VehicleId</code>	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The vehicle that's marked as an asset. This field is a relationship field. Relationship Name Vehicle Relationship Type Lookup Refers To Vehicle

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

[AssetContactParticipantChangeEvent](#)

Change events are available for the object.

[AssetContactParticipantFeed](#)

Feed tracking is available for the object.

[AssetContactParticipantHistory](#)

History is available for tracked fields of the object.

AssetDowntimePeriod

Represents a period during which an asset is not able to perform as expected. Downtime periods include planned activities, such as maintenance, and unplanned events, such as mechanical breakdown. This object is available in API version 49.0 and later.

Supported Calls

```
create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(),
retrieve(), search(), undelete(), update(), upsert()
```

Fields

Field	Details
AssetDowntimePeriodNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description The unique number of this asset downtime period record.
AssetId	Type reference Properties Create, Filter, Group, Sort Description The ID of the asset this asset downtime period record is for.
Description	Type textarea Properties Create, Nillable, Update Description The description of this asset downtime period.
DowntimeType	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The type of this asset downtime period. Possible values are: - Planned - Unplanned
EndTime	Type dateTime Properties Create, Filter, Sort, Update Description The time this asset downtime period ended.
IsExcluded	Type boolean

Field	Details
	Properties Defaulted on create, Filter, Group, Sort Description Whether this asset downtime period is excluded from the calculation of accumulated downtime and accumulated unplanned downtime, and therefore not included in availability and reliability calculations.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp for when the current user last viewed a record related to this record.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (LastReferencedDate) and not viewed.
StartTime	Type dateTime Properties Create, Filter, Sort, Update Description The time this asset downtime period started.

AssetRelationship

Represents a non-hierarchical relationship between assets due to an asset modification; for example, a replacement, upgrade, or other circumstance. In Revenue Lifecycle Management, this object represents an asset or assets grouped in a bundle or set. This object is available in API version 41.0 and later.

Asset relationships appear in the Primary Assets and Related Assets related lists on asset records in the UI.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Some fields are available only in Revenue Cloud. Field availability is noted in the field detail column.

Fields

Field Name	Details
AssetId	Type reference Properties Create, Filter, Group, Sort Description The unique identifier of the new asset, which is the asset that is taking the place of the existing asset. This field is a relationship field. Relationship Name Asset Relationship Type Lookup Refers To Asset
AssetRelationshipNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description An auto-generated number identifying the asset relationship.
AssetRole	Type picklist Properties Filter, Group, Nillable, Restricted picklist, Sort Description Describes the position of the main asset relative to the other assets in the relationship. This field is available in API version 58.0 and later. This field is available in orgs with Revenue Cloud. Possible values are: • <code>Add-on</code>—The main asset is an add-on. • <code>Bundle</code>—The main asset is the bundle parent. • <code>Set</code>—The asset is the main asset in the set.

Field Name	Details
CurrencyIsoCode	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Three-letter ISO 4217 currency code associated with the asset. The default value is USD.
FromDate	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description The date when the new asset was installed.
GroupingKey	Type string Properties Filter, Group, idLookup, Nillable, Sort Description Read-only field used to indicate the bundle that an asset belongs to. For example, if two assets have the same GroupingKey value, then it means that the assets are bundled together. This field is available in API v.60.0 and later. This field is available in orgs with Revenue Cloud.
ProductRelationshipTypeId	Type reference Properties Filter, Group, Nillable, Sort Description The unique identifier of the record that describes the relationship between the main and associated assets. This field is available in API version 58.0 and later. This field is available in orgs with Revenue Cloud. This field is a relationship field. Relationship Name ProductRelationshipType Relationship Type Lookup

Field Name	Details
	Refers To ProductRelationshipType
ProductRelatedComponent	Type reference Properties Filter, Group, Nillable, Sort Description The product related component that's associated with the asset relationship. This field is a relationship field. This field is available in API 60.0 and later in Revenue Cloud. Relationship Name ProductRelatedComponent Relationship Type Lookup Refers To ProductRelatedComponent
RelatedAssetId	Type reference Properties Create, Filter, Group, Sort, Update Description The existing asset that is being modified. This field is a relationship field. Relationship Name RelatedAsset Relationship Type Lookup Refers To Asset
RelatedAssetPricing	Type picklist Properties Filter, Group, Nillable, Restricted picklist, Sort Description Specifies whether the price of the related asset is included in the bundle price. Valid values are: IncludedInBundlePrice

Field Name	Details
	NotIncludedInBundlePrice This field is available in API version 59.0 and later in Revenue Cloud.
RelatedAssetQtyScaleMethod	Type picklist Properties Filter, Group, Nillable, Restricted picklist, Sort Description Specifies how the quantity of the related asset changes relative to the quantity of the parent asset. Valid values are: - Constant - Proportional This field is available in API version 59.0 and later in Revenue Cloud.
RelatedAssetRole	Type picklist Properties Filter, Group, Nillable, Restricted picklist, Sort Description Describes the position of the associated asset relative to other assets in the relationship. This field is available in API version 58.0 and later. This field is available in orgs with Revenue Cloud. Valid values are: - Add-on—The main asset is an add-on. - Bundle—The main asset is the bundle parent. - Set—The asset is the main asset in the set. - Simple—The asset is purchased individually and isn't associated with variations. - Variation Parent—The main asset is the variation parent.
RelationshipType	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description The type of relationship between the existing asset and the new asset. This field comes with three values—Replacement, Upgrade, and Crossgrade—, but you can create more values in Setup. Possible values are:

Field Name	Details
	• Crossgrade—The new asset is a crossgrade of an existing asset. For example, changing a subscription to a plan with the same service, but that runs for a longer amount of time. • Replacement—The new asset is replacing an existing asset. For example, a customer's faulty widget that was under warranty is being replaced with a new one. • Upgrade—The new asset is an upgrade of an existing asset. For example, upgrading a customer's existing subscription plan to a new plan with more services. The default value is <code>Replacement</code>.
ToDate	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description The date when the modified asset is uninstalled.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

AssetRelationshipChangeEvent (API version 62.0)

Change events are available for the object.

AssetRelationshipFeed

Feed tracking is available for the object.

AssetRelationshipHistory

History is available for tracked fields of the object.

AssetRelationshipOwnerSharingRule (API version 58.0)

Sharing rules are available for the object.

AssetRelationshipShare (API version 58.0)

Sharing is available for the object.

AssetWarranty

Defines the warranty terms applicable to an asset along with any exclusions and extensions. This object is available in API version 50.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
AssetId	Type reference Properties Create, Filter, Group, Sort Description The ID of the asset this warranty term applies to.
AssetWarrantyNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description The identifier of the asset warranty record.
EndDate	Type date Properties Create, Filter, Group, Nillable, Sort, Update Description The date on which this warranty term expires.
ExchangeType	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update Description The type of exchange offered by this warranty term.
Exclusions	Type textarea Properties Create, Nillable, Update Description Description of any exclusions.
ExpensesCovered	Type percent Properties Create, Filter, Nillable, Sort, Update

Field	Details
	Description The percentage of expenses covered.
ExpensesCoveredEndDate	Type date Properties Create, Filter, Group, Nillable, Sort, Update Description The date on which cover for expenses ends.
IsTransferable	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Defines whether the warranty term can be transferred to a new owner.
LaborCovered	Type percent Properties Create, Filter, Nillable, Sort, Update Description The percentage of labor covered.
LaborCoveredEndDate	Type date Properties Create, Filter, Group, Nillable, Sort, Update Description The date on which cover for labor ends.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the asset warranty term was last modified. Its label in the user interface is <code>Last Modified Date</code> .
LastViewedDate	Type dateTime

Field	Details
	Properties Filter, Nillable, Sort Description The date when the asset warranty term was last viewed.
PartsCovered	Type percent Properties Create, Filter, Nillable, Sort, Update Description The percentage of parts covered.
PartsCoveredEndDate	Type date Properties Create, Filter, Group, Nillable, Sort, Update Description The date on which cover for parts ends.
Pricebook2Id	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the price book item associated with this asset warranty term.
StartDate	Type date Properties Create, Filter, Group, Sort, Update Description The date on which cover under this warranty term starts.
WarrantyTermId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the warranty term this asset warranty term extends.
WarrantyType	Type picklist

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The type of the warranty.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

[AssetWarrantyChangeEvent](#)

Change events are available for the object.

AssignedResource

Represents a service resource who is assigned to a service appointment in Field Service and Lightning Scheduler. Assigned resources appear in the Assigned Resources related list on service appointments. This object is available in API version 38.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field Name	Details
ActualTravelTime	Type double
	Properties Create, Filter, Nillable, Sort, Update
	Description The number of minutes that the service resource needs to travel to the assigned service appointment. You can enter a value with up to two decimal places.
ApptAssistantInfoUrl	Type textarea
	Properties Create, Nillable, Update
	Description The URL that contains the status of the mobile worker approaching the service appointment, the Community URL, and the expiry of the URL. Available in version 51.0 and later.

Field Name	Details
AssignedResourceNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description An auto-generated number identifying the resource assignment.
EstimatedTravelTime	Type double Properties Create, Filter, Nillable, Sort, Update Description The estimated number of minutes needed for the service resource to travel to the service appointment they're assigned to. You can enter a value with up to two decimal places.
LocationStatus	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The status of the mobile worker approaching the service appointment. When the location status changes to one of these values, a status update containing <code>ApptAssistantInfoUrl</code> is sent to the customer. Available in version 51.0 and later. Possible values are: • EnRoute • LastMile
IsPrimaryResource	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether the service resource is a primary resource or not. The default value is false. Available in API version 47.0 and later.
ServiceAppointmentId	Type reference Properties Create, Filter, Group, Sort

Field Name	Details
	Description The service appointment that the resource is assigned to. This is a relationship field. Relationship Name ServiceAppointment Relationship Type Lookup Refers To ServiceAppointment
ServiceCrewId	Type reference Properties Create, Update, Filter, Group, Sort, Nillable Description The service crew that the resource is assigned to.  Note: Since service resources can represent crews or individuals, appointments are typically assigned to crews in the following way: 1. Create a service resource of the Crew type that represent the crew. 2. Create an assigned resource on the service appointment and select the crew resource in the <code>ServiceResourceId</code> field. As an alternative, you can assign appointments to crew members separately. This lets you track each member's travel time and see a list of the crew members in the Assigned Resources related list. To take this approach, create an assigned resource for each crew member. List the crew member in the <code>ServiceResourceId</code> field and the crew they belong to in the <code>ServiceCrewId</code> field.
ServiceResourceId	Type reference Properties Create, Update, Filter, Group, Sort Description The resource who is assigned to the service appointment. This is a relationship field. Relationship Name ServiceResource Relationship Type Lookup Refers To ServiceResource

Field Name	Details
Transaction	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The last transaction ID of the scheduling and optimization request that updated this object. The transaction ID is automatically generated and populated by the Enhanced Scheduling and Optimization engine. Available in API version 63.0 and later.

Usage

You can assign multiple service resources to a service appointment. Service resources who are assigned to service appointments cannot be deactivated until they are removed from the appointments.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

AssignedResourceChangeEvent (API version 48.0)

Change events are available for the object.

AssignedResourceHistory (API version 61.0)

History is available for tracked fields of the object.

AssignedResourceFeed

Feed tracking is available for the object.

Associated Location

Represents a link between an account and a location in Field Service. You can associate multiple accounts with one location. For example, a shopping center location may have multiple customer accounts.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field Name	Details
ActiveFrom	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description Date and time the associated location is active.
ActiveTo	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description Date and time the associated location stops being active.
AssociatedLocationNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description Auto-generated number identifying the associated location.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date the associated location was last modified.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date the associated location was last viewed.
LocationId	Type reference Properties Create, Filter, Group, Sort, Update

Field Name	Details
	Description The location associated with the address. This is a relationship field. Relationship Name Location Relationship Type Lookup Refers To Location
ParentRecordId	Type reference Properties Create, Filter, Group, Sort Description The account associated with the location. This is a relationship field. Relationship Name ParentRecord Relationship Type Lookup Refers To Account
Type	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update Description Picklist of address types. The values are: • Bill To • Ship To

Associated Objects

This object has the following associated objects. Unless noted, they are available in the same API version as this object.

AssociatedLocationChangeEvent (API version 62.0)

Change events are available for the object.

AssociatedLocationHistory

History is available for tracked fields of the object.

AttributeDefinition

Represents a product, asset, or object attribute, for example, a hardware specification or software detail. This object is available in API version 57.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field	Details
DataType	Type picklist Properties Create, Filter, Group, Restricted picklist, Sort, Update Description The data type of the attribute definition. Possible values are: • Checkbox • Date • Datetime • Number • Picklist • Text
DefaultValue	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The default value for this attribute.
Description	Type textarea Properties Create, Nillable, Update Description Description of this attribute.

Field	Details
DeveloperName	Type string Properties Create, Filter, Group, Sort Description The unique name of the attribute definition record. This name must begin with a letter and use only alphanumeric characters and underscores. It can't include spaces, end with an underscore, or have two consecutive underscores. The developer name is used for internal purpose and must be unique for all records (including deleted records). If the system doesn't find the name unique, it automatically overrides the user input and creates a unique name. For external use, the developer name need not be fixed.
IsActive	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates that the attribute definition is active. Active attributes definitions can be selected for assets. The default value is <code>false</code>.
IsRequired	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether the attribute definition is required for an asset. The default value is <code>false</code>.
Label	Type string Properties Create, Filter, Group, Sort, Update Description The label for the attribute. Displays a friendly name for the attribute, for example, threshold monitor lightning component and recordset filter criteria rule.
LastReferencedDate	Type dateTime

Field	Details
	Properties Filter, Nillable, Sort Description The date the attribute definition was last referenced.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date the attribute definition was last viewed.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The name of the attribute.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The owner of the attribute definition. This field is a polymorphic relationship field. Relationship Name Owner Relationship Type Lookup Refers To Group, User
PicklistId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the attribute picklist with the valid values for this attribute. This field is a relationship field.

Field	Details
	Relationship Name Picklist Relationship Type Lookup Refers To AttributePicklist
SourceSystemIdentifier	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The identifier of the attribute definition in an external system.
UnitOfMeasureId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the measurement unit for this attribute. This field is a relationship field. Relationship Name UnitOfMeasure Relationship Type Lookup Refers To UnitOfMeasure

Usage

Add asset descriptors to the Asset object instead of creating multiple custom attributes on an asset. This helps scale to a high volume of various assets in the system. When you create the AttributeDefinition, you can provide a unique API name. If the API name is not unique, the system appends a number to the end of the API name. The value of this number depends on how many times the same name has been used.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

AttributeDefinitionHistory

History is available for tracked fields of the object.

AttributeDefinitionOwnerSharingRule

Sharing rules are available for the object.

AttributeDefinitionShare

Sharing is available for the object.

AttributePicklist

Represents a custom picklist for an asset attribute. This object is available in API version 57.0 and later.

Supported Calls

```
create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(),
retrieve(), search(), undelete(), update(), upsert()
```

Special Access Rules

Field Service must be enabled.

Fields

Field	Details
DataType	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The data type of this picklist. Possible values are: • Boolean • Currency • Date • Datetime • Number • Percent • Text The default value is Boolean.
Description	Type textarea Properties Create, Nillable, Update

Field	Details
	Description A description of the picklist. Maximum size is 32000 alphanumeric characters. Can include the following special characters: @! - < > * ? + = % # () / \ & ' £ € \$ ".
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date the attribute picklist was last referenced.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date the attribute picklist was last viewed.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The name of the picklist. Names must be unique.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The ID of the owner of the attribute picklist record. This field is a polymorphic relationship field. Relationship Name Owner Relationship Type Lookup Refers To Group, User
Status	Type picklist

Field	Details
	Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The status of the attribute picklist. Possible values are: • Active • Draft • Inactive The default value is <code>Draft</code>.
UnitOfMeasureId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the unit of measure associated with the product. This field is a relationship field. This field is available when Revenue Cloud is enabled. This field is available in API version 63.0 and later. Relationship Name UnitOfMeasure Refers To UnitOfMeasure

Usage

The AttributePicklist object is the parent object and the AttributePicklistValue object contains the picklist values. Let's say you need an asset attribute to track the T-shirt size, which can be small, medium, or large. Create an AttributePicklist parent record as a Text type for the T-shirt size attribute. Then create AttributePicklistValue records, one for each picklist value small, medium, and large, and associate them with the parent record.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

AttributePicklistHistory

History is available for tracked fields of the object.

AttributePicklistOwnerSharingRule

Sharing rules are available for the object.

AttributePicklistShare

Sharing is available for the object.

AttributePicklistValue

Represents the values of an asset attribute picklist. This object is available in API version 57.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field	Details
Abbreviation	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description A short name of the picklist value that's displayed at run time. Use up to 255 alphanumeric characters. Can include the following special characters: @ ! - < > * ? + = % # () / \ & ' £ € \$ " .
Code	Type string Properties Create, Filter, Group, Sort, Update Description A picklist value code unique to the picklist. Maximum size is 80 alphanumeric characters. Can include the following special characters: @ ! - < > * ? + = % # () / \ & ' £ € \$ " .
DisplayValue	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The displayed picklist value if it's different from the Name field. For example, the Name '5' could have a DisplayValue 'Five'.
IsDefault	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update

Field	Details
	Description Indicates whether the picklist value is the default for the associated picklist. Only one value can be the default for a picklist. The default value is <code>false</code>.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date the attribute picklist value was last referenced.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date the attribute picklist value was last viewed.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The name of the attribute picklist value.
PicklistId	Type reference Properties Create, Filter, Group, Sort Description The ID of the picklist that the value is associated with. This field is a relationship field. Relationship Name Picklist Relationship Type Lookup Refers To AttributePicklist

Field	Details
Sequence	Type double Properties Create, Filter, Nillable, Sort, Update Description The order in which the picklist value appears in the picklist.
Status	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The status of the attribute picklist value. Possible values are: • Active • Draft • Inactive The default value is <code>Draft</code>.
Value	Type string Properties Create, Filter, Group, Sort, Update Description The text value for a picklist item if the picklist data type is text. This value must be unique within a picklist. Maximum size is 255 alphanumeric characters. Can include the following special characters: @ ! - < > * ? + = % # () / \ & ' £ € \$ % " .

Usage

The `AttributePicklistValue` object is the child object and the `AttributePicklist` object contains the picklist. Let's say you need an asset attribute to track the T-shirt size, which can be small, medium, or large. Create an `AttributePicklist` parent record as a Text type for the T-shirt size attribute. Then create `AttributePicklistValue` records, one for each picklist value small, medium, and large, and associate them with the parent record..

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

AttributePicklistValueHistory

History is available for tracked fields of the object.

ContractLineItem

Represents a product covered by a service contract (customer support agreement). This object is available in API version 18.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
AssetId	Type reference Properties Create, Filter, Nillable, Update Description Required. ID of the Asset associated with the contract line item. Must be a valid asset ID.
Description	Type textarea Properties Create, Nillable, Update Description Description of the contract line item.
Discount	Type percent Properties Create, Filter, Nillable, Update Description The discount for the product as a percentage. When updating, if you specify <code>Discount</code> without specifying <code>TotalPrice</code>, the <code>TotalPrice</code> will be adjusted to accommodate the new <code>Discount</code> value, and the <code>UnitPrice</code> will be held constant. If you specify both <code>Discount</code> and <code>Quantity</code>, you must also specify either <code>TotalPrice</code> or <code>UnitPrice</code> so the system can determine which one to automatically adjust.
EndDate	Type date Properties Create, Filter, Nillable, Update

Field	Details
	Description The last day the contract line item is in effect.
LastReferencedDate	Type date Properties Filter, Nillable, Sort, Update Description The timestamp when the current user last accessed this record, a record related to this record, or a list view.
LastViewedDate	Type date Properties Filter, Nillable, Sort, Update Description The timestamp when the current user last viewed this record or list view. If this value is null, the user might have only accessed this record or list view (<code>LastReferencedDate</code>) but not viewed it.
LineItemNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Update Description Automatically-generated number that identifies the contract line item.
ListPrice	Type currency Properties Filter, Nillable Description Corresponds to the <code>UnitPrice</code> on the <code>PricebookEntry</code> that is associated with this line item, which can be in the standard pricebook or a custom pricebook. A client application can use this information to show whether the unit price (or sales price) of the line item differs from the pricebook entry list price.
LocationId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The location associated with the contract line item. If you have access to the location entity, it doesn't necessarily mean you can access the location id field. To access the location, you must have <code>userHasLocation</code> user access.
ParentContractLineItemId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The line item's parent line item, if it has one.
PricebookEntryId	Type reference Properties Create, Filter, Update Description Required. ID of the associated PricebookEntry. Only exists if Product2 is enabled.
Product2Id	Type reference Properties Filter, Group, Nillable, Sort Description The product related to the contract line item.
Quantity	Type double Properties Create, Filter, Update Description Number of units of the contract line item (product) included in the associated service contract.
RootContractLineItemId	Type reference Properties Filter, Group, Nillable, Sort Description (Read only) The top-level line item in a contract line item hierarchy. Depending on where a line item lies in the hierarchy, its root could be the same as its parent.

Field	Details
ServiceContractId	Type reference Properties Create, Filter Description Required. ID of the ServiceContract associated with the contract line item. Must be a valid service contract ID.
StartDate	Type date Properties Create, Filter, Nillable, Update Description The first day the contract line item is in effect.
Status	Type picklist Properties Filter, Nillable Description Status of the contract line item.
Subtotal	Type currency Properties Filter, Nillable Description Contract line item's sales price multiplied by the <code>Quantity</code>.
TotalPrice	Type currency Properties Filter, Nillable Description This field is available only for backward compatibility. It represents the total price of the <code>ContractLineItem</code>. If you specify <code>Discount</code> and <code>Quantity</code>, this field or <code>UnitPrice</code> is required. This field is nillable, but you can't set both <code>TotalPrice</code> and <code>UnitPrice</code> to null in the same update request. To insert the <code>TotalPrice</code> for a contract line item via the API (given only a unit price and the quantity), calculate this field as the unit price multiplied by the quantity.

Field	Details
UnitPrice	Type currency Properties Create, Filter, Update Description The unit price for the contract line item. In the user interface, this field's value is calculated by dividing the total price of the contract line item by the quantity listed for that line item. Label is Sales Price. This field or <code>TotalPrice</code> is required. You can't specify both. If you specify <code>Discount</code> and <code>Quantity</code>, this field or <code>TotalPrice</code> is required.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ContractLineItemChangeEvent (API version 44.0)

Change events are available for the object.

ContractLineItemFeed

Feed tracking is available for the object.

ContractLineItemHistory

History is available for tracked fields of the object.

ContractLineOutcome

Represents information on a contract line outcome's captured data and other related parameters that are used when capturing data. This object is available in API version 58.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

- Field Service must be enabled.
- Entitlements must be enabled.

Fields

Field	Details
CalculationMethod	Type picklist

Field	Details
	Properties Create, Filter, Group, Restricted picklist, Sort, Update Description The method used for calculating the contract line outcome's captured data to determine the outcome value. Select Average or As Captured to calculate the contract line outcome. Average calculates the outcome value based on the average of all data captured to date. As Captured calculates the outcome value based on the asset's current data at the time of the compliance check. Possible values are: • AsCaptured • Average
CaptureFrequency	Type picklist Properties Create, Filter, Group, Restricted picklist, Sort, Update Description The frequency at which data capturing and contract compliance check for the contract line outcome occurs. Possible values are: • Daily • Monthly • Weekly
ComplianceStatus	Type picklist Properties Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort Description Indicates if the criteria were met. Compliant—The outcome is compliant with the contract. Not Compliant—The outcome isn't compliant with the contract. Not Available—The outcome's compliance information isn't available yet. Invalid—The outcome isn't valid because the option selected for the Criteria Field of the recordset filter criteria was deleted. To restart the calculation, create a new contract line outcome. Possible values are: • Compliant • Invalid • NotAvailable • NotCompliant The default value is NotAvailable.

Field	Details
ContractLineItemId	Type reference Properties Create, Filter, Group, Sort, Update Description The contract line item associated with the contract line outcome. This field is a relationship field. Relationship Name ContractLineItem Relationship Type Lookup Refers To ContractLineItem
Description	Type textarea Properties Create, Nillable, Update Description A description of the contract line outcome.
EndDate	Type dateTime Properties Create, Filter, Sort, Update Description The contract line outcome's data capture end date.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date and time when the contract line outcome was last modified. Its UI label is Last Modified Date.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort

Field	Details
	Description The date and time when the contract line outcome was last viewed.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The name of the contract line outcome.
NextDataCaptureDate	Type dateTime Properties Filter, Nillable, Sort Description The date of the next data capture and compliance check based on the capture frequency. The date is auto-populated and updated after each capture
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The contract line outcome's owner. By default, the owner is the user who created the contract line outcome record. Its UI label is Contract Line Outcome Owner. This field is a polymorphic relationship field. Relationship Name Owner Relationship Type Lookup Refers To Group, User
RecordsetFilterCriteriaId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the recordset filter criteria in which the contract line outcome's conditions are defined. This field is a relationship field.

Field	Details
	Relationship Name RecordsetFilterCriteria Relationship Type Lookup Refers To RecordsetFilterCriteria
ServiceContractId	Type reference Properties Filter, Group, Nillable, Sort Description The service contract associated with the contract line item and the contract line outcome. This field is a relationship field. Relationship Name ServiceContract Relationship Type Lookup Refers To ServiceContract
StartDate	Type dateTime Properties Create, Filter, Sort, Update Description The contract line outcome's data capture start date.

Usage

Use this object to define the data capture frequency and other related parameters that are used when capturing data in order to evaluate a service contract's compliance.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ContractLineOutcomeChangeEvent

Change events are available for the object.

ContractLineOutcomeFeed

Feed tracking is available for the object.

ContractLineOutcomeHistory

History is available for tracked fields of the object.

ContractLineOutcomeOwnerSharingRule

Sharing rules are available for the object.

ContractLineOutcomeShare

Sharing is available for the object.

SEE ALSO:

[ContractLineOutcomeData](#)

ContractLineOutcomeData

Represents the contract line outcome's captured data. It stores the data that was captured between the contract line outcome's start date and end date. This object is available in API version 58.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

- Field Service must be enabled.
- Entitlements must be enabled.

Fields

Field	Details
CalculatedValue	Type double Properties Filter, Nillable, Sort Description The value calculated based on the contract line outcome's calculation method and the captured data.
CaptureDate	Type dateTime Properties Create, Filter, Sort, Update Description The date and time when the data was captured.

Field	Details
ContractLineOutcomeId	Type reference Properties Create, Filter, Group, Sort Description The contract line outcome associated with the contract line outcome data record. This field is a relationship field. Relationship Name ContractLineOutcome Relationship Type Lookup Refers To ContractLineOutcome
KeyPerformanceIndicator	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The key performance indicators (fields or asset attributes) that define the contract line outcome's compliance status.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date and time when the contract line outcome data record was last modified. Its UI label is Last Modified Date.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date and time when the contract line outcome data record was last viewed.
Name	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort

Field	Details
	Description The name of the contract line outcome data record.
Value	Type double Properties Create, Filter, Nillable, Sort, Update Description The actual value of the key performance indicator.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ContractLineOutcomeDataChangeEvent

Change events are available for the object.

ContractLineOutcomeDataFeed

Feed tracking is available for the object.

ContractLineOutcomeDataHistory

History is available for tracked fields of the object.

ContractLineOutcomeDataOwnerSharingRule

Sharing rules are available for the object.

ContractLineOutcomeDataShare

Sharing is available for the object.

DigitalSignature

Represents a signature captured on a service report in field service.

Supported Calls

```
create(), delete(), describeObjects(), getDeleted(), getUpdated(), query(), retrieve(),
undelete()
```

Special Access Rules

Field Service must be enabled.

Fields

Field Name	Details
DigitalSignatureNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description An auto-generated number identifying the signature.
DocumentBody	Type base64 Properties Create Description The captured signature image.
DocumentContentType	Type picklist Properties Create, Filter, Group, Restricted picklist, Sort Description The data type of the captured signature. Possible values are: • audio/acc • audio/amr • audio/ogg • video/3gpp2 • video/3gpp • image/avif • text/calendar • audio/x-caf • image/webp
DocumentLength	Type int Properties Filter, Group, Nillable, Sort Description The length of the captured signature.
DocumentName	Type string

Field Name	Details
	Properties Create, Filter, Group, Sort Description The name of the captured signature image.
ParentId	Type reference Properties Create, Filter, Group, Sort Description ID of the service appointment, work order, or work order line item that the service report is generated for. This is a polymorphic relationship field. Relationship Name Parent Relationship Type Lookup Refers To AuthorizationFormConsent, Order, ServiceAppointment, WorkOrder, WorkOrderLineItem
Place	Type string Properties Create, Filter, Group, Nillable, Sort Description The place where the report was signed.
SignatureType	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Sort Description The role of the person signing the service report. Your org comes with one signature type, <code>Default</code>. A service report template can only contain one signature per type. If you plan to collect multiple signatures on service reports, create additional values for the Signature Type field. Create at least one value for every role that might need to sign a service report. For example, <code>Technician</code>, <code>Customer</code>, <code>Supervisor</code>, or <code>Supplier</code>. If some service reports will be signed by multiple people in one role—for example, all technicians present at an appointment—create numbered types: <code>Technician 1</code>, <code>Technician 2</code>, and so forth.

Field Name	Details
	 Note: You can create up to 1,000 signature types. You can't delete signature types, but you can deactivate them so they can't be used in service report templates. When you deactivate a type, it still appears on service report templates that used it, but you can't use it on new service report templates.
SignedBy	Type string Properties Create, Filter, Group, Nillable, Sort Description The name of the person signing.
SignedDate	Type dateTime Properties Create, Filter, Nillable, Sort Description The date and time of the signing.

Usage

Add signature blocks to service report templates to determine which signatures need to be gathered on reports that use the template. Service report templates can contain up to 20 signatures, and each signature must use a different Signature Type. For example, create a standard service report template that contains a customer signature and a technician signature.

To learn more about digital signatures, see [Guidelines for Using Signatures on Service Reports](#).

Associated Objects

This object has the following associated objects. Unless noted, they're available in the same API version as this object.

DigitalSignatureChangeEvent (Available in API version 57.0)

Change events are available for the object.

Entitlement

Represents the customer support an account or contact is eligible to receive. This object is available in API version 18.0 and later. Entitlements may be based on an asset, product, or service contract.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
AccountId	Type reference Properties Create, Filter, Group, Sort Description ID of the Account associated with the entitlement.
AssetId	Type reference Properties Filter, Group, Nillable, Sort Description Required. ID of the Asset associated with the entitlement. Must be a valid asset ID.
AssetWarrantyID	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The identifier of the asset warranty record. Must be a valid asset warranty ID. AssetWarranty is available only with Field Service.
BusinessHoursId	Type reference Properties Filter, Group, Nillable, Sort Description Required. ID of the BusinessHours associated with the entitlement. Must be a valid business hours ID.
CasesPerEntitlement	Type int Properties Filter, Group, Nillable, Sort Description The total number of cases the entitlement supports. This field is only available if <code>IsPerIncident</code> is <code>true</code>.
ContractLineItemId	Type reference

Field	Details
	Properties Filter, Group, Nillable, Sort Description Required. ID of the ContractLineItem associated with the entitlement. Must be a valid ID.
EndDate	Type date Properties Create, Filter, Nillable, Update Description The last day the entitlement is in effect.
IsPerIncident	Type boolean Properties Defaulted on create, Filter, Update Description Indicates whether the entitlement is limited to supporting a specific number of cases (<code>true</code>) or not (<code>false</code>).
LastReferencedDate	Type date Properties Filter, Nillable, Sort, Update Description The timestamp when the current user last accessed this record, a record related to this record, or a list view.
LastViewedDate	Type date Properties Filter, Nillable, Sort, Update Description The timestamp when the current user last viewed this record or list view. If this value is null, the user might have only accessed this record or list view (<code>LastReferencedDate</code>) but not viewed it.
LocationID	Type reference Properties Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description ID of the Location associated with the entitlement. Must be a valid location ID.
Name	Type string Properties Create, Filter, Update Description Required. Name of the entitlement.
SvcApptBookingWindowsId	Type reference Properties Create, Filter, Group, Sort, Nillable, Update Description The operating hours that the entitlement's work orders should respect. The label in the user interface is <code>Operating Hours</code> . Available only if Field Service is enabled.
RemainingCases	Type int Properties Create, Filter, Nillable, Update Description The number of cases the entitlement can support. This field decreases in value by one each time a case is created with the entitlement. This field is only available if <code>IsPerIncident</code> is selected.
RemainingWorkOrders	Type int Properties Create, Filter, Group, Nillable, Sort, Update Description The number of agreed work orders remaining to be created.
ServiceContractId	Type reference Properties Create, Filter, Nillable, Update Description Required. ID of the ServiceContract associated with the entitlement. Must be a valid ID.

Field	Details
SlaProcessId	Type reference Properties Create, Filter, Nillable, Update Description ID of the SlaProcess associated with the entitlement. This field is available in version 19.0 and later.
StartDate	Type date Properties Create, Filter, Nillable, Update Description The first date the entitlement is in effect.
Status	Type picklist Properties Filter, Nillable Description Status of the entitlement, such as <code>Expired</code>.
SvcApptBookingWindows	Type reference Properties Create, Filter, Group, Sort, Nillable, Update Description The operating hours of the entitlement. This field is visible only if Field Service is enabled.
Type	Type picklist Properties Create, Defaulted on create, Filter, Nillable, Update Description The type of entitlement, such as Web or phone support.
WorkOrdersPerEntitlement	Type int Properties Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description Total number of work orders available for this entitlement.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

EntitlementChangeEvent (API version 44.0)

Change events are available for the object.

EntitlementFeed (API version 23.0)

Feed tracking is available for the object.

EntitlementHistory

History is available for tracked fields of the object.

EntityMilestone

Represents a required step in a customer support process on a work order. The Salesforce user interface uses the term "object milestone." This object is available in API version 37.0 and later.



Note: Milestones on cases use the CaseMilestone object type.

Supported Calls

`delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`

Special Access Rules

- As of Summer '20 and later, only Salesforce admins, users with access to the Case, Entitlement, or Work Order objects, and users with the View Setup and Configuration permission can access this object.
- Entitlement management must be enabled.
- Work orders or Field Service must be enabled.

Fields

Field Name	Details
ActualElapsedTimeInDays	Type double Properties Filter, Nillable, Sort Description The number of days that it took to complete a milestone. (Elapsed Time) – (Stopped Time) = (Actual Elapsed Time)

Field Name	Details
	 Note: To display this field, select Enable stopped time and actual elapsed time on the Entitlement Settings page and add the field to the object milestone page layout.
ActualElapsedTimeInHrs	Type double Properties Filter, Nillable, Sort Description The number of hours that it took to complete a milestone. (Elapsed Time) – (Stopped Time) = (Actual Elapsed Time)  Note: To display this field, select Enable stopped time and actual elapsed time on the Entitlement Settings page and add the field to the object milestone page layout.
ActualElapsedTimeInMins	Type int Properties Filter, Group, Nillable, Sort Description The number of minutes that it took to complete a milestone. (Elapsed Time) – (Stopped Time) = (Actual Elapsed Time)  Note: To display this field, select Enable stopped time and actual elapsed time on the Entitlement Settings page and add the field to the object milestone page layout.
BusinessHoursId	Type reference Properties Filter, Group, Nillable, Sort Description The business hours on the milestone. If business hours aren't specified, the entitlement process business hours are used. If business hours are also not specified on the entitlement process, the business hours on the record are used.
CompletionDate	Type dateTime Properties Filter, Nillable, Sort, Update Description The date and time the milestone was completed.

Field Name	Details
CurrencyIsoCode	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Available only for orgs with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the organization.
ElapsedTimeInDays	Type double Properties Filter, Nillable, Sort Description The number of days it took to complete a milestone, including time during which the milestone was stopped. Automatically calculated to include the business hours on the record. Elapsed time is calculated only after the Completion Date field is populated. (Elapsed Time) – (Stopped Time) = (Actual Elapsed Time).
ElapsedTimeInHrs	Type double Properties Filter, Nillable, Sort Description The number of hours it took to complete a milestone, including time during which the milestone was stopped. Automatically calculated to include the business hours on the record. Elapsed time is calculated only after the Completion Date field is populated. (Elapsed Time) – (Stopped Time) = (Actual Elapsed Time).
ElapsedTimeInMins	Type int Properties Filter, Group, Nillable, Sort Description The number of minutes it took to complete a milestone, including time during which the milestone was stopped. Automatically calculated to include the business hours on the record. Elapsed time is calculated only after the Completion Date field is populated. (Elapsed Time) – (Stopped Time) = (Actual Elapsed Time).
IsCompleted	Type boolean Properties Defaulted on create, Filter, Group, Sort

Field Name	Details
	Description Green checkmark icon that indicates a milestone completion.
IsViolated	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Red exclamation point icon that indicates a milestone violation.
MilestoneTypeId	Type reference Properties Filter, Group, Nillable, Sort Description The ID of the milestone (for instance, First Response).
Name	Type string Properties Filter, Group, Sort, Update Description The name of the milestone.
ParentEntityId	Type reference Properties Filter, Group, Sort Description The ID of the record—for example, a work order—that contains the milestone.
SlaProcessId	Type reference Properties Filter, Group, Nillable, Sort Description The entitlement process associated with the milestone.
StartDate	Type dateTime

Field Name	Details
	Properties Filter, Nillable, Sort, Update Description The date and time that milestone tracking started.
StoppedTimeInDays	Type double Properties Filter, Nillable, Sort Description The number of days that an agent has been blocked from completing a milestone. For example, an agent may be waiting for a customer to reply with more information.  Note: To display this field, select Enable stopped time and actual elapsed time on the Entitlement Settings page and add the field to the object milestone page layout.
StoppedTimeInHrs	Type double Properties Filter, Nillable, Sort Description The number of hours that an agent has been blocked from completing a milestone. For example, an agent may be waiting for a customer to reply with more information.  Note: To display this field, select Enable stopped time and actual elapsed time on the Entitlement Settings page and add the field to the object milestone page layout.
StoppedTimeInMins	Type int Properties Filter, Group, Nillable, Sort Description The number of minutes that an agent has been blocked from completing a milestone. For example, an agent may be waiting for a customer to reply with more information.  Note: To display this field, select Enable stopped time and actual elapsed time on the Entitlement Settings page and add the field to the object milestone page layout.

Field Name	Details
TargetDate	Type dateTime Properties Filter, Nillable, Sort Description The date and time to complete the milestone.
TargetResponseInDays	Type double Properties Filter, Nillable, Sort Description The number of days to complete the milestone. Automatically calculated to include the business hours on the record.
TargetResponseInHrs	Type double Properties Filter, Nillable, Sort Description The number of hours to complete the milestone. Automatically calculated to include the business hours on the record.
TargetResponseInMins	Type int Properties Filter, Group, Nillable, Sort Description The number of minutes to complete the milestone. Automatically calculated to include the business hours on the record.
TimeRemainingInDays	Type string Properties Filter, Nillable, Sort Description The days that remain before a milestone violation. Automatically calculated to include the business hours on the record.
TimeRemainingInHrs	Type string

Field Name	Details
	Properties Filter, Group, Nillable, Sort Description The hours that remain before a milestone violation. Automatically calculated to include the business hours on the record.
TimeRemainingInMins	Type string Properties Group, Nillable, Sort Description The minutes that remain before a milestone violation. Automatically calculated to include the business hours on the record.
TimeSinceTargetInDays	Type string Properties Filter, Nillable, Sort Description The days that have elapsed since a milestone violation. Automatically calculated to include the business hours on the record.
TimeSinceTargetInHrs	Type string Properties Filter, Nillable, Sort Description The hours that have elapsed since a milestone violation. Automatically calculated to include the business hours on the record.
TimeSinceTargetInMins	Type string Properties Group, Nillable, Sort Description The minutes that have elapsed since a milestone violation. Automatically calculated to include the business hours on the record.

Usage

When you create an entitlement process, you select its type based on the type of record that you want the process to run on: Case or Work Order. Processes created before Summer '16 use the Case type. When a Work Order entitlement process runs on a work order, the

resulting milestones on the work order are object milestones. Conversely, when a Case entitlement process runs on a case, the resulting milestones are case milestones, a separate standard object.



Tip: If an entitlement has an entitlement process associated with it, don't use the entitlement for multiple types of support records. An entitlement process works only on records that match the process's type. For example, when a Case entitlement process is applied to an entitlement, the process runs only on cases associated with that entitlement. If a work order is also associated with the entitlement, the process doesn't run on the work order. To ensure that the milestones you set up work as expected, associate a customer's work orders and cases with different entitlements.

Customize page layouts, validation rules, and more for object milestones from the Object Milestones node in Setup under Entitlement Management.

Associated Objects

This object has the following associated objects. Unless noted, they are available in the same API version as this object.

EntityMilestoneFeed

Feed tracking is available for the object.

EntityMilestoneHistory

History is available for tracked fields of the object.

Expense

Represents an expense linked to a work order. Service resource technicians can log expenses, such as tools or travel costs. This object is available in API version 49.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
AccountId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the account associated with the linked work order.
Amount	Type currency Properties Create, Filter, Sort, Update Description The amount of the expense.

Field	Details
CurrencyIsoCode	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Available only if the multicurrency feature is enabled. Contains the ISO code for any currency allowed by the organization.
Description	Type textarea Properties Create, Nillable, Update Description A description for the expense.
Discount	Type percent Properties Create, Filter, Nillable, Sort, Update Description The percentage deducted from the <code>Subtotal</code> price. Available in version 51.0 and later.
ExpenseEndDate	Type date Properties Create, Filter, Group, Nillable, Sort, Update Description If the expense was incurred over multiple days, the Expense End Date is the last day that the expense covers.
ExpenseNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description The number that uniquely identifies the expense.
ExpenseStartDate	Type date Properties Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description If the expense was incurred over multiple days, the Expense Start Date is the first day that the expense covers.
ExpenseType	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description The type of expense. Possible values are: • Billable • Non-Billable The default value is Billable.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp for when the current user last viewed a record related to this record.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (LastReferencedDate) and not viewed.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The ID of the user who owns the expense record.
Quantity	Type double Properties Create, Filter, Nillable, Sort, Update

Field	Details
	Description The number of items purchased in this record. Available in version 51.0 and later.
Subtotal	Type currency Properties Filter, Nillable, Sort Description The subtotal price calculated as the product of <code>Quantity</code> and <code>UnitPrice</code> . Available in version 51.0 and later. This is a calculated field.
Title	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description A title that identifies the expense. This field is available in API version 50.0 and later.
TotalPrice	Type currency Properties Filter, Nillable, Sort Description The total price of the transaction which is equal to the discounted subtotal: <code>Subtotal - (Discount * Subtotal)</code> . Available in version 51.0 and later. This is a calculated field.
TransactionDate	Type date Properties Create, Filter, Group, Sort, Update Description The day that the expense was incurred, or the payment date for the expense.
UnitPrice	Type currency Properties Create, Filter, Nillable, Sort, Update

Field	Details
	Description The price of one item on the record. Available in version 51.0 and later.
WorkOrderId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the work order associated with the expense.

Associated Objects

This object has the following associated objects. Unless noted, they are available in the same API version as this object.

ExpenseChangeEvent (API version 55.0)

Change events are available for the object.

ExpenseFeed

Feed tracking is available for the object.

ExpenseHistory

History is available for tracked fields of the object.

ExpenseOwnerSharingRule

Sharing rules are available for the object.

ExpenseShare

Sharing is available for the object.

ExpenseReport

Represents a report that summarizes expenses. This object is available in API version 50.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
CurrencyIsoCode	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details
	Description Available only if the multicurrency feature is enabled. Contains the ISO code for any currency allowed by the organization.
Description	Type textarea Properties Create, Nillable, Update Description A description for the expense report.
ExpenseReportNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description An auto-generated number identifying the expense report.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp for when the current user last viewed a record related to this record.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (LastReferencedDate) and not viewed.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The ID of the user who owns the expense report record.
Title	Type string

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update Description A title that identifies the expense report.
TotalExpenseAmount	Type currency Properties Filter, Nillable, Sort Description The sum of all expense entries in the report. This is a calculated field.

Associated Objects

This object has the following associated objects. Unless noted, they are available in the same API version as this object.

ExpenseReportFeed

Feed tracking is available for the object.

ExpenseReportHistory

History is available for tracked fields of the object.

ExpenseReportShare

Sharing is available for the object.

ExpenseReportEntry

Represents an entry in an expense report. This object is available in API version 50.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
Amount	Type currency Properties Filter, Nillable, Sort Description The amount of the expense.

Field	Details
CurrencyIsoCode	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Available only if the multicurrency feature is enabled. Contains the ISO code for any currency allowed by the organization.
ExpenseId	Type reference Properties Create, Filter, Group, Sort, Update Description The expense that corresponds to the expense report entry.
ExpenseReportEntryNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description An auto-generated number identifying the expense report entry.
ExpenseReportId	Type reference Properties Create, Filter, Group, Sort Description The expense report that's associated with the expense report entry.
ExpenseType	Type picklist Properties Defaulted on create, Filter, Group, Nillable, Sort Description The type of expense. Possible values are: • Billable • Non-Billable The default value is <code>Billable</code>.
LastReferencedDate	Type dateTime

Field	Details
	Properties Filter, Nillable, Sort Description The timestamp for when the current user last viewed a record related to this record.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (LastReferencedDate) and not viewed.
Title	Type string Properties Filter, Group, Nillable, Sort Description A title that identifies the expense.
TransactionDate	Type date Properties Filter, Group, Nillable, Sort Description The day that the expense was incurred, or the payment date for the expense.

Associated Objects

This object has the following associated objects. Unless noted, they are available in the same API version as this object.

ExpenseReportEntryFeed

Feed tracking is available for the object.

ExpenseReportEntryHistory

History is available for tracked fields of the object.

FieldServiceMobileSettings

Represents a configuration of settings that control the Field Service iOS and Android mobile app experience. This object is available in API version 38.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field Name	Details
<code>AscAutomaticMode</code>	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Describes how status changes are handled. Possible values are: • <code>Off</code>—No automatic status changes. • <code>Manual</code>—The mobile worker can cancel or update the status change. • <code>Timed</code>—The mobile worker has a time period to prevent the status change. When the timer ends, the status changes. • <code>Automated</code>—The mobile worker is notified that the status has changed.
<code>AscCancellationTimerInSec</code>	Type int Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description For the Timed mode only. Time that the user has to cancel the appointment status change.
<code>AscCompletedStatus</code>	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description Status that indicates that a mobile worker completed a service appointment. Possible values are: • <code>Canceled</code> • <code>Cannot Complete</code> • <code>Completed</code> • <code>Dispatched</code> • <code>In Progress</code> • <code>None</code> • <code>Scheduled</code>

Field Name	Details
AscOnSiteStatus	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description Status that indicates that a mobile worker is at a service appointment. Possible values are: • Canceled • Cannot Complete • Completed • Dispatched • In Progress • None • Scheduled
AscRadiusInMeters	Type int Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description Service appointment radius that can trigger a status change.
AscTimeLimitationInMin	Type int Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description A time period when status changes can occur, before an appointment's scheduled start time and after the scheduled end time. The time is applied only if <code>IsAscTimeLimitEnabled</code> is <code>true</code>.
AscTravelStatus	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description Status that indicates that a mobile worker is traveling to a service appointment. Possible values are: • Canceled • Cannot Complete • Completed

Field Name	Details
	• Dispatched • In Progress • None • Scheduled
BgGeoLocationAccuracy	Type picklist Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update Description The accuracy of geolocation tracking of services resources while the app is running in the background. Lowering accuracy reduces battery consumption for mobile devices. Available in API version 41.0 and later. Picklist options: • Medium—Accurate to within about 100 meters. • Coarse—Accurate to within about 1 kilometer. • Very Coarse—Accurate to within about 3 kilometers. The default value is Coarse.
BgGeoLocationMinUpdateFreqMins	Type int Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The frequency of geolocation poling of services resources while the app is running in the background. Less frequent poling decreases battery consumption for mobile devices. The label in the UI is Minimum Update Frequency of Geo Location in Minutes (Background). Available in API version 41.0 and later.
BrandInvertedColor	Type string Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The color of toasts and the contrast color of the floating action button.
ContrastInvertedColor	Type string Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The color of secondary backgrounds in the UI.

Field Name	Details
ContrastPrimaryColor	Type string Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The color of primary text.
ContrastQuaternaryColor	Type string Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The color of secondary lines that delineate different areas of the UI.
ContrastQuinaryColor	Type string Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The color of primary backgrounds in the UI.
ContrastSecondaryColor	Type string Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The color of secondary text.
ContrastTertiaryColor	Type string Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The color of the icons on the settings screen and of primary lines that delineate different areas of the UI.
DaysBeforeCurrentServiceDate	Type int Properties Create, Defaulted on create, Filter, Group, Sort, Update

Field Name	Details
	Description Days before the current service date during which to prime service documents for offline use.
DayAfterCurrentServiceDate	Type int Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Days after the current service date during which to prime service documents for offline use.
DefaultListViewDeveloperName	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The API name of the default service appointment list view on the schedule screen.
DestinationType	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Determines if the mobile worker navigates to the destination based on the address or based on the latitude and longitude. Possible values are: - Address - Latitude and Longitude The default value is Address.
DeveloperName	Type string Properties Create, Filter, Group, Sort, Update Description The API name of the set of field service mobile settings. Only users with View DeveloperName OR View Setup and Configuration permission can view, group, sort, and filter this field.
FeedbackPrimaryColor	Type string

Field Name	Details
	Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The color of error messages.
FeedbackSecondaryColor	Type string Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The color of success messages.
FeedbackSelectedColor	Type string Properties Create, Defaulted on create, Group, Sort, Update Description The color indicating the user's current selection.
FutureDaysInDatePicker	Type int Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The number of days into the future that a user can select from the date picker on the schedule screen.
GeoLocationAccuracy	Type picklist Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update Description The accuracy of service resource geolocation tracking. Lowering accuracy reduces battery consumption for mobile devices. Picklist values: • Fine—Accurate to within 10 meters. • Medium—Accurate to within 100 meters. • Coarse—Accurate to within 1 kilometer. The default value is Medium.
GeoLocationMinUpdateFreqMins	Type int

Field Name	Details
	Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The minimum number of minutes between attempts to poll geolocation.
IsAscTimeLimitEnabled	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether AscTimeLimitationInMin is applied. Default is <code>true</code>
IsAssignmentNotification	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Controls whether service appointment notifications are sent when the service resource is assigned the appointment. Default is <code>false</code>. This field is available in API version 46.0 and later.
IsDefault	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Indicates that the set of field service mobile settings is the default set that is automatically assigned to users. You can't make a different settings record the default, but you can modify the default settings record. Default is <code>false</code>. Available in API version 41.0 and later.
IsDispatchNotification	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Controls whether service appointment notifications are sent when the service resource is dispatched for the appointment. Default is <code>false</code>. This field is available in API version 46.0 and later.
IsLimitedLocTrackingEnabled	Type boolean

Field Name	Details
	Properties Create, Defaulted on create, Filter, Group, Sort, Update Description When limited tracking for Appointment Assistant is enabled, the mobile worker's location is shown only on the way to a service appointment. The default value is <code>false</code>.
<code>IsOptimizedImageUploadEnabled</code>	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether to configure the size of images uploaded by your mobile workers. To optimize upload speeds, you can limit your file size to a defined maximum size using the <code>OptimizeImageSizeInMb</code> field. Resizing your images affects the resolution of your images. The default value is <code>false</code>.
<code>IsScheduleViewResourceAbsences</code>	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Determines whether resource absences appear in the Schedule tab of the mobile app. This field is available in API version 55.0 and later.
<code>IsSendLocationHistory</code>	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Controls whether geolocation tracking of services resources is enabled. Default is <code>false</code>.
<code>IsShowEditFullRecord</code>	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Controls whether users can edit records with the field service mobile app. Default is <code>false</code>.

Field Name	Details
IsTimeSheetEnabled	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Controls whether users can access time sheets on their mobile devices (Beta). Default is <code>false</code>.
IsTimeZoneEnabled	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Controls whether the time zone of timesheet entries on the mobile app is recorded. The current time zone is recorded in the <code>LocationTimeZone</code> field of the <code>TimeSheetEntry</code> object. Default is <code>false</code>. Available in API version 50.0 and later.
IsUseSalesforceMobileActions	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Reserved for future use.
Language	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The localization preference for a user. The format is a two letter language code and, if there's a dialect, followed by the two letter dialect, for example, <code>fr</code> for French, and <code>fr_BE</code> for Belgian French
MasterLabel	Type string Properties Create, Filter, Group, Sort, Update Description The label in the UI for the set of field service mobile settings. Available in API version 41.0 and later.

Field Name	Details
MaxNumberOfServiceAppointments	Type int Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Sets the maximum number of service appointments to use for offline priming of service documents. If you don't have dates on your service appointments, this setting helps to optimize offline priming in place of <code>DaysBeforeCurrentServiceDate</code> and <code>DaysBeforeCurrentServiceDate</code> fields.
MetadataCacheTimeDays	Type int Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The number of days that org metadata, such as layouts, is kept in the app's local cache of memory.
NavbarBackgroundColor	Type string Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The color of the top bar in the app.
NavbarInvertedColor	Type string Properties Create, Defaulted on create, Group, Sort, Update Description The secondary color of the tap bar in the app.
OptimizeImageSizeInMb	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description Configure the size of images uploaded by your mobile workers. To optimize upload speeds, you can limit your file size to a defined maximum size. Resizing your images affects the resolution of your images. Enter 0.2 or higher. Used only if <code>IsOptimizedImageUploadEnabled</code> is <code>true</code>.

Field Name	Details
PastDaysInDatePicker	Type int Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The number of days into the past that a user can select from the date picker on the schedule screen.
PrimaryBrandColor	Type string Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The main branding color used throughout the UI.
QuickStatusChangeFlowName	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The name of an existing Field Service flow with a Quick Status Change action to change the work order or service appointment status or both. This applies to flows invoked on the mobile app only. This field is available in API version 51.0 and later.
RecordDataCacheTimeMins	Type int Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The number of minutes that record data is kept in the app's local cache of memory.
SecondaryBrandColor	Type string Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The color of action buttons.
TimeIntervalSetupMins	Type picklist

Field Name	Details
	Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update
	Description Controls the spacing of picklist options for time values such as when creating resource absences.
UpdateScheduleTimeMins	Type int
	Properties Create, Defaulted on create, Filter, Group, Sort, Update
	Description The minimum number of minutes between attempts to update a user's schedule. The user's schedule might not refresh on this cadence if the user's device isn't connected to a network or doesn't have adequate battery life.

Usage

Field Service Mobile settings allow you to create sets of settings to apply to different field service mobile users. The settings apply to both the Android and iOS versions of the app.

For example, suppose you want to accommodate workers that are color blind, or who work in dark or bright conditions. You can choose different branding options for different workers to suit their needs, and assign them to their profiles.

Associated Objects

This object has the following associated objects. Unless noted, they are available in the same API version as this object.

FieldServiceMobileSettingsChangeEvent (API version 55.0)

Change events are available for the object.

FldSvcObjChg

Represents a change made to one of a service appointment's tracked fields. This object is available in API version 63.0 and later.

Supported Calls

`describeLayout()`, `describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

- Field Service must be enabled.
- The Field Service managed package must be installed.
- The **Track the lifecycle of service appointments** setting in **Setup > Field Service Settings** must be enabled.
- The Platform Integration User must have the Manage Service Appointment Lifecycle and the View Service Appointment Lifecycle permissions.
- To view this object, users must have the View Service Appointment Lifecycle user permission.

Fields

Field	Details
Activity	Type picklist Properties Filter, Group, Nillable, Restricted picklist, Sort Description The available scheduling activities for the service appointment. Possible values are: • AddedToBundle—Currently not supported • BundleMemberAdded—Currently not supported • BundleMemberRemoved—Currently not supported • Created • Deleted • RemovedFromBundle—Currently not supported • Rescheduled—An appointment is considered rescheduled if a change is made to its assigned service resource or to its scheduled start time. • Scheduled • ServiceResourceAssigned • StatusChanged—The service appointment status was changed. The manual activities of scheduling, rescheduling, and unscheduling are also reported as status changes because they change the status of an appointment. • Unscheduled • Updated—Captures changes made to one or more of the tracked scheduling fields that aren't associated with another activity.
ActivityDetails	Type string Properties Filter, Nillable, Sort Description Details about the scheduling activity such as the scheduling policy that was used and the unscheduling reason (when applicable).
ActivityTimeStamp	Type dateTime Properties Filter, Nillable, Sort Description Date and time of when the change to the service appointment was made.

Field	Details
IsPrimary	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Indicates whether the change was made directly to the service appointment or indirectly. If the change was made directly to the appointment, it's flagged as Primary. If it was made to another appointment and affected this one, it's flagged as Secondary.
OriginalSvcAppointment	Type string Properties Filter, Group, Nillable, Sort Description The ID of the service appointment that was changed. The field value isn't deleted when the Service Appointment object is deleted.
ServiceAppointmentId	Type reference Properties Filter, Group, Nillable, Sort Description The ID of the service appointment that was changed. Because this is a relationship field, the field value is deleted when the Service Appointment object is deleted. This field is a relationship field. Relationship Name ServiceAppointment Relationship Type Lookup Refers To ServiceAppointment
TimeZone	Type picklist Properties Filter, Group, Nillable, Restricted picklist, Sort Description The time zone of the service appointment or the assigned resource.
Transaction	Type string

Field	Details
	Properties Filter, Nillable, Sort Description The last transaction ID of the scheduling and optimization request that updated this object. The transaction ID is automatically generated and populated by the Enhanced Scheduling and Optimization engine.
UserId	Type reference Properties Filter, Group, Nillable, Sort Description The user who made the change to the service appointment. If an activity is a scheduled job, it's registered with the System Administrator user who configured it. This field is a relationship field. Relationship Name User Relationship Type Lookup Refers To User

SEE ALSO:

[FldSvcObjChgDtl](#)

FldSvcObjChgDtl

Represents the details of a change made to one of a service appointment's tracked fields. This object is available in API version 63.0 and later.

Supported Calls

`describeLayout()`, `describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

- Field Service must be enabled.
- The Field Service managed package must be installed.
- The **Track the lifecycle of service appointments** setting in **Setup > Field Service Settings** must be enabled.
- The Platform Integration User must have the Manage Service Appointment Lifecycle and the View Service Appointment Lifecycle permissions.
- To view this object, users must have the View Service Appointment Lifecycle user permission.

Fields

Field	Details
FieldChangedName	Type string Properties Filter, Group, Sort Description The API name of the field that was changed.
FieldChangedValue	Type string Properties Filter, Nillable, Sort Description The updated value of the field that was changed.
FldSvcObjChgId	Type reference Properties Filter, Group, Sort Description The parent record associated with the details of the change. The parent record has multiple detail records associated with it. Each record includes the name and the current value of the tracked fields. This field is a relationship field. Relationship Name FldSvcObjChg Relationship Type Lookup Refers To FldSvcObjChg

SEE ALSO:

[FldSvcObjChg](#)

FSL__Time_Dependency__c

Represents a dependency between two service appointments. This object is used to define scheduling relationships between two appointments. It allows you to determine the order and timing in which dependent appointments should be scheduled.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

This object is available in orgs that have the Field Service managed package installed. Records of this object are subject to standard sharing and security settings.

Fields

Field	Details
<code>FSL__Dependency__c</code>	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description The type of the dependency. Possible values are: • Immediately Follow • Same Start • Start After Finish • Start After Finish And Same Day The default value is Same Start.
<code>FSL__Root_Service_Appointment__c</code>	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The service appointment that's the root appointment of the entire dependency chain. The root appointment is the same service appointment for all the dependencies in the chain. This field is automatically populated by the system when creating a dependency. When creating a dependency via API, the API user must have permissions for this field. We recommend leaving this field empty when a dependency is created and avoiding any custom modifications. This field is a relationship field. Relationship Name <code>FSL__Root_Service_Appointment__r</code> Relationship Type Lookup Refers To ServiceAppointment

Field	Details
FSL__Same_Resource__c	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description An indication of whether the two service appointments that share a dependency should be assigned to the same service resource. The default value is <code>false</code>.
FSL__Service_Appointment_1__c	Type reference Properties Create, Filter, Group, Sort Description The first service appointment on which the second service appointment is dependent. This field is a relationship field. Relationship Name FSL__Service_Appointment_1__r Relationship Type Master-detail Refers To ServiceAppointment (the master object)
FSL__Service_Appointment_2__c	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The second service appointment that's dependent on the first service appointment. This field is a relationship field. Relationship Name FSL__Service_Appointment_2__r Relationship Type Lookup Refers To ServiceAppointment
Name	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort

Field	Details
	Description An auto-generated unique identifier for the appointment dependency record.

GeolocationBasedAction

Represents a geolocation-based action, which is an action that's triggered when a user enters, exits, or is within the area of the associated object. Available in API version 61.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field	Details
ActionData	Type textarea Properties Create, Nillable, Update Description The details of the selected action type.
ActionType	Type picklist Properties Create, Filter, Group, Restricted picklist, Sort, Update Description The type of action. Possible values are: - PlatformAlert - QuickAction - ViewRecord
Description	Type textarea

Field	Details
	Properties Create, Nillable, Update Description The description of the action.
InitialTimeInvoked	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description Captures the first time the mobile worker invoked this action.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last accessed this record indirectly, for example, through a list view or related record.
LastTimeInvoked	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description Captures the last time the mobile worker invoked this action.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last viewed this record or list view. If this value is null, and <code>LastReferenceDate</code> isn't null, the user accessed this record or list view indirectly.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The name of the action.

Field	Details
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description ID of the owner of this object. This field is a polymorphic relationship field. Relationship Name Owner Relationship Type Lookup Refers To Group, User
Radius	Type int Properties Create, Filter, Group, Sort, Update Description The distance in meters from the location of the associated object that triggers the action.
ReferenceRecordId	Type reference Properties Create, Filter, Group, Sort, Update Description The ID of the record that the action is associated with. This field is a relationship field. Relationship Name ReferenceRecord Relationship Type Lookup Refers To ServiceAppointment
TriggerType	Type picklist Properties Create, Filter, Group, Restricted picklist, Sort, Update Description The event that triggered this action.

Field	Details
	Possible values are: • <code>GeoFenceEnter</code>—Enter • <code>GeoFenceExit</code>—Exit

LinkedArticle

Represents a knowledge article that is attached to a work order, work order line item, or work type. This object is available in API version 37.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Knowledge must be enabled in your org. Field Service must be enabled. Only users that have access to the Knowledge article and the parent record linked to it can access this object.

In Knowledge in Salesforce Classic, only Field Service objects such as Work Order, Work Type, and Work Order Line Item are supported for linked articles. In Lightning Knowledge, other social objects such as Chat, Messaging, Voice Call, and Social Post are supported for linked articles.

To call `update()` to attach or detach articles, enable the Read user permission on the Knowledge object and the Edit user permission on the object whose article you update. Available in API version 58.0 and later.

Fields

Field Name	Details
<code>CurrencyIsoCode</code>	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Available only for orgs with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the organization.
<code>KnowledgeArticleId</code>	Type reference Properties Create, Filter, Group, Nillable, Sort Description The ID of the Knowledge article attached to the record. The label in the user interface is Knowledge Article ID.

Field Name	Details
KnowledgeArticleVersionId	Type reference Properties Create, Filter, Group, Nillable, Sort Description The version of the Knowledge article attached to the record. This field lists the title of the attached version and links to the version. The label in the user interface is Article Version. When you attach an article to a work order, that version of the article stays associated with the work order, even if later versions are published. If needed, you can detach and reattach an article to a work order to link the latest version.
LinkedEntityId	Type reference Properties Create, Filter, Group, Nillable, Sort Description The ID of the record that the Knowledge article is attached to. The label in the user interface is Linked Record ID.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The title of the article. The label in the user interface is Article Title.
RecordTypeId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the article's record type, if used. This field is only available for Lightning Knowledge.
Type	Type string Properties Filter, Group, Nillable, Sort Description (Read only) The type of record that the Knowledge article is attached to. For example, work order. The label in the user interface is Linked Object Type.

Usage

Admins can customize linked articles' page layouts, fields, validation rules, and more from the Linked Articles page in Setup.

Associated Objects

This object has the following associated objects. Unless noted, they are available in the same API version as this object.

LinkedArticleChangeEvent (API version 62.0)

Change events are available for the object.

LinkedArticleFeed

Feed tracking is available for the object.

LinkedArticleHistory

History is available for tracked fields of the object.

Location

Represents a warehouse, service vehicle, work site, or other element of the region where your team performs field service work. In API version 49.0 and later, you can associate activities with specific locations. Activities, such as the tasks and events related to a location, appear in the activities timeline when you view the location detail page. Also in API version 49.0 and later, Work.com users can view Employees as a related list on Location records. In API version 51.0 and later, this object is available for Omnichannel Inventory and represents physical locations where inventory is available for fulfilling orders.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

At least one of these features must be enabled:

- Commerce Store
- Contact Tracing for Employees
- Employee Experience
- Field Service
- Fulfillment Orders
- Health Cloud
- Industries Insurance
- Industries Visit
- Locations
- Omnichannel Inventory
- Public Sector
- Retail Execution
- Work.com

Fields

Field Name	Details
AssignedFoCount	Type int Properties Create, Filter, Group, Nillable, Sort, Update Description The number of fulfillment orders assigned to the location. Confirming held fulfillment order capacity increments this value. To reset the location's capacity, set this value to 0. This field is available when Order Management is installed and configured. By default, it's hidden by field-level security. This field is available in API version 55.0 and later.
CloseDate	Type date Properties Create, Filter, Group, Nillable, Sort, Update Description Date the location closed or went out of service.
ConstructionEndDate	Type date Properties Create, Filter, Group, Nillable, Sort, Update Description Date construction ended at the location.
ConstructionStartDate	Type date Properties Create, Filter, Group, Nillable, Sort, Update Description Date construction began at the location.
DefaultPickupTime	Type time Properties Create, Filter, Group, Nillable, Sort, Update Description Default pickup time at the location.

Field Name	Details
	This field is available when Order Management is installed and configured. By default, it's hidden by field-level security. This field is available in API version 61.0 and later.
DefaultProcessingTime	Type int Properties Create, Filter, Group, Nillable, Sort, Update Description Default processing time at the location. This field is available when Order Management is installed and configured. By default, it's hidden by field-level security. This field is available in API version 61.0 and later.
DefaultProcessingTimeUnit	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update Description Default processing time unit at the location. Possible values are: • Hours • Days • Weeks This field is available when Order Management is installed and configured. By default, it's hidden by field-level security. This field is available in API version 61.0 and later.
Description	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description A brief description of the location.
DrivingDirections	Type string Properties Create, Filter, Nillable, Sort, Update Description Directions to the location.

Field Name	Details
EarliestPickupTimeOffset	Type integer Properties Create, Filter, Nillable, Sort, Update Description The earliest pickup time for BOPIS. This value is measured in minutes after the start of business hours.
ExternalReference	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description Identifier of a location.
FoCapacity	Type int Properties Create, Filter, Group, Nillable, Sort, Update Description The maximum number of fulfillment orders that can be assigned to the location per time period. If this value is null, then this location's capacity isn't limited. This field is available when Order Management is installed and configured. By default, it's hidden by field-level security. This field is available in API version 55.0 and later.
FulfillingBusinessHours	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description Fulfilling business hours at the location. This field is available when Order Management is installed and configured. By default, it's hidden by field-level security. This field is available in API version 61.0 and later.
FoCapacity	Type int Properties Create, Filter, Group, Nillable, Sort, Update

Field Name	Details
	Description The maximum number of fulfillment orders that can be assigned to the location per time period. If this value is null, then this location's capacity isn't limited. This field is available when Order Management is installed and configured. By default, it's hidden by field-level security. This field is available in API version 55.0 and later.
IsEligibleForPickup	Type boolean Properties Create, Filter, Nillable, Sort, Update Description Indicates whether the location supports BOPIS
IsInventoryLocation	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether the location stores parts.  Note: This field must be selected if you want to associate the location with product items.
IsMobile	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether the location moves. For example, a truck or tool box.
LastReferencedDate	Type datetime Properties Filter, Nillable, Sort Description The date when the location was last modified. Its label in the user interface is <code>Last Modified Date</code>.
LastViewedDate	Type datetime

Field Name	Details
	Properties Filter, Nillable, Sort Description The date the location was last viewed.
LatestPickupTimeOffset	Type integer Properties Create, Filter, Nillable, Sort, Update Description The latest pickup time for BOPIS. This value is measured in minutes before the end of business hours.
Latitude	Type double Properties Create, Filter, Nillable, Sort, Update Description The latitude of the location.
Location	Type location Properties Nillable Description The geographic location.
LocationLevel	Type int Properties Filter, Group, Nillable, Sort Description The location's position in a location hierarchy. If the location has no parent or child locations, its level is 1. Locations that belong to a hierarchy have a level of 1 for the root location, 2 for the child locations of the root location, 3 for their children, and so forth.
LocationType	Type picklist Properties Create, Filter, Group, Sort, Update

Field Name	Details
	Description Picklist of location types. It has no default values, so you must populate it before creating any location records.
LogoId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description A ContentAsset representing a logo for the location. This field is available in API version 50.0 and later. This is a relationship field. Relationship Name Logo Relationship Type Lookup Refers To ContentAsset
Longitude	Type double Properties Create, Filter, Nillable, Sort, Update Description The longitude of the location.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The name of the location. For example, Service Van #4.
OpenDate	Type date Properties Create, Filter, Group, Nillable, Sort, Update Description Date the location opened or came into service.

Field Name	Details
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The location's owner or driver. This is a polymorphic relationship field. Relationship Name Owner Relationship Type Lookup Refers To Group, User
ParentLocationId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The location's parent location. For example, if vans are stored at a warehouse when not in service, the warehouse is the parent location. This is a relationship field. Relationship Name ParentLocation Relationship Type Lookup Refers To Location
PickupProcessingTime	Type integer Properties Create, Filter, Nillable, Sort, Update Description The processing time required for BOPIS orders at this location.
PossessionDate	Type date Properties Create, Filter, Group, Nillable, Sort, Update

Field Name	Details
	Description The date the location was purchased.
Priority	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update Description The priority of the location when routing orders. No default values are included. Add values to the picklist and reference them in your custom routing logic. This field is available when Order Management is installed and configured. By default, it's hidden by field-level security. This field is available in API version 55.0 and later.
RemodelEndDate	Type date Properties Create, Filter, Group, Nillable, Sort, Update Description Date when remodel construction ended at the location.
RemodelStartDate	Type date Properties Create, Filter, Group, Nillable, Sort, Update Description Date when remodel construction started at the location.
RootLocationId	Type reference Properties Filter, Group, Nillable, Sort Description (Read Only) The top-level location in the location's hierarchy. This is a relationship field. Relationship Name RootLocation Relationship Type Lookup Refers To Location

Field Name	Details
ShouldSyncWithOci	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether the location should sync its data with Omnichannel Inventory. The default value is <code>false</code>. This field is available in API version 51.0 and later.
ShouldTrackFoCapacity	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether the location should track its fulfillment order capacity. The default value is <code>false</code>. This field is available when Order Management is installed and configured. By default, it's hidden by field-level security. This field is available in API version 55.0 and later.
TimeZone	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Picklist of available time zones.
VisitorAddressId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description Lookup to an account's or client's address. This is a relationship field. Relationship Name VisitorAddress Relationship Type Lookup Refers To Address

Usage

Before creating any location records, add at least one value to the Location Type picklist. LocationType is a required field.

To track your inventory in Salesforce, create product items, which represent the stock of a particular product at a particular location. For example, create a product item that represents the 500 bolts you have in stock at your Warehouse A location. Each product item must be associated with a location.

To get a more granular picture of your field service operation, associate locations with service territories. For example, if a warehouse is located in a particular service territory, add it as a service territory location.



Important: “Location” in Salesforce can also refer to the geolocation compound field found on many standard objects. When referencing the Location object in your Apex code, always use `Schema.Location` instead of `Location` to prevent confusion with the standard Location compound field. If referencing both the Location object and the Location field in the same snippet, you can differentiate between the two by using `System.Location` for the field and `Schema.Location` for the object.

Associated Objects

This object has the following associated objects. Unless noted, they are available in the same API version as this object.

LocationChangeEvent (API version 48.0)

Change events are available for the object.

LocationFeed

Feed tracking is available for the object.

LocationHistory

History is available for tracked fields of the object.

LocationOwnerSharingRule

Sharing rules are available for the object.

LocationShare

Sharing is available for the object.

MaintenanceAsset

Represents an asset covered by a maintenance plan in field service. Assets can be associated with multiple maintenance plans.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field Name	Details
AssetId	Type reference

Field Name	Details
	Properties Create, Filter, Group, Sort, Update Description The asset associated with the maintenance asset.
ContractLineItemId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description Contract line item associated with the maintenance asset. This field can only list a contract line item that is associated with the asset, and whose parent service contract is associated with the parent maintenance plan.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the maintenance asset was last modified. Its label in the user interface is Last Modified Date.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the product request was last viewed.
MaintenanceAssetNumber	Type string Properties Autonumber, Defaulted on create, Filter, Sort Description An auto-assigned number that identifies the maintenance asset.
MaintenancePlanId	Type reference Properties Create, Filter, Group, Sort Description Maintenance plan associated with the maintenance asset.

Field Name	Details
NextSuggestedMaintenanceDate	Type date Properties Create, Filter, Group, Nillable, Sort, Update Description The suggested date of service for the maintenance asset's first work order (not the date the work order is created). This corresponds to the work order's SuggestedMaintenanceDate. If left blank when the maintenance asset is created, this field inherits its initial value from the related maintenance plan. This field auto-updates after each batch is generated. Its label in the user interface is Date of the first work order in the next batch.
WorkTypeId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description Work type associated with the maintenance asset. Work orders generated from the maintenance plan inherit its work type's duration, required skills and products, and linked articles. Maintenance assets covered by the plan use the same work type, though you can update them to use a different one.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

MaintenanceAssetChangeEvent (API version 48.0)

Change events are available for the object.

MaintenanceAssetFeed

Feed tracking is available for the object.

MaintenanceAssetHistory

History is available for tracked fields of the object.

MaintenancePlan

Represents a preventive maintenance schedule for one or more assets in field service.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field Name	Details
AccountId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The associated account, which typically represents the customer receiving the maintenance service.
ContactId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The associated contact.
Description	Type textarea Properties Create, Nillable, Update Description A brief description of the plan.
DoesAutoGenerateWorkOrders	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Turns on auto-generation of work order batches for a maintenance plan and prohibits the manual generation of work orders via the Generate Work Orders action. If this option is selected, a new batch of work orders is generated for the maintenance plan on the <code>NextSuggestedMaintenanceDate</code> listed on each maintenance asset, or on the maintenance plan if no assets are included. If a <code>GenerationHorizon</code> is specified, the date of generation is that many days earlier.
DoesGenerateUponCompletion	Type boolean

Field Name	Details
	Properties Create, Defaulted on create, Filter, Group, Sort, Update Description If both this option and <code>DoesAutoGenerateWorkOrders</code> are set to true, a new batch of work orders isn't generated until the last work order generated from the maintenance plan is completed. A work order is considered completed when its status falls into one of the following status categories: Cannot Complete, Canceled, Completed, or Closed. If a maintenance plan covers multiple assets, work orders are generated per asset. If a maintenance asset's final work order is completed late, its work order generation is delayed, which may cause a staggered generation schedule between maintenance assets.
EndDate	Type date Properties Create, Filter, Group, Nillable, Sort, Update Description The last day the maintenance plan is valid.
Frequency	Type int Properties Create, Filter, Group, Sort, Update Description (Optional) Amount of time between work orders. The unit is specified in the <code>FrequencyType</code> field.
FrequencyType	Type picklist Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update Description (Optional) The unit of frequency: • Days • Weeks • Months • Years For example, to perform monthly maintenance visits you need a work order for each visit, so enter 1 as the <code>Frequency</code> and select Months.

Field Name	Details
GenerationHorizon	Type int Properties Create, Filter, Group, Nillable, Sort, Update Description Moves up the timing of batch generation if DoesAutoGenerateWorkOrders is set to true. A generation horizon of 5 means the new batch of work orders is generated 5 days before the maintenance asset's (or maintenance plan's, if there are no assets) NextSuggestedMaintenanceDate. The generation horizon must be a whole number.
GenerationTimeframe	Type int Properties Create, Filter, Group, Sort, Update Description (Required) How far in advance work orders are generated in each batch. The unit is specified in the GenerationTimeframeType field.
GenerationTimeframeType	Type picklist Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update Description (Required) The generation timeframe unit: • Days • Weeks • Months • Years For example, if you need work orders for six months, enter 6 and select Months.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last interacted with this record, directly or indirectly.

Field Name	Details
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last viewed this record or list view. If this value is null, it's possible that the user only accessed this record or list view (LastReferencedDate), but not viewed it.
LocationId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description Where the service takes place.
MaintenancePlanNumber	Type string Properties Autonumber, Defaulted on create, Filter, Sort Description (Read Only) An auto-assigned number that identifies the maintenance plan.
MaintenancePlanTitle	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description A name for the maintenance plan.
MaintenanceWindowEndDays	Type int Properties Create, Filter, Group, Nillable, Sort, Update Description Days after the suggested service date on the work order that its service appointment can be scheduled.
MaintenanceWindowStartDays	Type int Properties Create, Filter, Group, Nillable, Sort, Update

Field Name	Details
	Description Days before the suggested service date on the work order that its service appointment can be scheduled. The maintenance window start and end fields affect the Earliest Start Permitted and Due Date fields on the maintenance plan's work orders' service appointments. For example, if you enter 3 for both the maintenance window start and end, the Earliest Start Permitted and the Due Date will be 3 days before and 3 days after, respectively, the Suggested Maintenance Date on each work order. If the maintenance window fields are left blank, the service appointment date fields list their work order's suggested maintenance date.
NextSuggestedMaintenanceDate	Type date Properties Create, Filter, Group, Sort, Update Description The suggested date of service for the first work order (not the date the work order is created). This corresponds to the work order's SuggestedMaintenanceDate. You can use this field to enforce a delay before the first maintenance visit (for example, if monthly maintenance should begin one year after the purchase date). Its label in the user interface is Date of the first work order in the next batch. For example, if you want the first maintenance visit to take place on May 1, enter May 1. When you generate work orders, the earliest work order will list a suggested maintenance date of May 1, and the dates on the later work orders will be based on the GenerationTimeframe and Frequency.  Important: Maintenance assets also list a NextSuggestedMaintenanceDate, which is initially inherited from the maintenance plan. If the plan has maintenance assets, this date auto-updates on the maintenance assets after each batch is generated, but doesn't update on the maintenance plan itself because batch timing is calculated at the maintenance asset level. If the plan doesn't have maintenance assets, this date auto-updates on the maintenance plan after each batch is generated.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The owner of the maintenance plan.

Field Name	Details
ServiceContractId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The service contract associated with the maintenance plan. The service contract can't be updated if any child maintenance asset is associated with a contract line item from the service contract.
StartDate	Type date Properties Create, Filter, Group, Sort, Update Description The first day the maintenance plan is valid.
SvcApptGenerationMethod	Type picklist Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update Description The service appointment generation method. - One service appointment per work order - One service appointment per work order line item If your existing maintenance plans have work orders or work order line items associated with them, you can't change their generation methods. To change pre-existing maintenance plan generation methods, either delete the work orders and regenerate them or delete the maintenance plan and recreate it with the needed generation methods. If Work Order Generation Method is set to One work order per asset, you can't set a Service Appointment Generation Method. If Work Order Generation Method is set to One work order line item per asset, you must select a Service Appointment Generation Method.
WorkOrderGenerationMethod	Type picklist Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update Description The work order generation method. One work order per asset

Field Name	Details
	One work order line item per asset If your existing maintenance plans have work orders or work order line items associated with them, you can't change their generation methods. To change pre-existing maintenance plan generation methods, either delete the work orders and regenerate them or delete the maintenance plan and recreate it with the needed generation methods. If Work Order Generation Method is left as None, the generation is defaulted to one work order per asset. When One work order line item per asset is set, and all maintenance assets have the same Next Suggested Maintenance Date on the maintenance plan, they are grouped in one work order. However, if maintenance assets have different Next Suggested Maintenance Dates, multiple work orders are created for each date. If Work Order Generation Method is set to One work order per asset, you can't set a Service Appointment Generation Method.
WorkOrderGenerationStatus	Type picklist Properties Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort Description (Read Only) Indicates the status of work order generation: - NotStarted—the default value, work order generation has not started - InProgress—work order generation is underway - Completed—work order generation is complete - Unsuccessful—it was not possible to generate work orders You can generate only one batch at a time.
WorkTypeId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The associated work type. Work orders generated from the maintenance plan inherit its work type's duration, required skills and products, and linked articles. Maintenance assets covered by the plan use the same work type, though you can update them to use a different one.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

MaintenancePlanChangeEvent (API version 48.0)

Change events are available for the object.

MaintenancePlanFeed

Feed tracking is available for the object.

MaintenancePlanHistory

History is available for tracked fields of the object.

MaintenancePlanOwnerSharingRule

Sharing rules are available for the object.

MaintenancePlanShare

Sharing is available for the object.

MaintenanceWorkRule

Represents the recurrence pattern for a maintenance record. This object is available in API version 49.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
DoesFloatingWorkOrder	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates that the maintenance plan uses the floating work order adjustment. The default is false.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the line item was last modified. Its label in the user interface is <code>Last Modified Date</code> .
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort

Field	Details
	Description The date when the line item was last viewed.
Name	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description Name of this maintenance work rule.
NextSuggestedMaintenanceDate	Type date Properties Create, Filter, Group, Nillable, Sort, Update Description The next date on which this rule will generate maintenance items.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The assigned owner of the maintenance work rule.
ParentMaintenancePlanId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The maintenance plan associated with the maintenance work rule.
ParentMaintenanceRecordId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The maintenance record this work rule applies to.
RecordsetFilterCriteriaId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description ID of the recordset filter criteria associated with this maintenance work rule. Available in API version 52.0 and later.
RecurrencePattern	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The RRULE that defines the pattern of recurrence for this work order rule.
SortOrder	Type int Properties Create, Filter, Group, Sort, Update Description The sort order that applies to this work order rule.
Title	Type string Properties Create, Filter, Group, Sort, Update Description The title of this work order rule.
Type	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The type of maintenance work rule. Available values are: - Criteria-based - Calendar-based (default) Available in API version 52.0 and later.
WorkTypeId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the work type that this work order rule generates.

Associated Objects

This object has the following associated objects. Unless noted, they are available in the same API version as this object.

MaintenanceWorkRuleChangeEvent

Change events are available for the object.

MaintenanceWorkRuleFeed

Feed tracking is available for the object.

MaintenanceWorkRuleHistory

History is available for tracked fields of the object.

MaintenanceWorkRuleOwnerSharingRule

Sharing rules are available for the object.

MaintenanceWorkRuleShare

Sharing is available for the object.

MobileSettingsAssignment

Represents the assignment of a particular field service mobile settings configuration to a user profile. This object is available in API version 41.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()` —available in API version 51.0 and later, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field Name	Details
<code>FieldServiceMobileSettingsId</code>	Type reference Properties Create, Filter, Group, Sort, Update Description The ID of a set of field service mobile settings.
<code>ProfileId</code>	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the profile to associate with the set of field service mobile settings.

OperatingHours

Represents the hours in which a service territory, service resource, or account is available for work. OperatingHours is used by Field Service, Salesforce Scheduler, Salesforce Meetings, Sales Engagement, and Workforce Engagement. This object is available in API version 38.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field Name	Details
Description	Type textarea Properties Create, Nillable, Update Description The description of the operating hours. Add any details that aren't included in the name.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the operating hours record was last modified. Its label in the user interface is <code>Last Modified Date</code> .
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the operating hours record was last viewed.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The name of the operating hours. For example, <i>Summer Hours</i> , <i>Winter Hours</i> , or <i>Peak Season Hours</i> .

Field Name	Details
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The ID of the owner of the operating hours record. This field is available in API version 59.0. This field is a polymorphic relationship field. Relationship Name Owner Relationship Type Lookup Refers To Group, User
TimeZone	Type picklist Properties Create, Filter, Group, Restricted picklist, Sort, Update Description The time zone that the operating hours fall within.

Usage

By default, only System Administrators can view, create, and assign operating hours.

Service territory members—which are service resources who can work in the territory—automatically use their service territory’s operating hours. If a resource needs different operating hours than their territory, create separate operating hours for them from the Operating Hours tab. Then, select the desired hours in the `Operating Hours` lookup field on the service territory member detail page.

To view a service resource’s operating hours for a particular territory, navigate to their Service Territories related list and click the Member Number for the territory. You reach the service territory member detail page, which lists the member’s operating hours and dates during which they belong to the territory.

Associated Objects

This object has the following associated objects. Unless noted, they’re available in the same API version as this object.

OperatingHoursChangeEvent (API version 54.0)

Change events are available for the object.

OperatingHoursHistory (API version 62.0)

History is available for tracked fields of the object.

OperatingHoursHoliday

Represents the day or hours for which a service territory or service resource is unavailable in Field Service, Salesforce Scheduler, Salesforce Meetings, Sales Engagement, or Workforce Engagement. This object is available in API version 54.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field	Details
DateAndTime	Type string Properties Filter, Group, Nillable, Sort Description (Read-Only) The date or time for the holiday.
HolidayId	Type reference Properties Create, Filter, Group, Sort, Update Description The ID of the holiday that's related to the operating hours indicated in the OperatingHoursId field. This value is a relationship field. Relationship Name Holiday Relationship Type Lookup Refers To Holiday
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date and time that the current user last viewed a record related to this object.

Field	Details
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp for when the current user last viewed this object.
OperatingHoursHolidayNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description (Read-Only) An auto-generated number identifying the operating hours holiday.
OperatingHoursId	Type reference Properties Create, Filter, Group, Sort Description The ID of the operating hours that's related to the holiday that's indicated in the HolidayId field. This value is a relationship field. Relationship Name OperatingHours Relationship Type Lookup Refers To OperatingHours

Associated Objects

This object has the following associated objects. If the API version isn't specified, the associated objects are either available in the same API versions as the OperatingHoursHoliday object or in the specified API version and later.

[OperatingHoursHolidayChangeEvent](#) (API version 62.0)

Change events are available for the object.

[OperatingHoursHolidayHistory](#) (API version 62.0)

History is available for tracked fields of the object.

Pricebook2

Represents a price book that contains the list of products that your org sells.



Note: Price books are represented by Pricebook2 objects. As of API version 8.0, the Pricebook object is no longer available. Requests containing Pricebook are refused, and responses don't contain the Pricebook object.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
Description	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description Text description of the price book.
IsActive	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether the price book is active (<code>true</code>) or not (<code>false</code>). Inactive price books are hidden in many areas in the user interface. You can change this field's value as often as necessary. Label is Active.
IsArchived	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Indicates whether the price book has been archived (<code>true</code>) or not (<code>false</code>). This field is read only.
IsDeleted	Type boolean Properties Defaulted on create, Filter Description Indicates whether the price book has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.

Field	Details
IsStandard	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Indicates whether the price book is the standard price book for the org (<code>true</code>) or not (<code>false</code>). Every org has one standard price book—all other price books are custom price books.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp for when the current user last viewed a record related to this record.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp for when the current user last viewed this record. If this value is null, it's possible that this record was referenced (<code>LastReferencedDate</code>) and not viewed.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description Required. Name of this object. This field is read-only for the standard price book. Label is Price Book Name.
ValidFrom	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description The date and time when a Commerce price book is initially valid. If this field is <code>null</code>, the price book is valid immediately when active. Available in API version 48.0 and later.
ValidTo	Type dateTime

Field	Details
	Properties Create, Filter, Nillable, Sort, Update
	Description The date and time when a Commerce price book is valid to. If this field is <code>null</code> , the price book is valid until it's deactivated. Available in API version 48.0 and later.

Usage

A price book is a list of products that your org sells.

- Each org has one standard price book that defines the standard or generic list price for each product or service that it sells.
- An org can have multiple custom price books to use for specialized purposes, such as for discounts, different channels or markets, or select accounts or opportunities. While your client application can create, delete, and update custom price books, your client application can only update the standard price book.
- For some orgs, the standard price book is the only price needed. If you set up other price books, you can reference the standard price book when setting up list prices in custom price books.

Use this object to query standard and custom price books that have been configured for your org. A common use of this object is to allow your client application to obtain valid Pricebook2 object IDs for use when configuring PricebookEntry records via the API.

Your client application can perform the following tasks on PricebookEntry objects:

- Query
- Create for the standard price book or custom price books.
- Update
- Delete
- Change the `IsActive` field when creating or updating records

PriceBook2, Product2, and PricebookEntry Relationships

In the API:

- Price books are represented by Pricebook2 records (as of version 8.0, the Pricebook object is no longer available).
- Products are represented by Product2 records (as of version 8.0, the Product object is no longer available).
- Each price book contains zero or more entries (represented by PricebookEntry records) that specify the products that are associated with the price book. A price book entry defines the price for which you sell a product at a particular currency.

These objects are defined only for those orgs that have products enabled as a feature. If the org doesn't have the products feature enabled, the Pricebook2 object doesn't appear in the `describeGlobal()` call, and you can't access it via the API.

If you delete a Pricebook2 while a line item references PricebookEntry in the price book, the line item is unaffected, but the Pricebook2 is archived and unavailable from the API.

For a visual diagram of the relationships between Pricebook2 and other objects, see Product and Schedule Objects.

Price Book Setup

The process of setting up a price book via the API usually means:

1. Load product data into Product2 records (creating one Product2 record for each product that you want to add).

2. For each Product2 record, create a PricebookEntry that links the Product2 record to the standard Pricebook2. Define a standard price for a product at a given currency (if you have multicurrency enabled) before defining a price for that product in the same currency in a custom price book.
3. Create a Pricebook2 record to represent a custom price book.
4. For each Pricebook2 record, creating a PricebookEntry for every Product2 that you want to add, specifying unique properties for each PricebookEntry (such as the `UnitPrice` and `CurrencyIsoCode`) as needed.

Code Sample—Java

```
public void pricebookSample() {
    try {
        //Create a custom pricebook
        Pricebook2 pb = new Pricebook2();
        pb.setName("Custom Pricebok");
        pb.setIsActive(true);
        SaveResult[] saveResults = connection.create(new SObject[]{pb});
        pb.setId(saveResults[0].getId());

        // Create a new product
        Product2 product = new Product2();
        product.setIsActive(true);
        product.setName("Product");
        saveResults = connection.create(new SObject[]{product});
        product.setId(saveResults[0].getId());

        // Add product to standard pricebook
        QueryResult result = connection.query(
            "select Id from Pricebook2 where isStandard=true"
        );
        SObject[] records = result.getRecords();
        String stdPbId = records[0].getId();

        // Create a pricebook entry for standard pricebook
        PricebookEntry pbe = new PricebookEntry();
        pbe.setPricebook2Id(stdPbId);
        pbe.setProduct2Id(product.getId());
        pbe.setIsActive(true);
        pbe.setUnitPrice(100.0);
        saveResults = connection.create(new SObject[]{pbe});

        // Create a pricebook entry for custom pricebook
        pbe = new PricebookEntry();
        pbe.setPricebook2Id(pb.getId());
        pbe.setProduct2Id(product.getId());
        pbe.setIsActive(true);
        pbe.setUnitPrice(100.0);
        saveResults = connection.create(new SObject[]{pbe});
    } catch (ConnectionException ce) {
        ce.printStackTrace();
    }
}
```

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

Pricebook2ChangeEvent (API version 48.0)

Change events are available for the object.

Pricebook2History

History is available for tracked fields of the object.

Product2

Represents a product that your company sells.

This object has several fields that are used only for quantity and revenue schedules (for example, annuities). Schedules are available only for orgs that have enabled the products and schedules features. If these features aren't enabled, the schedule fields don't appear, and you can't query, create, or update the fields.



Note: As of API version 8.0, the Product object is no longer available. Requests that contain Product are refused, and responses don't contain the Product object. Use the Products2 object instead.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

The `ConfigureDuringSale` and `IsSoldOnlyWithOtherProds` fields are available in version 58.0 and later when Industry Automotive or Subscription Management is enabled.

Fields

Field	Details
BillingPolicyId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the related billing policy. This field is available when Subscription Management is enabled. This field is available in API version 55.0 and later. This field is a relationship field. Relationship Name BillingPolicy Relationship Type Lookup Refers To BillingPolicy

Field	Details
CanUseQuantitySchedule	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether the product can have a quantity schedule (<code>true</code>) or not (<code>false</code>). Label is Quantity Scheduling Enabled.
CanUseRevenueSchedule	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether the product can have a revenue schedule (<code>true</code>) or not (<code>false</code>). Label is Revenue Scheduling Enabled.
ConnectionReceivedId	Type reference Properties Filter, Group, Nillable, Sort Description ID of the PartnerNetworkConnection that shared this record with your org. This field is available when Salesforce to Salesforce is enabled.
ConnectionSentId	Type reference Properties Filter, Group, Nillable, Sort Description ID of the PartnerNetworkConnection that this record is shared with. This field is available Salesforce to Salesforce is enabled. In API version 16.0 and later, this value is <code>null</code>. Use PartnerNetworkRecordConnection object to forward records to connections.
ConfigureDuringSale	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Determines whether a user can edit a configuration when creating a bundle order or quote. This field is available in API version 58.0 and later. This field is available when Industries Automotive or Subscription Management is enabled.

Field	Details
	Possible values are: - Allowed—Changes are allowed while adding line items to a bundle; for example, when adding products or editing quantity. - NotAllowed—Changes aren't allowed.
CurrencyIsoCode	Type picklist Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update Description Available only for orgs with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the org.
Description	Type textarea Properties Create, Filter, Nillable, Sort, Update Description A text description of this record. Label is Product Description.
DisplayUrl	Type url Properties Create, Filter, Nillable, Sort, Update Description URL leading to a specific version of a record in the linked external data source.
ExternalDataSourceId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description ID of the related external data source.
ExternalId	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The unique identifier of a record in the linked external data source. For example, <i>ID #123</i>.

Field	Details
Family	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update Description Name of the product family associated with this record. Product families are configured as picklists in the user interface. To obtain a list of valid values, call <code>describeObjects()</code> and process the result for the values associated with the <code>Family</code> field. Label is Product Family.
IsActive	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether this record is active (<code>true</code>) or not (<code>false</code>). Inactive Product2 records are hidden in many areas in the user interface. You can change the <code>IsActive</code> flag on a Product2 object as often as necessary. Label is Active.
IsArchived	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Describes whether the product is archived. The default value is <code>false</code>.
IsDeleted	Type boolean Properties Defaulted on create, Filter Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.
IsSerialized	Type boolean Properties Create, Filter, Group, Sort, Update Description Indicates if a product is a serialized product (<code>true</code>) or not (<code>false</code>). Label is Serialized.

Field	Details
IsSoldOnlyWithOtherProds	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Determines whether the product can be sold independently or only as part of a bundle. This field is available in API version 58.0 and later. This field is available when Industries Automotive or Subscription Management is enabled. The default value is <code>false</code>, which means that the product can be sold independently.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last interacted with this record, directly or indirectly.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last viewed this record or list view. If this value is null, it's possible that the user only accessed this record or list view (<code>LastReferencedDate</code>), but not viewed it.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description Required. Default name of this record. Label is Product Name.
NumberOfQuantityInstallments	Type int Properties Create, Filter, Group, Nillable, Sort, Update Description If the product has a quantity schedule, the number of installments.

Field	Details
NumberOfRevenueInstallments	Type int Properties Create, Filter, Group, Nillable, Sort, Update Description If the product has a revenue schedule, the number of installments.
ProductClass	Type picklist Properties Defaulted on create, Filter, Group, Restricted picklist, Sort Description This field is read-only. Its value is determined by the value of the <code>Type</code> field and whether the product is associated with a <code>ProductAttribute</code> record. It describes whether a product is a bundle, set, or simple product, a variation parent, or a product variation. Possible values are: • <code>Bundle</code>—This product is a parent or component in a product bundle. • <code>Set</code>—This product is included in a product set. • <code>Simple</code>—This product has no variations • <code>VariationParent</code>—This product is a variation parent. It's the base product for one or more product variations and, though it has its own stock-keeping unit (SKU), isn't a sellable entity. Instead, it's the parent of sellable entities—its variations. • <code>Variation</code>—This product is a variation of a parent product. Each variation has its own SKU. When the value of <code>ProductClass = VariationParent</code>, it never changes. The value of <code>ProductClass</code> changes between <code>Simple</code> and <code>Variation</code> when you attach or detach a <code>ProductAttribute</code> record to the product. If you attach a <code>ProductAttribute</code> record to a product, then the product's <code>ProductClass</code> value changes to <code>Variation</code>. Conversely, when you detach all <code>ProductAttribute</code> records from a product, the <code>ProductClass</code> value changes to <code>Simple</code>. The default value is <code>Simple</code>. This field is available in API version 50.0 and later. It was introduced to support of B2B and B2C Commerce implementations.
ProductCode	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description Default product code for this record. Your org defines the product's code-naming pattern.

Field	Details
QuantityInstallmentPeriod	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description If the product has a quantity schedule, the amount of time covered by the schedule.
QuantityScheduleType	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The type of the quantity schedule, if the product has one.
QuantityUnitOfMeasure	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update Description Unit of the product; for example, kilograms, liters, or cases. This field comes with only one value, Each, so consider creating your own. The <code>QuantityUnitOfMeasure</code> field on <code>ProductItem</code> inherits this field's values.
RecalculateTotalPrice	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Changes behavior of <code>OpportunityLineItem</code> calculations when a line item has child schedule rows for the <code>Quantity</code> value. When enabled, if the rollup quantity changes, then the quantity rollup value is multiplied against the sales price to change the total price.
RevenueInstallmentPeriod	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description If the product has a revenue schedule, the time period covered by the schedule.
RevenueScheduleType	Type picklist

Field	Details
	Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The type of the revenue schedule, if the product has one.
StockCheckMethod	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The method for how a product's inventory is checked. Stock checks on parent products are common when bundles are prepackaged and individual child components can't be sold separately. Stock checks on child products are common when bundles aren't prepackaged and must be put together during fulfillment. If bundles aren't prepackaged, child components can usually be sold separately. Possible values are: • <code>Null</code>—Check stock on the product SKU. • <code>DoNotCheck</code>—The stock shouldn't be check. • <code>ParentProduct</code>—If the product is a parent of a bundle, check stock on the parent product. • <code>ChildProducts</code>—If the product is a parent of a bundle, check stock on the child components.
StockKeepingUnit	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The SKU for the product. Use in tandem with or instead of the <code>ProductCode</code> field. For example, you can track the manufacturer's identifying code in the Product Code field and assign the product a SKU when you resell it.
TaxPolicyId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the related tax policy. This field is available when Subscription Management is enabled. This field is available in API version 55.0 and later. This field is a relationship field.

Field	Details
	Relationship Name TaxPolicy Relationship Type Lookup Refers To TaxPolicy
TransferRecordMode	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update Description If serialized, indicates when the serial number is recorded. This field is visible based on field-level security. The value affects the read-only value of the <code>Product2TransferMode</code> field on the <code>ProductTransfer</code> object. Possible values are: • <code>SendAndReceive</code> —The serial number is recorded when sending or receiving. • <code>ReceiveOnly</code> —The serial number is recorded when receiving only.
Type	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort Description The type of product. This field's value affects the read-only value of the <code>ProductClass</code> field on the <code>Product2</code> object. The following mappings define how the <code>Type</code> selection updates the <code>ProductClass</code>. • <code>Base</code>—When <code>Type = Base</code>, then <code>ProductClass = VariationParent</code>. • <code>Null</code>—When <code>Type = Null</code>, then <code>ProductClass = Simple</code> for standalone products. • <code>Null</code>—When <code>Type = Null</code>, then <code>ProductClass = Variation</code> for variation products. • <code>Bundle</code>—When <code>Type = Bundle</code>, then <code>ProductClass = Bundle</code>. • <code>Set</code>—When <code>Type = Set</code>, then <code>ProductClass = Set</code>.  Note: • Revenue Cloud doesn't support products with these specific combinations: <code>Type = Base</code> and <code>ProductClass = VariationParent</code> or <code>Type = Null</code> and <code>ProductClass = Variation</code>.

Field	Details
	- Values <code>Null</code>, <code>Base</code>, <code>Bundle</code>, and <code>Set</code> are available in environments where both Commerce and Revenue Cloud co-exist. - The <code>Type</code> field can only be updated from <code>Null</code> to <code>Bundle</code> for products with a <code>Simple ProductClass</code>. This field is available when Revenue Cloud, B2B Commerce, B2C Commerce, or other clouds with PCM add-on is enabled. This field is available in API version 50.0 and later.
<code>UnitOfMeasureId</code>	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the unit of measure associated with the product. This field is a relationship field. This field is available when Revenue Cloud is enabled. This field is available in API version 63.0 and later. Relationship Name <code>UnitOfMeasure</code> Refers To <code>UnitOfMeasure</code>

Schedule Enabled Flags

When enabling the schedules feature, you can decide whether to enable quantity schedules, revenue schedules, or both. In addition, you can use the API to control quantity and revenue scheduling at the product level via the `CanUseQuantitySchedule` and `CanUseRevenueSchedule` flags. A value of `true` for either flag indicates that the product and any `OpportunityLineItems` can have a schedule of that type. These flags can be set when creating or updating `Product2` records.

Default Schedule Fields

The remaining schedule fields for this object define default schedules. Default schedule values are used to create an `OpportunityLineItemSchedule` when an `OpportunityLineItem` is created for the Product.

The default schedule fields support the following valid values (all fields are also nillable).

Field	Valid Values
<code>RevenueScheduleType</code>	Divide, Repeat
<code>RevenueInstallmentPeriod</code>	Daily, Weekly, Monthly, Quarterly, Yearly
<code>NumberOfRevenueInstallments</code>	Integer from 1 to 150, inclusive.
<code>QuantityScheduleType</code>	Divide, Repeat

Field	Valid Values
QuantityInstallmentPeriod	Daily, Weekly, Monthly, Quarterly, Yearly
NumberOfQuantityInstallments	Integer from 1 to 150, inclusive

When you attempt to set the schedule fields when creating or updating, the API applies cross-field integrity checks. The integrity requirements are:

- If the schedule type is nil, the installment period and number of installments must be nil.
- If the schedule type is set to any value, then the installment period and number of installments must be non-nil.

Any create or update that fails these integrity checks is rejected with an error.

These default schedule fields, `CanUseQuantitySchedule`, and `CanUseRevenueSchedule`, are restricted picklist fields and are available only if the org has the schedules feature enabled.

Usage

Use this object to define the default product information for your org. This object is associated by reference with `Pricebook2` objects via `PricebookEntry` objects. The same product can be represented in different price books as price book entries. In fact, the same product can be represented multiple times (as separate `PricebookEntry` records) in the same price book with different prices or currencies. A product can only have one price for a given currency within the same price book. To be used in custom price books, all standard prices must be added as price book entries to the standard price book.



Note: Note: You can't create lookup fields to `Product2` object, which have **Required** check box set to true or the **Don't Allow Deletion** radio button selected, as the platform would otherwise interpret this and throw an error that you cannot create a master-detail relationship to the object.

You can query the products that have been configured for your org. For example, you can allow your client application to obtain valid product IDs for use when configuring `PricebookEntry` records via the API. Your client application can perform the following tasks on `PricebookEntry` objects:

- Query
- Create for the standard price book or custom price books.
- Update
- Delete
- Change the `IsActive` field when creating or updating records

This object is defined only for those orgs that have products enabled as a feature. If the org doesn't have the products feature, this object doesn't appear in the `describeGlobal` call, and you can't describe or query this object.

If you try to delete a product via the API but there's an opportunity that uses that product, the delete fails. The workaround is to delete the product in the user interface, which gives you an option to archive the product.



Note: On opportunities and opportunity products, the workflow rules, validation rules, and Apex triggers fire when an update to a child opportunity product or schedule causes an update to the parent record. This means your custom application logic is enforced when there are updates to the parent record, ensuring higher data quality and compliance with your organization's business policies.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

Product2ChangeEvent (API version 44.0)

Change events are available for the object.

Product2Feed (API version 18.0)

Feed tracking is available for the object.

Product2History

History is available for tracked fields of the object.

Product2OwnerSharingRule (API version 50.0)

Sharing rules are available for the object.

ProductConsumed

Represents an item from your inventory that was used to complete a work order or work order line item in field service.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.



Note: To create products consumed, you need Read permission on product items.



Note: To delete or undelete product consumed for non-serialized products, you need Edit, Create, and Read permission on product consumed. For product consumed records that lookup to serialized products, you need Modify All Data or Modify All Records permission on product consumed.

Fields

Field Name	Details
Description	Type textarea Properties Create, Nillable, Update Description Notes and context about the product consumed.
IsConsumed	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update

Field Name	Details
	Description Indicates that a product consumed has been processed if the Product2 it refers to has IsSerialized=true selected. The default is false.
IsLocked	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Indicates whether the product consumed record is locked or not. The default value is <code>false</code>.
IsProduct2Serialized	Type boolean Properties Create, Filter, Nillable, Sort, Update Description Indicates if a product is a serialized product. The default is false.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the product consumed was last modified. Its label in the user interface is Last Modified Date.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the product consumed was last viewed.
MayEdit	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Indicates whether the product consumed record can be edited or not. The default value is <code>false</code>.

Field Name	Details
PricebookEntryId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description Price book associated with the product consumed. If the work order and the product item's associated product are related to the same price book, the Price Book Entry auto-populates based on the product item.
Product2Id	Type reference Properties Filter, Group, Nillable, Sort Description Product associated with the product consumed.
ProductConsumedNumber	Type string Properties Autonumber, Defaulted on create, Filter, Sort Description (Read Only) Auto-generated number identifying the product consumed.
ProductItemId	Type reference Properties Create, Filter, Group, Nillable, Sort Description Product item associated with the product consumed. Creating a product consumed record subtracts the quantity consumed from the linked product item's quantity.
ProductName	Type string Properties Filter, Group, Nillable, Sort Description Name for the product consumed.
QuantityConsumed	Type double

Field Name	Details
	Properties Create, Filter, Sort, Update Description The quantity of products consumed.
QuantityUnitOfMeasure	Type picklist Properties Filter, Group, Nillable, Sort Description Units of the consumed item; for example, kilograms or liters. Quantity Unit of Measure picklist values are inherited from the Quantity Unit of Measure field on products.
TotalPrice	Type currency Properties Filter, Nillable, Sort Description The total price paid for the product items.
UnitPrice	Type currency Properties Create, Filter, Nillable, Sort, Update Description The price per unit of the product consumed.
WorkOrderId	Type reference Properties Create, Filter, Group, Sort Description Work order that the product was consumed for.
WorkOrderLineItemId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description Work order line item that the product was consumed for.

Usage

When a product is consumed during the completion of a work order, create a product consumed record to track its consumption. You can add products consumed to work orders or work order line items. Track product consumption at the line item level if you want to know which products were used for each line item's tasks.

The way you use products consumed depends on how closely you want to track the state of your inventory in Salesforce. If you want to track the entire lifecycle of items in your inventory, including their storage, transfer, and consumption, link your products consumed records to product items. This approach ensures that your inventory numbers auto-update to reflect the consumption of products from your inventory. If you want to track product consumption only, however, specify a Price Book Entry on each product consumed record and leave the Product Item field blank.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ProductConsumedChangeEvent (API version 48.0)

Change events are available for the object.

ProductConsumedFeed

Feed tracking is available for the object.

ProductConsumedHistory

History is available for tracked fields of the object.

ProductConsumedState

Represents the status of an item from your inventory that was used to complete a work order or work order line item in Field Service. This object is available in API version 57.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.



Note: To create products consumed, you need Read permission on product items.



Note: To delete products consumed, you need Edit, Create, and Read permission on product items.

Fields

Field	Details
ConsumedState	Type picklist Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update

Field	Details
	Description The new state of the serialized product after it's consumed
IsLocked	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Indicates whether the product consumed record is locked or not. The default value is <code>false</code> .
MayEdit	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Indicates whether the product consumed record can be edited or not. The default value is <code>false</code> .
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description Name of the product.
ProductConsumedId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The Product Consumed that is used for consuming the serialized products Relationship Name Owner Relationship Type Lookup
SerializedProductId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update

Field	Details
	Description The serialized product being consumed.
	Relationship Name Owner
	Relationship Type Lookup

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as ProductConsumedState.

Product Consumed State History

History is available for tracked fields of the object.

ProductItem

Represents the stock of a particular product at a particular location in field service, such as all bolts stored in your main warehouse.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field Name	Details
LastReferencedDate	Type dateTime
	Properties Filter, Nillable, Sort
	Description The date when the product item was last modified. Its label in the user interface is Last Modified Date.
LastViewedDate	Type dateTime
	Properties Filter, Nillable, Sort

Field Name	Details
	Description The date when the product item was last viewed.
LocationId	Type reference Properties Create, Filter, Group, Sort Description Location associated with the product item. This usually indicates where the product item is stored. This is a relationship field. Relationship Name Location Relationship Type Lookup Refers To Location
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The product item's owner. This is a polymorphic relationship field. Relationship Name Owner Relationship Type Lookup Refers To Group, User
Product2Id	Type reference Properties Create, Filter, Group, Sort Description Product associated with the product item, which represents the type of product in your inventory. This is a relationship field.

Field Name	Details
	Relationship Name Product2 Relationship Type Lookup Refers To Product2
ProductItemNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description (Read Only) Auto-generated number identifying the product item.
ProductName	Type string Properties Filter, Group, Nillable, Sort Description A name for the product item. Try to select a name that indicates what is being stored where; for example, Batteries in Warehouse A.
QuantityOnHand	Type double Properties Create, Filter, Sort, Update Description The quantity at the location. If you want to add a serial number, this value must be 1.
QuantityUnitOfMeasure	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update Description Units of the product item; for example, kilograms or liters. Quantity Unit of Measure picklist values are inherited from the Quantity Unit of Measure field on products.
SerialNumber	Type string Properties Create, Filter, Group, Nillable, Sort, Update

Field Name	Details
	Description A unique number for identification purposes. If you want to enter a serial number, the Quantity on Hand must be 1.

Usage

Each product item is associated with a product and a location in Salesforce. If a product is stored at multiple locations, the product will be tracked in a different product item for each location.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ProductItemChangeEvent (API version 48.0)

Change events are available for the object.

ProductItemFeed

Feed tracking is available for the object.

ProductItemHistory

History is available for tracked fields of the object.

ProductItemOwnerSharingRule

Sharing rules are available for the object.

ProductItemShare

Sharing is available for the object.

ProductItemTransaction

Represents an action taken on a product item in field service. Product item transactions are auto-generated records that help you track when a product item is replenished, consumed, or adjusted.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `undelete()`, `upsert()`

Special Access Rules

- Field Service must be enabled.
- Only users with Modify All Data or Modify All Records permissions can delete this object.

Fields

Field Name	Details
Description	Type textarea Properties Create, Nillable, Update Description A description of the transaction. The description is blank when the transaction record is created, but can be updated.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last interacted with this record, directly or indirectly.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last viewed this record or list view. If this value is null, it's possible that the user only accessed this record or list view (LastReferencedDate), but not viewed it.
ProductItemId	Type reference Properties Create, Filter, Group, Sort Description The associated product item. This is a relationship field. Relationship Name ProductItem Relationship Type Lookup Refers To ProductItem

Field Name	Details
ProductItemTransactionNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description (Read Only) Auto-generated number identifying the product item transaction.
Quantity	Type double Properties Create, Filter, Sort Description The quantity of the product item involved in the transaction. If inventory was consumed, the quantity is negative.
RelatedRecordId	Type reference Properties Filter, Group, Nillable, Sort Description (Read Only) The product consumed or product transfer related to the action. If the action wasn't related to consumption or transfer, the related record is blank. This is a polymorphic relationship field. Relationship Name RelatedRecord Relationship Type Lookup Refers To ProductTransfer, Visit
TransactionType	Type picklist Properties Create, Filter, Group, Restricted picklist, Sort Description The action that the transaction tracks. • Replenished: When a part is stocked at a location. A Replenished transaction is created when a product item is created. • Consumed: When parts are consumed to complete a work order. A Consumed transaction is created when a record is added to the Products Consumed related list on a work order or work order line item.

Field Name	Details
	- Adjusted: When there's a discrepancy or a change in consumption. An Adjusted transaction is created when a product item's Quantity on Hand is edited, a product consumed is updated or delete, or a product transfer is deleted. - Transferred: When parts are transferred between locations.

Associated Objects

This object has these associated objects. Unless noted, they're available in the same API version as this object.

ProductItemTransactionChangeEvent

Change events are available for the object.

ProductItemTransactionFeed

Feed tracking is available for the object.

ProductItemTransactionHistory

History is available for tracked fields of the object.

ProductRequest

Represents an order for a part or parts in field service.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Authenticated external users can create and update ProductRequest objects.

Fields

Field Name	Details
AccountId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The account associated with the product request. This is a relationship field. Relationship Name Account

Field Name	Details
	Relationship Type Lookup
	Refers To Account
CaseId	Type reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The case associated with the product request. This is a relationship field.
	Relationship Name Case
	Relationship Type Lookup
	Refers To Case
CurrencyIsoCode	Type picklist
	Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description Available only if the multicurrency feature is enabled. Contains the ISO code for any currency allowed by the organization. The label in the user interface is <code>Currency ISO Code</code> .
Description	Type textarea
	Properties Create, Nillable, Update
	Description A text field for details not recorded in the provided fields.
DestinationLocationId	Type reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description Where the product is delivered.

Field Name	Details
	This is a relationship field. Relationship Name DestinationLocation Relationship Type Lookup Refers To Location
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the product request was last modified. Its label in the user interface is Last Modified Date.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the product request was last viewed.
NeedByDate	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description Date the product must be delivered by.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The owner of the shipment. This is a polymorphic relationship field. Relationship Name Owner Relationship Type Lookup

Field Name	Details
	Refers To Group, User
ProductRequestNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description An auto-assigned number that identifies the shipment.
ShipToAddress	Type address Properties Filter, Nillable Description The address that the product is to be delivered to.
ShipToCity	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The city that the product is to be delivered to.
ShipToCountry	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The country that the product is to be delivered to.
ShipToCountryCode	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update Description A two letter uppercase country code conforming to the ISO 3166-1 alpha-2 standard.
ShipToGeocodeAccuracy	Type picklist

Field Name	Details
	Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The accuracy of the geocode for the shipping address.
ShipToLatitude	Type double Properties Create, Filter, Nillable, Sort, Update Description The latitude of the location where the product is to be delivered to.
ShipToLongitude	Type double Properties Create, Filter, Nillable, Sort, Update Description The longitude of the location where the product is to be delivered to.
ShipToPostalCode	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The postal code of the address where the product is to be delivered to.
ShipToState	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The name of the state where the product is to be delivered to.
ShipToStateCode	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update Description A two letter uppercase state code conforming to the ISO 3166-1 alpha-2 standard.

Field Name	Details
ShipToStreet	Type textarea Properties Create, Filter, Group, Nillable, Sort, Update Description The street address where the product is to be delivered to.
ShipmentType	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description The type of shipment. The picklist includes the following values by default: • None • Rush • Overnight • Next Business Day • Pick Up
SourceLocationId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The location the product is shipped from. This is a relationship field. Relationship Name SourceLocation Relationship Type Lookup Refers To Location
Status	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description Status of the product transfer.

Field Name	Details
WorkOrderId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The work order that the product request is related to. This is a relationship field. Relationship Name WorkOrder Relationship Type Lookup Refers To WorkOrder
WorkOrderLineItemId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The work order line item that the product request is related to. This is a relationship field. Relationship Name WorkOrderLineItem Relationship Type Lookup Refers To WorkOrderLineItem

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ProductRequestChangeEvent (API version 48.0)

Change events are available for the object.

ProductRequestFeed

Feed tracking is available for the object.

ProductRequestHistory

History is available for tracked fields of the object.

ProductRequestOwnerSharingRule

Sharing rules are available for the object.

ProductRequestShare

Sharing is available for the object.

ProductRequestLineItem

Represents a request for a part in field service. Product request line items are components of product requests.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled. You can't use product request line item as a master in an master detail relationship (through a custom field) with a custom object with data.

Fields

Field Name	Details
AccountId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The account associated with the product request line item. This is a relationship field. Relationship Name Account Relationship Type Lookup Refers To Account
CareProgramEnrolleeId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the care program enrollee associated with the product request line item. This field is available from API version 49.0 and later. This is a relationship field. Relationship Name CareProgramEnrollee

Field Name	Details
	Relationship Type Lookup
	Refers To CareProgramEnrollee
CaseId	Type reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The case associated with the product request line item. This is a relationship field.
	Relationship Name Case
	Relationship Type Lookup
	Refers To Case
Description	Type textarea
	Properties Create, Nillable, Update
	Description Details not recorded in the provided fields.
DestinationLocationId	Type reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description Where the product is delivered. This is a relationship field.
	Relationship Name DestinationLocation
	Relationship Type Lookup
	Refers To Location

Field Name	Details
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last interacted with this record, directly or indirectly.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last viewed this record or list view. If this value is null, it's possible that the user only accessed this record or list view (LastReferencedDate), but not viewed it.
NeedByDate	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description Date the product must be delivered by.
ParentId	Type reference Properties Create, Filter, Group, Sort Description The product request that the line item belongs to. This is a relationship field. Relationship Name Parent Relationship Type Lookup Refers To ProductRequest
Product2Id	Type reference

Field Name	Details
	Properties Create, Filter, Group, Sort, Update Description The product associated with the product request line item. This is a relationship field. Relationship Name Product2 Relationship Type Lookup Refers To Product2
ProductRequestLineItemNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description (Read Only) An auto-assigned number that identifies the product request line item.
QuantityRequested	Type double Properties Create, Filter, Sort, Update Description The amount requested.
QuantityUnitOfMeasure	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update Description Units of the requested product; for example, grams, liters, or units. The picklist values can be customized.
ShipToAddress	Type address Properties Filter, Nillable Description The physical address where the product is needed.

Field Name	Details
ShipToCity	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The city of the address where the product is needed.
ShipToCountry	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The country of the address where the product is needed.
ShipToGeocodeAccuracy	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Accuracy level of the geocode for the address where the product is needed. See Compound Field Considerations and Limitations for details on geolocation compound fields. This field is available in the API only.
ShipToLatitude	Type double Properties Create, Filter, Nillable, Sort, Update Description Used with Longitude to specify the precise geolocation of the address where the product is needed. Acceptable values are numbers between –90 and 90 with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields. This field is available in the API only.
ShipToLongitude	Type double Properties Create, Filter, Nillable, Sort, Update Description Used with Latitude to specify the precise geolocation of the address where the product is needed. Acceptable values are numbers between –180 and 180 with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields. This field is available in the API only.

Field Name	Details
ShipToPostalCode	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The postal code of the address where the product is needed.
ShipToState	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The state of the address where the product is needed.
ShipToStreet	Type textarea Properties Create, Filter, Group, Nillable, Sort, Update Description The street of the address where the product is needed.
ShipmentType	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description The type of shipment. The picklist includes the following values, which can be customized: • Rush • Overnight • Next Business Day • Pick Up
SourceLocationId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description Where the product is at the time of the request. This is a relationship field.

Field Name	Details
	Relationship Name SourceLocation Relationship Type Lookup Refers To Location
Status	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description The status of the shipment. The picklist includes the following values, which can be customized: • Draft • Submitted • Received
WorkOrderId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The work order for which the product is needed. This is a relationship field. Relationship Name WorkOrder Relationship Type Lookup Refers To WorkOrder
WorkOrderLineItemId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The work order line item for which the product is needed. This is a relationship field.

Field Name	Details
	Relationship Name WorkOrderLineItem
	Relationship Type Lookup
	Refers To WorkOrderLineItem

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ProductRequestLineItemChangeEvent (API version 48.0)

Change events are available for the object.

ProductRequestLineItemFeed

Feed tracking is available for the object.

ProductRequestLineItemHistory

History is available for tracked fields of the object.

ProductRequired

Represents a product that is needed to complete a work order or work order line item in field service.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field Name	Details
LastReferencedDate	Type dateTime
	Properties Filter, Nillable, Sort
	Description The date when the product required was last modified. Its label in the user interface is Last Modified Date.

Field Name	Details
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the product required was last viewed.
ParentRecordId	Type reference Properties Create, Filter, Group, Sort Description The work order or work order line item that the product is required for. This is a polymorphic relationship field. Relationship Name ParentRecord Relationship Type Lookup Refers To Visit, WorkOrder, WorkOrderLineItem, WorkType
ParentRecordType	Type string Properties Filter, Group, Nillable, Sort Description Indicates whether the parent record is a work order or a work order line item.
Product2Id	Type reference Properties Create, Filter, Group, Sort, Update Description The required product. This is a relationship field. Relationship Name Product2 Relationship Type Lookup Refers To Product2

Field Name	Details
ProductName	Type string Properties Filter, Group, Nillable, Sort Description The name of the product required.
ProductRequiredNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description (Read only) Auto-generated number identifying the product required.
QuantityRequired	Type double Properties Create, Filter, Nillable, Sort, Update Description Quantity required of the product.
QuantityUnitOfMeasure	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update Description Units of the required product; for example, kilograms or liters. Quantity Unit of Measure picklist values are inherited from the Quantity Unit of Measure field on products.

Usage

Required products can be added to work types, work orders, and work order line items to ensure that the assigned service resource arrives with the right equipment.

Adding required products to work types saves you time and keeps your business processes consistent. Work orders and work order line items inherit their work type's required products. For example, if all light bulb replacement jobs require a ladder and a light bulb, add the ladder and light bulb as required products to your Light Bulb Replacement work type. When it's time to create a work order for a customer's light bulb replacement, applying that work type to the work order adds the required products.

Associated Objects

This object has the following associated objects. Unless noted, they are available in the same API version as this object.

ProductRequiredChangeEvent

Change events are available for the object.

ProductRequiredFeed

Feed tracking is available for the object.

ProductRequiredHistory

History is available for tracked fields of the object.

ProductServiceCampaign

Represents a set of activities to be performed on a product service campaign asset, such as a product recall for safety issues or product defects. This object is available in API version 51.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field	Details
Description	Type textarea Properties Create, Nillable, Update Description Description of the product service campaign.
EndDate	Type date Properties Create, Filter, Group, Nillable, Sort, Update Description The date on which the product service campaign ends.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date and time that the asset was last modified. The UI label is Last Modified Date.

Field	Details
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date and time that the asset was last viewed.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The product service campaign's owner. By default, the product service campaign owner is the user who created the product service campaign record. The UI label is Product Service Campaign Owner.
Priority	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update Description The priority of the product service campaign. Possible values are: • Critical • High • Low • Medium
Product2Id	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description ID of the Product2 associated with this campaign. The UI label is Product.
ProductServiceCampaignName	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The name of the product service campaign.

Field	Details
StartDate	Type date Properties Create, Filter, Group, Sort, Update Description The date on which the product service campaign starts.
Status	Type picklist Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The status of the product service campaign. The picklist includes the following values, which can be customized: • New—Product service campaign created, but there hasn't yet been any activity. • In Progress—Product service campaign has begun. • On Hold—Work is paused. • Completed—Work is complete. • Cannot Complete—Work couldn't be completed. • Closed—All work and associated activity is complete. • Canceled—Work is canceled, typically before any work began.
StatusCategory	Type picklist Properties Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort Description The category that each <code>Status</code> value falls into. The <code>StatusCategory</code> field has eight default values: seven values that are identical to the default <code>Status</code> values, and <code>None</code> for statuses without a status category. If you create custom <code>Status</code> values, you must indicate which category it belongs to. For example, if you create a <i>Waiting for Response</i> value, add it the <i>On Hold</i> category. To learn which processes reference <code>StatusCategory</code>, see How are Status Categories Used?
Type	Type picklist Properties Create, Filter, Group, Sort, Update

Field	Details
	Description The type of the product service campaign. The picklist includes the following values, which can be customized: • Modification—The asset requires an on-site alteration. • Recall—The asset must be returned to the manufacturer for modification or upgrade. • Service—The asset needs to be serviced. • Upgrade—The asset needs updating.
WorkTypeId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The work type associated with the product service campaign. A customer uses this field as a guide when setting work type for work orders for the product service campaign. Duration, Duration Type, and required skills.

Associated Objects

This object has the following associated objects. Unless noted, they are available in the same API version as this object.

ProductServiceCampaignFeed

Feed tracking is available for the object.

ProductServiceCampaignHistory

History is available for tracked fields of the object.

ProductServiceCampaignOwnerSharingRule

Sharing rules are available for the object.

ProductServiceCampaignShare

Sharing is available for the object.

ProductServiceCampaignItem

Represents a product service campaign's asset. This object is available in API version 51.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field	Details
AssetId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The asset associated with the product service campaign. Must be present if Product2Id is not present.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date and time that the asset was last modified. Its UI label is Last Modified Date.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date and time that the asset was last viewed.
Product2Id	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the Product2 associated with this campaign. The UI label is Product. Must be present if AssetID is not present.
ProductServiceCampaignId	Type reference Properties Create, Filter, Group, Sort Description Required. The item's parent product service campaign record.
ProductServiceCampaignItemNumber	Type string

Field	Details
	Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description The ID of the product service campaign item.
Status	Type picklist Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The status of the product service campaign item. The picklist includes the following values, which can be customized: • New—Product service campaign item created, but there hasn't yet been any activity. • In Progress—Product service campaign item has begun. • On Hold—Product service campaign item is paused. • Completed—Product service campaign item is complete. • Cannot Complete—Product service campaign item couldn't be completed. • Closed—All product service campaign item and associated activity is complete. • Canceled—Product service campaign item is canceled, typically before any work began.
StatusCategory	Type picklist Properties Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort Description The category that each <code>Status</code> value falls into. The <code>StatusCategory</code> field has eight default values: seven values that are identical to the default <code>Status</code> values, and <code>None</code> for statuses without a status category. If you create custom <code>Status</code> values, you must indicate which category it belongs to. For example, if you create a <i>Waiting for Response</i> value, add it to the <i>On Hold</i> category. To learn which processes reference <code>StatusCategory</code>, see How are Status Categories Used?

Associated Objects

This object has the following associated objects. Unless noted, they are available in the same API version as this object.

ProductServiceCampaignItemFeed

Feed tracking is available for the object.

ProductServiceCampaignItemHistory

History is available for tracked fields of the object.

ProductServiceCampaignItemOwnerSharingRule

Sharing rules are available for the object.

ProductServiceCampaignItemShare

Sharing is available for the object.

ProductServiceCampaignItemStatus

Represents a status for a product service campaign item in field service. This object is available in API version 51.0 and later.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

Field Service must be enabled.

Fields

Field	Details
<code>ApiName</code>	Type string Properties Filter, Group, idLookup, Sort Description The API name of the status value.
<code>IsDefault</code>	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Indicates that the status value is the default status on product service campaign items when <code>true</code> . Only one status value can be the default.
<code>MasterLabel</code>	Type string Properties Filter, Group, Nillable, Sort Description The label for the picklist value in the UI.
<code>SortOrder</code>	Type int

Field	Details
	Properties Filter, Group, Nillable, Sort Description The value's position in the dropdown list in the UI.
StatusCode	Type picklist Properties Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort Description The status category that the value corresponds to. The Status Category field has seven values that are identical to the default Status values.

Usage

The Status field on product service campaign items comes with the following values:

- New—Product service campaign item created, but there hasn't been any activity.
- In Progress—Work has begun.
- On Hold—Work is paused.
- Completed—Work is complete.
- Cannot Complete—Work couldn't be completed.
- Closed—All work and associated activity is complete.
- Canceled—Work is canceled, typically before any work began.

The ProductServiceCampaignItemStatus object corresponds to the Status field. Adding a value to the Status field—for example, Canceled By Supplier—creates a product service campaign item status record, and vice versa.



Note: Product service campaign items also come with a Status Category field whose values are identical to the default status values. If you create custom status values, you must indicate which category it belongs to. For example, if you create a *Customer Absent* value, add it to the *Cannot Complete* category. To learn which processes reference StatusCategory, see [How are Status Categories Used?](#)

ProductServiceCampaignStatus

Represents a status for a product service campaign in field service. This object is available in API version 51.0 and later.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

Field Service must be enabled.

Fields

Field	Details
ApiName	Type string Properties Filter, Group, idLookup, Sort Description The API name of the status value.
IsDefault	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Indicates that the status value is the default status on product service campaigns when <code>true</code>. Only one status value can be the default.
MasterLabel	Type string Properties Filter, Group, Nillable, Sort Description The label for the picklist value in the UI.
SortOrder	Type int Properties Filter, Group, Nillable, Sort Description The value's position in the dropdown list in the UI.
StatusCode	Type picklist Properties Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort Description The status category that the value corresponds to. The Status Category field has seven values that are identical to the default Status values.

Usage

The Status field on product service campaigns comes with the following values:

- New—Product service campaign created, but there hasn't been any activity.
- In Progress—Work has begun.
- On Hold—Work is paused.
- Completed—Work is complete.
- Cannot Complete—Work couldn't be completed.
- Closed—All work and associated activity is complete.
- Canceled—Work is canceled, typically before any work began.

The `ProductServiceCampaignStatus` object corresponds to the `Status` field. Adding a value to the `Status` field—for example, `Canceled By Supplier`—creates a product service campaign status record, and vice versa.

 **Note:** Product service campaigns also come with a `Status Category` field whose values are identical to the default status values. If you create custom status values, you must indicate which category it belongs to. For example, if you create a *Customer Absent* value, add it to the *Cannot Complete* category. To learn which processes reference `StatusCategory`, see [How are Status Categories Used?](#)

ProductTransfer

Represents the transfer of inventory between locations in field service.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field Name	Details
Description	Type textarea Properties Create, Nillable, Update Description Details not recorded in the provided fields.
DestinationLocationId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The place the product is to be delivered. This is a relationship field.

Field Name	Details
	Relationship Name DestinationLocation Relationship Type Lookup Refers To Location
ExpectedPickupDate	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description Date the product is expected to be picked up.
IsReceived	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Checkbox identifying that the product was received.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the product request was last modified. Its label in the user interface is Last Modified Date.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the product request was last viewed.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update

Field Name	Details
	Description Owner of the product transfer. This is a polymorphic relationship field. Relationship Name Owner Relationship Type Lookup Refers To Group, User
Product2Id	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description Lookup field for the product associated with the product transfer. This is a relationship field. Relationship Name Product2 Relationship Type Lookup Refers To Product2
Product2TransferRecordMode	Type reference Properties Filter, Group, Nillable, Sort Description If serialized, indicates when the serial number is recorded. It is visible on the product transfer as a read-only field depending on the field-level security. Possible values are: • <code>SendAndReceive</code> —The serial number is recorded when sending or receiving. • <code>ReceiveOnly</code> —The serial number is recorded when receiving only. Relationship Name Product2.TransferRecordMode Relationship Type Lookup Refers To Product2.TransferRecordMode

Field Name	Details
ProductRequestId	Type reference Properties Filter, Group, Nillable, Sort Description Lookup field for the product request associated with the product transfer. This is a relationship field. Relationship Name ProductRequest Relationship Type Lookup Refers To ProductRequest
ProductRequestLineItemId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description Lookup field for the product request line item associated with the product transfer. This is a relationship field. Relationship Name ProductRequestLineItem Relationship Type Lookup Refers To ProductRequestLineItem
ProductTransferNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description An auto-assigned number that identifies the product transfer.
QuantityReceived	Type double Properties Create, Filter, Nillable, Sort, Update Description Amount of product received at the destination location.

Field Name	Details
QuantitySent	Type double Properties Create, Filter, Sort, Update Description Amount of product sent from the source location.
QuantityUnitOfMeasure	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update Description The units of the product, for example grams, liters, or units.
ReceivedById	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description Lookup field for the contact who received the product at the destination location. This is a polymorphic relationship field. Relationship Name ReceivedBy Relationship Type Lookup Refers To Group, User
ReturnOrderId	Type reference Properties Filter, Group, Nillable, Sort Description The return order associated with the product transfer. This is a relationship field. Relationship Name ReturnOrder Relationship Type Lookup Refers To ReturnOrder

Field Name	Details
ReturnOrderLineItemId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The return order line item associated with the product transfer. This is a relationship field. Relationship Name ReturnOrderLineItem Relationship Type Lookup Refers To ReturnOrderLineItem
ShipmentExpectedDeliveryDate	Type dateTime Properties Filter, Nillable, Sort Description Lookup field for the shipment related to the product transfer.
ShipmentId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description Lookup field for the shipment related to the product transfer. This is a relationship field. Relationship Name Shipment Relationship Type Lookup Refers To Shipment
ShipmentStatus	Type picklist Properties Defaulted on create, Filter, Group, Nillable, Sort Description Lookup field for the shipment related to the product transfer.

Field Name	Details
ShipmentTrackingNumber	Type string Properties Filter, Group, Nillable, Sort Description Lookup field for the shipment related to the product transfer.
ShipmentTrackingUrl	Type url Properties Filter, Group, Nillable, Sort Description Lookup field for the shipment related to the product transfer.
SourceLocationId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description Lookup field for the source location related to the product transfer. This is a relationship field. Relationship Name SourceLocation Relationship Type Lookup Refers To Location
SourceProductItemId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description Lookup field for the product item related to the product transfer. Relationship Name SourceProductItem Relationship Type Lookup Refers To ProductItem

Field Name	Details
Status	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description Status of the product transfer.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ProductTransferChangeEvent (API version 48.0)

Change events are available for the object.

ProductTransferFeed

Feed tracking is available for the object.

ProductTransferHistory

History is available for tracked fields of the object.

ProductTransferOwnerSharingRule

Sharing rules are available for the object.

ProductTransferShare

Sharing is available for the object.

ProductWarrantyTerm

Defines the relationship between a product or product family and warranty term. This object is available in API version 50.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
CoveredProductFamily	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update Description The product family that the warranty term applies to.

Field	Details
CoveredProductId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the product that the warranty term applies to.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the product warranty term was last modified. Its label in the user interface is <code>Last Modified Date</code>.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the product warranty term was last viewed.
ProductWarrantyTermNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description The identifier for this product warranty term.
WarrantyTermId	Type reference Properties Create, Filter, Group, Sort Description The ID of the warranty term.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ProductWarrantyTermChangeEvent (API version 62.0)

Change events are available for the object.

RecordsetFilterCriteria

Represents a set of filters that can be used to match service appointments or assets based on your criteria fields. For example, you can create recordset filter criteria so that only service appointments that satisfy the filter criteria are matched to the filtered shifts, and likewise only maintenance work rules that satisfy your criteria are matched to assets. This object is available in API version 50.0 and later. Assets and maintenance work rules are available in API version 52.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field	Details
Description	Type textarea Properties Create, Nillable, Update Description The description of the recordset filter criteria.
FilteredObject	Type picklist Properties Create, Filter, Group, Restricted picklist, Sort Description The object used to define the filter criteria. Available in API version 52.0 or later. Possible values are: • Asset • ServiceAppointment
IsActive	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update

Field	Details
	Description Indicates whether the recordset filter criteria is associated with shifts or maintenance work rules (<code>true</code>) or not (<code>false</code>).
<code>LastReferencedDate</code>	Type dateTime Properties Filter, Nillable, Sort Description The date when the recordset filter criteria was last referenced.
<code>LastViewedDate</code>	Type dateTime Properties Filter, Nillable, Sort Description The date when the recordset filter criteria was last viewed.
<code>LogicalOperator</code>	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Defines the logic to evaluate multiple recordset filter criteria rules. Available in API version 53.0 and later. Possible values are: • AND • OR
<code>Name</code>	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The name of the recordset filter criteria.
<code>OwnerId</code>	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update

Field	Details
	Description The owner of the recordset filter criteria.
SourceObject	Type picklist Properties Create, Filter, Group, Restricted picklist, Sort Description The source object that the filtered criteria are applied to. Shifts and maintenance work rules are available in API version 52.0 and later. Appointment bundle objects are available in API version 53.0 and later. Possible values are: • <code>ApptBundleAggrPolicy</code>—Appointment Bundle Aggregation Policy • <code>ApptBundleConfig</code>—Appointment Bundle Config • <code>Shift</code> • <code>ContractLineOutcome</code> • <code>MaintenanceWorkRule</code>
Usage Rate Field	Type picklist Properties Create, Filter, Group, Restricted picklist, Sort Description Stores the daily usage rate of the asset. The unit for the usage rate must be per day.
Usage Rate Unit	Type picklist Properties Create, Filter, Group, Restricted picklist, Sort Description Defines the rate for Usage Rate Field. Possible values are: • <code>DAYS</code>

Usage

Let's say an employee is open to working a 9 am to 5 pm shift on a Sunday but only for emergency appointments. In this case, the `SourceObject` is `Shift` and the `FilteredObject` is `ServiceAppointment`. The service appointments available for that shift are filtered for emergency appointments using the `RecordsetFilterCriteriaRule` object.

`RecordSetFilterCriteria` isn't available for report types.

Associated Objects

This object has the following associated objects. Unless noted, they are available in the same API version as this object.

RecordsetFilterCriteriaFeed

Feed tracking is available for the object.

RecordsetFilterCriteriaHistory

History is available for tracked fields of the object.

RecordsetFilterCriteriaOwnerSharingRule

Sharing rules are available for the object.

RecordsetFilterCriteriaShare

Sharing is available for the object.

RecordsetFilterCriteriaRule

Represents a rule using fields from the designated source object to create filters on the filtered, or target, object. RecordsetFilterCriteriaRule is associated with the RecordsetFilterCriteria object. This object is available in API version 50.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field	Details
CriteriaField	Type picklist Properties Create, Filter, Group, Restricted picklist, Sort, Update Description The field the filter rule is applied to. Asset fields are available in API version 52.0 and later. Possible values are derived from the source object's standard and custom fields. Possible standard source objects are <code>Asset</code> and <code>ServiceAppointment</code>. The format is, for example, <code>Asset.PricingSource</code> or <code>ServiceAppointment.GroupAppointmentAccessType</code>. All standard and custom fields are allowed except those with these field types: • <code>encryptedstring</code> • <code>multipicklist</code> • <code>textarea</code> • <code>url</code>

Field	Details
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the recordset filter criteria rule was last referenced.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the recordset filter criteria rule was last viewed.
NextOccurence	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description This field's value is compared to the Usage Field to determine if the rule is true. Possible values are derived from the source object's standard and custom fields. Possible standard source objects are <code>Asset</code> and <code>ServiceAppointment</code>. The format is, for example, <code>Asset.PricingSource</code> or <code>ServiceAppointment.GroupAppointmentAccessType</code>. All standard and custom fields are allowed except those with these field types: • <code>encryptedstring</code> • <code>multipicklist</code> • <code>textarea</code> • <code>url</code>
Operator	Type picklist Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update Description The relational operator between <code>CriteriaField</code> and <code>Value</code>. Available in API version 52.0 or later. Possible values are: • <code>Equals—Default</code> • <code>GreaterOrEqual</code> • <code>GreaterThan</code>

Field	Details
	- LessOrEqual - LessThan
RecordsetFilterCriteriaId	Type reference Properties Create, Filter, Group, Sort Description The ID of the RecordsetFilterCriteria record to associate this rule with.
RecordsetFilterCriteriaRuleNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description The automatically assigned number of the recordset filter criteria rule.
Type	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The type of criteria rule. Possible values are: - Criteria—Default - Usage - UsageCounter— Usage (Counter) - UsageDuration— Usage (Duration)
Value	Type textarea Properties Create, Nillable, Update Description The expected value of CriteriaField applied to the filter rule.
Usage Rate Field	Type picklist Properties Create, Filter, Group, Restricted picklist, Sort

Field	Details
	Description Stores the daily usage rate of the asset. The unit for the usage rate must be per day. Possible values are derived from the source object's standard and custom fields. Possible standard source objects are <code>Asset</code> and <code>ServiceAppointment</code> . The format is, for example, <code>Asset.PricingSource</code> or <code>ServiceAppointment.GroupAppointmentAccessType</code> .
Usage Rate Unit	Type picklist Properties Create, Filter, Group, Restricted picklist, Sort Description Defines the rate for Usage Rate Field. Possible values are: DAYS

Usage

If you want to create a filter rule for service appointments with a dispatched status, set `CriteriaField` to `ServiceAppointment.Status` and `Value` to `Dispatched`. Then add the ID from a `RecordsetFilterCriteria` record to `RecordsetFilterCriteriaId` to associate this rule with a filter criteria for shifts.

RecordsetFltrCritMonitor

Monitors whether the value of an asset attribute is within the threshold of a recordset filter criteria (RFC). You can monitor one or more RFCs for an `Asset`. This object is available in API version 57.0 and later.

Supported Calls

```
create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(),
retrieve(), search(), undelete(), update(), upsert()
```

Special Access Rules

Field Service must be enabled.

Fields

Field	Details
AssetId	Type reference Properties Create, Filter, Group, Sort, Update

Field	Details
	Description The ID of the asset to link the RFC to. This field is a relationship field. Relationship Name Asset Relationship Type Lookup Refers To Asset
Description	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The description of the RFC associated with the recordset filter criteria monitor.
IsWithinThreshold	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Indicates whether the value of the asset attribute is within the threshold of the RFC. The default value is <code>false</code>.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The name of the recordset filter criteria monitor.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date the value was last referenced.
LastViewedDate	Type dateTime

Field	Details
	Properties Filter, Nillable, Sort Description The date the value was last viewed.
RecordsetFilterCriteriaId	Type reference Properties Create, Filter, Group, Sort Description The ID of the recordset filter criteria. This field is a relationship field. Relationship Name RecordsetFilterCriteria Relationship Type Lookup Refers To RecordsetFilterCriteria

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

RecordsetFltrCritMonitorChangeEvent

Change events are available for the object.

RecordsetFltrCritMonitorHistory

History is available for tracked fields of the object.

ResourceAbsence

Represents a time period in which a service resource is unavailable to work in Field Service, Salesforce Scheduler, or Workforce Engagement. This object is available in API version 38.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `update()`, `upsert()`

Special Access Rules

Field Service or Workforce Engagement must be enabled.

Fields

Field Name	Details
AbsenceNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description (Read only) An auto-generated number identifying the absence.
Address	Type address Properties Filter Description The compound form of the address associated with the absence.
City	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The city of the address associated with the absence. Maximum length is 40 characters.
Country	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The country of the address associated with the absence. Maximum length is 80 characters.
Description	Type textarea Properties Create, Nillable, Update Description The description of the absence.
End	Type dateTime

Field Name	Details
	Properties Create, Filter, Sort, Update Description The date and time when the absence ends.
GeocodeAccuracy	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The level of accuracy of a location's geographical coordinates compared with its physical address. Usually provided by a geocoding service based on the address's latitude and longitude coordinates. This field is available in the API only.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the resource absence was last modified. Its label in the user interface is <code>Last Modified Date</code>.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the resource absence was last viewed.
Latitude	Type double Properties Create, Filter, Nillable, Sort, Update Description Used with <code>Longitude</code> to specify the precise geolocation of the address associated with the absence. Acceptable values are numbers between -90 and 90 with up to 15 decimal places.  Note: This field is available in the API only.
Longitude	Type double

Field Name	Details
	Properties Create, Filter, Nillable, Sort, Update Description Used with <code>Latitude</code> to specify the precise geolocation of the address associated with the absence. Acceptable values are numbers between –180 and 180 with up to 15 decimal places.  Note: This field is available in the API only.
Postal Code	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The postal code of the address associated with the absence. Maximum length is 20 characters.
ResourceId	Type reference Properties Create, Filter, Group, Sort, Update Description The absent service resource. This is a relationship field. Relationship Name Resource Relationship Type Lookup Refers To ServiceResource
Start	Type dateTime Properties Create, Filter, Sort, Update Description The date and time when the absence begins.
State	Type string Properties Create, Filter, Group, Nillable, Sort, Update

Field Name	Details
	Description The state of the address associated with the absence. Maximum length is 80 characters.
Street	Type textarea Properties Create, Filter, Group, Nillable, Sort, Update Description The street number and name of the address associated with the absence.
Type	Type picklist Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The type of absence: <i>Meeting, Training, Medical, or Vacation</i> . The default value is <i>Vacation</i> . You can add custom values if needed, but the name <i>Break</i> is reserved for the Field Service managed package.

Usage

Resource absences you define periods of time when a service resource is unavailable to work. Unless you're using the Field Service managed package, service resources can still be assigned to appointments that conflict with their absences.



Tip: Create a trigger that sends an approval request to a supervisor when a service resource creates an absence.

If you're not using the Field Service managed package, a calendar view isn't available for individual service resources.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ResourceAbsenceChangeEvent (API version 48.0)

Change events are available for the object.

ResourceAbsenceFeed

Feed tracking is available for the object.

ResourceAbsenceHistory

History is available for tracked fields of the object.

ResourcePreference

Represents an account's preference for a specified service resource on field service work.

Resource preferences indicate which service resources can be assigned to field service work. You can designate service resources as preferred, required, or excluded on specific accounts, assets, locations, work orders, or work order line items. Work orders inherit their associated account's resource preferences.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field Name	Details
<code>LastReferencedDate</code>	Type dateTime Properties Filter, Nillable, Sort Description The date when the resource preference was last modified.
<code>LastViewedDate</code>	Type dateTime Properties Filter, Nillable, Sort Description The date when the resource preference was last viewed.
<code>PreferenceType</code>	Type picklist Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update Description Resource preference type. Values include: - Preferred: Indicates that the customer would like their field service work assigned to the resource. - Required: Indicates that the resource must be assigned to the customer's field service work. - Excluded: Indicates that the customer doesn't want their field service work assigned to the resource.

Field Name	Details
	Resource preferences serve more as a suggestion than a requirement. You can still assign a service appointment to any resource regardless of the related work order's resource preferences.
RelatedRecordId	Type reference Properties Create, Filter, Group, Sort Description The work order or account with the resource preference. This field is a polymorphic relationship. Relationship Name RelatedRecord Relationship Type Lookup Refers To Accounts, Assets, Locations, Work Orders, or Work Order Line Items
ResourcePreferenceNumber	Type string Properties Autonumber, Defaulted on create, Filter, Sort Description An auto-generated number identifying the resource preference.
ServiceResourceId	Type reference Properties Create, Filter, Group, Sort, Update Description The service resource that is preferred, required, or excluded. This is a relationship field. Relationship Name ServiceResource Relationship Type Lookup Refers To ServiceResource

Associated Objects

This object has these associated objects. Unless noted, they're available in the same API version as this object.

ResourcePreferenceChangeEvent (API version 54.0)

Change events are available for the object.

ResourcePreferenceFeed

Feed tracking is available for the object.

ResourcePreferenceHistory

History is available for tracked fields of the object.

ReturnOrder

Represents the return or repair of inventory or products in Field Service, or the return of order products in Order Management. This object is available in API version 42.0 and later.

Return orders are available in Lightning Experience, Salesforce Classic, the Salesforce mobile app, the Field Service mobile app for Android and iOS, and communities built using Salesforce Tabs + Visualforce.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service or Order Management must be enabled. If return orders are enabled by a Salesforce Order Management license, they must be created with a Status corresponding to the Status Category Activated. The default Statuses corresponding to Activated are Submitted and Approved.

Fields

Field Name	Details
AccountId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The account associated with the return order. This is a relationship field. Relationship Name Account Relationship Type Lookup Refers To Account

Field Name	Details
CaseId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The case associated with the return order. This is a relationship field. Relationship Name Case Relationship Type Lookup Refers To Case
ContactId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The contact associated with the return order. This is a relationship field. Relationship Name Contact Relationship Type Lookup Refers To Contact
CurrencyIsoCode	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Available only for orgs with the multicurrency feature enabled. ISO code for the currency of the OrderSummary associated with the ReturnOrder. Possible values are: • DKK—Danish Krone • EUR—Euro • GBP—British Pound • USD—U.S. Dollar

Field Name	Details
	The default value is USD. This field is available in API version 49.0 and later.
Description	Type textarea Properties Create, Nillable, Update Description Notes or context about the return order.
DestinationLocationId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The location where the items are being returned to. For example, if the return order tracks the return of products from a technician's van to a warehouse, the warehouse is the destination location. This is a relationship field. Relationship Name DestinationLocation Relationship Type Lookup Refers To Location
ExpectedArrivalDate	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description The date when the items are expected to arrive at the destination location.
ExpirationDate	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description Authorizations can't be captured after their expiration dates. This field is available in API version 50.0 and later.

Field Name	Details
GrandTotalAmount	Type currency Properties Filter, Nillable, Sort Description Total, including adjustments and tax, of the products, fees, and delivery charges on the return order. This includes all return order line items associated with the return order. This amount is equal to TotalAmount + TotalTaxAmount. This is a calculated field. This field is available in API version 50.0 and later.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the return order was last modified. Its label in the user interface is <code>Last Modified Date</code>.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the return order was last viewed.
LifeCycleType	Type picklist Properties Filter, Group, Nillable, Restricted picklist, Sort Description Specifies whether the order summary is managed by Salesforce Order Management (MANAGED) or by an external system (UNMANAGED). An unmanaged order summary is stored in Salesforce for reference purposes. - Some Order Management APIs reject input records that are associated with unmanaged order summaries. - Order Management does not update financial bucket fields on some records that are associated with unmanaged order summaries. - A user with the <code>EditUnmanagedOrderSummaries</code> or <code>B2BCommerceIntegrator</code> permission can edit certain fields on objects related to unmanaged order summaries that are normally only accessible via APIs. Possible values are:

Field Name	Details
	- MANAGED—Managed - UNMANAGED—Unmanaged This field is available in API version 50.0 and later.
OrderId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The order associated with the return order. When you associated a return order with an order, you can associate the return order's line items with order products. This is a relationship field. Relationship Name Order Relationship Type Lookup Refers To Order
OrderSummaryId	Type reference Properties Create, Filter, Group, Nillable, Sort Description ID of the order summary associated with the return order. This field is available in API version 50.0 and later.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The owner of the return order. This is a polymorphic relationship field. Relationship Name Owner Relationship Type Lookup Refers To Group, User

Field Name	Details
ProductRequestId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The product request associated with the return order. When you associated a return order with a product request, you can associate the return order's line items with the product request's line items. A return order might be related to a product request if the return order tracks the return of unused products or products to be repaired or replaced. For example, a technician creates a product request for three motors to prepare for a field visit. If the technician finds that only two motors are needed, they can create a return order to return the third to the original location, and list the product request in this field. This is a relationship field. Relationship Name ProductRequest Relationship Type Lookup Refers To ProductRequest This field is available only if Field Service or Health Cloud is enabled.
ProductServiceCampaignId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The product service campaign associated with the return order This field is available only if Field Service is enabled.
RefundInstructionsHint	Type textarea Properties Nillable Description Stores a JSON representation of the payment credit and refund sequences for ensure credit, ensure refund, and the change orders associated with it. This field is available in API version 65.0 and later.

Field Name	Details
ReturnOrderNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description (Read only) Auto-generated number identifying the return order.
ReturnedById	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description ID of the user returning the items. This is a relationship field. Relationship Name ReturnedBy Relationship Type Lookup Refers To User
ShipFromAddress	Type address Properties Filter, Nillable Description The return shipping address. This address tracks the location of the items at the start of the return or repair. For example, if a customer is returning an item, the Ship From address is the customer's address.
ShipFromCity	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The city of the return shipping address. This address tracks the location of the items at the start of the return or repair. For example, if a customer is returning an item, the Ship From address is the customer's address.
ShipFromCountry	Type string

Field Name	Details
	Properties Create, Filter, Group, Nillable, Sort, Update Description The country of the return shipping address. This address tracks the location of the items at the start of the return or repair. For example, if a customer is returning an item, the Ship From address is the customer's address.
ShipFromGeocodeAccuracy	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Accuracy level of the geocode for the return shipping address. See Compound Fields Considerations and Limitations for details on geolocation compound fields. This field is available in the API only.
ShipFromLatitude	Type double Properties Create, Filter, Nillable, Sort, Update Description Used with Longitude to specify the precise geolocation of the return shipping address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places. See Compound Fields Considerations and Limitations for details on geolocation compound fields. This field is available in the API only.
ShipFromLongitude	Type double Properties Create, Filter, Nillable, Sort, Update Description Used with Latitude to specify the precise geolocation of the return shipping address. Acceptable values are numbers between –180 and 180 with up to 15 decimal places. See Compound Fields Considerations and Limitations for details on geolocation compound fields. This field is available in the API only.
ShipFromPostalCode	Type string Properties Create, Filter, Group, Nillable, Sort, Update

Field Name	Details
	Description The postal code of the return shipping address. This address tracks the location of the items at the start of the return or repair. For example, if a customer is returning an item, the Ship From address is the customer's address.
ShipFromState	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The state of the return shipping address. This address tracks the location of the items at the start of the return or repair. For example, if a customer is returning an item, the Ship From address is the customer's address.
ShipFromStreet	Type textarea Properties Create, Filter, Group, Nillable, Sort, Update Description The street of the return shipping address. This address tracks the location of the items at the start of the return or repair. For example, if a customer is returning an item, the Ship From address is the customer's address.
ShipmentType	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description The type of shipment associated with the return order. Available values are: • Standard (default value) • Rush • Overnight • Next Business Day • Pick Up
SourceLocationId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update

Field Name	Details
	Description The items' location at the start of the return or repair. For example, if the return order tracks the return of products from a technician's service vehicle to a warehouse, the service vehicle is the source location. This is a relationship field. Relationship Name SourceLocation Relationship Type Lookup Refers To Location
Status	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description The status of the return order. Available values are: • Draft • Submitted • Approved • Canceled • Closed If return orders are enabled by a Salesforce Order Management license, they must be created with a Status corresponding to the Status Category Activated. The default Statuses corresponding to Activated are Submitted and Approved.
StatusCategory	Type picklist Properties Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort Description Status category of the return order. Processing of the return order depends on this value. Each status category corresponds to one or more statuses. Possible values are: • Activated • Canceled • Closed • Draft • Pending

Field Name	Details
	This field is available in API version 50.0 and later.
TaxLocaleType	Type picklist Properties Filter, Group, Nillable, Restricted picklist, Sort Description The system used to handle tax on the original order associated with the return order. Gross usually applies to taxes like value-added tax (VAT), and Net usually applies to taxes like sales tax. Possible values are: • <code>Automatic</code> (displays most prices and taxes as combined values) • <code>Gross</code> (displays most prices and taxes as combined values) • <code>Net</code> (displays most prices and taxes as separate values) This field is available in API version 50.0 and later.
TotalAmount	Type currency Properties Filter, Nillable, Sort Description Adjusted total, not including tax, of the return order line items, including products, fees, and delivery charges, on the ReturnOrder. This is a calculated field. This field is available in API version 50.0 and later.
TotalDeliveryAdjustAmount	Type currency Properties Filter, Nillable, Sort Description Total amount of the price adjustments applied to the delivery charges on the return order. This value only includes adjustments to return order line items of type code Charge. This is a calculated field. This field is available in API version 50.0 and later.
TotalDeliveryAdjustAmtWithTax	Type currency Properties Filter, Nillable, Sort

Field Name	Details
	Description Total amount of the price adjustments applied to the delivery charges on the return order, inclusive of tax. This value only includes adjustments to return order line items of type code Charge. This amount is equal to $\text{TotalDeliveryAdjustAmount} + \text{TotalDeliveryAdjustTaxAmount}$. This is a calculated field. This field is available in API version 50.0 and later.
TotalDeliveryAdjustTaxAmount	Type currency Properties Filter, Nillable, Sort Description Tax on the TotalDeliveryAdjustAmount. This is a calculated field. This field is available in API version 50.0 and later.
TotalDeliveryAmount	Type currency Properties Filter, Nillable, Sort Description Total of the delivery charges on the return order. This value only includes return order line items of type code Charge. This is a calculated field. This field is available in API version 50.0 and later.
TotalDeliveryAmtWithTax	Type currency Properties Filter, Nillable, Sort Description Total amount of the delivery charges on the return order, inclusive of tax. This value only includes return order line items of type code Charge. This amount is equal to $\text{TotalDeliveryAmount} + \text{TotalDeliveryTaxAmount}$. This is a calculated field. This field is available in API version 50.0 and later.
TotalDeliveryTaxAmount	Type currency

Field Name	Details
	Properties Filter, Nillable, Sort Description Tax on the TotalDeliveryAmount. This is a calculated field. This field is available in API version 50.0 and later.
TotalFeeAdjustAmount	Type currency Properties Filter, Nillable, Sort Description Total amount of the price adjustments applied to the fees on the return order. This value only includes adjustments to return order line items of type Fee. This is a calculated field. This field is available in API version 56.0 and later.
TotalFeeAdjustAmtWithTax	Type currency Properties Filter, Nillable, Sort Description Total amount of the price adjustments applied to the fees on the return order, inclusive of tax. This value only includes adjustments to return order line items of type Fee. This amount is equal to TotalFeeAdjustAmount + TotalFeeAdjustTaxAmount. This is a calculated field. This field is available in API version 56.0 and later.
TotalFeeAdjustTaxAmount	Type currency Properties Filter, Nillable, Sort Description Tax on the TotalFeeAdjustAmount. This is a calculated field. This field is available in API version 56.0 and later.
TotalFeeAmount	Type currency

Field Name	Details
	Properties Filter, Nillable, Sort Description Total of the fees on the return order. This value only includes return order line items of type Fee. This is a calculated field. This field is available in API version 56.0 and later.
TotalFeeAmtWithTax	Type currency Properties Filter, Nillable, Sort Description Total amount of the fees on the return order, inclusive of tax. This value only includes return order line items of type Fee. This amount is equal to TotalFeeAmount + TotalFeeTaxAmount. This is a calculated field. This field is available in API version 56.0 and later.
TotalFeeTaxAmount	Type currency Properties Filter, Nillable, Sort Description Tax on the TotalFeeAmount. This is a calculated field. This field is available in API version 56.0 and later.
TotalProductAdjustAmount	Type currency Properties Filter, Nillable, Sort Description Total amount of the price adjustments applied to the products on the return order. This value only includes adjustments to return order line items of type code Product. This is a calculated field. This field is available in API version 50.0 and later.
TotalProductAdjustAmtWithTax	Type currency

Field Name	Details
	Properties Filter, Nillable, Sort Description Total amount of the price adjustments applied to the products on the return order, inclusive of tax. This value only includes adjustments to return order line items of type code Product. This amount is equal to TotalProductAdjustAmount + TotalProductAdjustTaxAmount. This is a calculated field. This field is available in API version 50.0 and later.
TotalProductAdjustTaxAmount	Type currency Properties Filter, Nillable, Sort Description Tax on the TotalProductAdjustmentAmount. This is a calculated field. This field is available in API version 50.0 and later.
TotalProductAmount	Type currency Properties Filter, Nillable, Sort Description Total of the product charges on the return order. This value only includes return order line items of type code Product. This is a calculated field. This field is available in API version 50.0 and later.
TotalProductAmtWithTax	Type currency Properties Filter, Nillable, Sort Description Total amount of the product charges on the return order, inclusive of tax. This value only includes return order line items of type code Product. This amount is equal to TotalProductAmount + TotalProductTaxAmount. This is a calculated field. This field is available in API version 50.0 and later.

Field Name	Details
TotalProductTaxAmount	Type currency Properties Filter, Nillable, Sort Description Tax on the TotalProductAmount. This is a calculated field. This field is available in API version 50.0 and later.
TotalTaxAmount	Type currency Properties Filter, Nillable, Sort Description Tax on the TotalAmount. This is a calculated field. This field is available in API version 50.0 and later.

Usage

You can use return orders to track customer returns, customer repairs, or the return of inventory from a technician's van stock to a warehouse or supplier. Customers can initiate a return from a community, or agents can create return orders in response to a customer call or technician request.

Return orders are composed of return order line items, which allow you to add details about the items being returned. To represent the returned items, each line item must list one or more of the following: product, product item, asset, product request line item, and order product. Return orders can be associated with a product request, case, account, contact, and order if needed. This versatility lets you use return orders to track a wide range of return scenarios.

Example

```
{
  "RefundInstructionsHint": {
    "PaymentCreditSequence": [
      {
        "OrderPaymentSummaryId": "0bMxx00000000000001",
        "Amount": 50,
        "CreditType": "GIFT_CARD",
        "Rank": 1
      },
      {
        "OrderPaymentSummaryId": "0bMxx00000000000002",
        "Amount": 50,
        "CreditType": "CHECK",

```

```

        "Rank": 2
      }
    ],
    "RefundSequence": [
      {
        "OrderPaymentSummaryId": "0bMxx0000000000001",
        "Amount": 50,
        "Rank": 1
      },
      {
        "OrderPaymentSummaryId": "0bMxx0000000000002",
        "Amount": 50,
        "Rank": 2
      }
    ],
    "ChangeOrders": [
      {
        "ChangeOrderId": "801xx000003Gd01111",
        "FeeChangeOrderId": null,
        "NetAmount": -75
      }
    ]
  }
}

```

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ReturnOrderChangeEvent (API version 48.0)

Change events are available for the object.

ReturnOrderFeed

Feed tracking is available for the object.

ReturnOrderHistory

History is available for tracked fields of the object.

ReturnOrderOwnerSharingRule

Sharing rules are available for the object.

ReturnOrderShare

Sharing is available for the object.

ReturnOrderLineItem

Represents a specific product that is returned or repaired as part of a return order in Field service, or a specific order item that is returned as part of a return order in Order Management. This object is available in API version 42.0 and later.

Return orders are available in Lightning Experience, Salesforce Classic, the Salesforce mobile app, the Field Service mobile app for Android and iOS, and communities built using Salesforce Tabs + Visualforce.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service or Order Management must be enabled.

Fields

Field Name	Details
<code>AssetId</code>	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The asset associated with the return order line item. One or more of the following fields must be filled out: <code>AssetId</code>, <code>OrderItemId</code>, <code>Product2Id</code>, <code>ProductItemId</code>, and <code>ProductRequestLineItemId</code>. This is a relationship field. Relationship Name Asset Relationship Type Lookup Refers To Asset
<code>ChangeOrderItemId</code>	Type reference Properties Create, Filter, Group, Nillable, Sort Description ID of the change order item associated with the return order line item. This field is available in API version 50.0 and later. This is a relationship field. Relationship Name ChangeOrderItem Relationship Type Lookup Refers To OrderItem

Field Name	Details
CurrencyIsoCode	Type picklist Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort Description Available only for orgs with the multicurrency feature enabled. Contains the ISO code for the currency of the original Order associated with the ReturnOrderLineItem. Possible values are: • DKK—Danish Krone • EUR—Euro • GBP—British Pound • USD—U.S. Dollar The default value is USD. This field is available in API version 49.0 and later.
Description	Type textarea Properties Create, Nillable, Update Description Notes or context about the return order line item.
DestinationLocationId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The location where the items are being returned to. For example, if the return order tracks the return of products from a technician's van to a warehouse, the warehouse is the destination location. This is a relationship field. Relationship Name DestinationLocation Relationship Type Lookup Refers To Location
GrossUnitPrice	Type currency

Field Name	Details
	Properties Create, Filter, Nillable, Sort, Update Description Unit price, including tax, of the product represented by the associated order item summary. This field is available in API version 50.0 and later.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the return order line item was last modified. Its label in the user interface is <code>Last Modified Date</code>.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the return order line item was last viewed.
OrderItemId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The order product associated with the return order line item. One or more of the following fields must be filled out: <code>AssetId</code>, <code>OrderItemId</code>, <code>Product2Id</code>, <code>ProductId</code>, and <code>ProductRequestLineItemId</code>. This is a relationship field. Relationship Name OrderItem Relationship Type Lookup Refers To OrderItem
OrderItemSummaryId	Type reference

Field Name	Details
	Properties Create, Filter, Group, Nillable, Sort Description ID of the order item summary associated with the return order line item. This field is available in API version 50.0 and later.
ProcessingPlan	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description Indicates the preferred fate of the items following their return. Available values are: • Repair—Repair the items and return them to the owner • Discard—Discard the items • Salvage—Salvage the items' working parts • Restock—Return the items to your inventory
Product2Id	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The product associated with the return order line item. One or more of the following fields must be filled out: AssetId, OrderItemId, Product2Id, ProductItemId, and ProductRequestLineItemId. This is a relationship field. Relationship Name Product2 Relationship Type Lookup Refers To Product2
ProductItemId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update

Field Name	Details
	Description The product item representing the location of the product at the start of the return. One or more of the following fields must be filled out: AssetId, OrderItemId, Product2Id, ProductItemId, and ProductRequestLineItemId. This is a relationship field. Relationship Name ProductItem Relationship Type Lookup Refers To ProductItem
ProductRequestLineItemId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The product request line item associated with the return order line item. One or more of the following fields must be filled out: AssetId, OrderItemId, Product2Id, ProductItemId, and ProductRequestLineItemId. This is a relationship field. Relationship Name ProductRequestLineItem Relationship Type Lookup Refers To ProductRequestLineItem
ProductServiceCampaignId	Type reference Properties Filter, Group, Nillable, Sort Description The product service campaign associated with the return order line item.
ProductServiceCampaignItemId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The product service campaign item associated with the return order line item.

Field Name	Details
QuantityExpected	Type double Properties Create, Filter, Nillable, Sort, Update Description The quantity of items expected to be returned. This field is available in API version 50.0 and later.
QuantityReceived	Type double Properties Create, Filter, Nillable, Sort, Update Description The actual quantity of items received for return. This field is available in API version 50.0 and later.
QuantityRejected	Type double Properties Create, Filter, Nillable, Sort, Update Description The quantity of items rejected for return. This field is available in API version 50.0 and later.
QuantityReturned	Type double Properties Create, Filter, Sort, Update Description The quantity of items being returned. If multiple types of products are being returned, track each product in a different return order line item.
QuantityUnitOfMeasure	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update Description Units of the returned items; for example, kilograms or liters. Quantity Unit of Measure picklist values are inherited from the Quantity Unit of Measure field on products.

Field Name	Details
ReasonForRejection	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description Reason for rejecting returned items on this return order line item. Possible values are: • Damaged Item • Expired Warranty • Missing Item or Part • Wrong Item The default value is Missing Item or Part. This field is available in API version 50.0 and later.
ReasonForReturn	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description The reason the items are being returned. Available values are: • Damaged • Defective • Duplicate Order • Wrong Item • Wrong Quantity • Not Satisfied • Outdated • Other The default value is Damaged.
ReasonForChangeText	Type string Properties Filter, Group, Sort Description Details about the reason for return change
RepaymentMethod	Type picklist

Field Name	Details
	Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description The method by which the customer or owner will be reimbursed for the items being returned. Available values are: • Replace—The items will be replaced • Refund—The items will be returned and the owner will be refunded • Credit—The items will be returned and the owner will receive credit for them • Return—The items will be returned to the owner (for example, following their repair)
ReturnOrderId	Type reference Properties Create, Filter, Group, Sort Description The return order that the return order line item belongs to. This is a relationship field. Relationship Name ReturnOrder Relationship Type Lookup Refers To ReturnOrder
ReturnOrderLineItemNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description (Read only) Auto-generated number that identifies the return order line item.
SourceLocationId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The items' location at the start of the return or repair. For example, if the return order tracks the return of products from a technician's service vehicle to a warehouse, the service vehicle is the source location.

Field Name	Details
	This is a relationship field. Relationship Name SourceLocation Relationship Type Lookup Refers To Location
TotalAdjustmentAmount	Type currency Properties Filter, Nillable, Sort Description Total of all price adjustments applied to the return order line item. This is a calculated field. This field is available in API version 50.0 and later.
TotalAdjustmentAmountWithTax	Type currency Properties Filter, Nillable, Sort Description Total amount of the price adjustments applied to the return order line item, inclusive of tax. This amount is equal to TotalAdjustmentAmount + TotalAdjustmentTaxAmount. This is a calculated field. This field is available in API version 50.0 and later.
TotalAdjustmentTaxAmount	Type currency Properties Filter, Nillable, Sort Description Tax on the TotalAdjustmentAmount. This is a calculated field. This field is available in API version 50.0 and later.
TotalAmount	Type currency Properties Filter, Nillable, Sort

Field Name	Details
	Description Total, including adjustments and tax, of the return order line item. This is a calculated field. This field is available in API version 50.0 and later.
TotalLineAmount	Type currency Properties Create, Defaulted on create, Filter, Nillable, Sort, Update Description Total, not including adjustments or tax, of the return order line item. This field is available in API version 50.0 and later.
TotalLineAmountWithTax	Type currency Properties Filter, Nillable, Sort Description Total price of the return order line item, inclusive of tax. This amount is equal to TotalLineAmount + TotalLineTaxAmount. This is a calculated field. This field is available in API version 50.0 and later.
TotalLineTaxAmount	Type currency Properties Filter, Nillable, Sort Description Tax on the TotalLineAmount. This is a calculated field. This field is available in API version 50.0 and later.
TotalPrice	Type currency Properties Filter, Nillable, Sort Description Total, including adjustments but not tax, of the return order line item. Equal to UnitPrice times Quantity. This is a calculated field.

Field Name	Details
TotalTaxAmount	Type currency Properties Filter, Nillable, Sort Description Tax on the TotalAmount. This is a calculated field. This field is available in API version 50.0 and later.
Type	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Type of the return order line item. Matches the type of the associated order item summary. Delivery Charge indicates that the return order line item represents a delivery charge. Fee indicates that it represents another type of fee, such as a return fee. Order Product indicates that it represents any other type of product, service, or charge. Each type corresponds to one type code, shown here in parentheses. Possible values are: • Delivery Charge (Charge) • Fee (Charge) This value is available in API v56.0 and later. • Order Product (Product) This field is available in API version 50.0 and later.
TypeCode	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Type code of the return order line item. Matches the type code of the associated order item summary. Processing depends on this value. Charge indicates that the return order line item represents a delivery charge. Product indicates that it represents an other type of product, service, or charge. Each type category corresponds to one or more types. Possible values are: • Charge • Product This field is available in API version 50.0 and later.

Field Name	Details
UnitPrice	Type currency Properties Create, Defaulted on create, Filter, Nillable, Sort, Update Description Unit price of the return order line item. This field is available in API version 50.0 and later.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ReturnOrderLineItemChangeEvent (API version 48.0)

Change events are available for the object.

ReturnOrderLineItemFeed

Feed tracking is available for the object.

ReturnOrderLineItemHistory

History is available for tracked fields of the object.

SerializedProduct

Records serial numbers for each individual product in an inventory. This object is available in API version 50.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

To use this object, you need the Manage Industries Visit permission and the Industries Visit permission set license. This rule doesn't apply to using serialized products in Field Service.

Fields

Field	Details
AssetId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description Reference to an asset record.

Field	Details
ExpirationDate	Type date Properties Create, Filter, Group, Nillable, Sort, Update Description The date the product expires.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The last date and time this record was referenced.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The last date and time this record was viewed.
Name	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description The name of the serialized product record. This is auto-generated.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Reference to the owner of the record.
Product2Id	Type reference Properties Create, Filter, Group, Sort, Update Description The product that is serialized.

Field	Details
ProductItemId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description Reference to the inventory, a product item record, this serialized product belongs to.
SerialNumber	Type string Properties Create, Filter, Group, Sort, Update Description The serial number of the product.
Status	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The status of the product. Possible values include: • Available • Consumed • Damaged • Lost • Sent The default value is Available.

SerializedProductTransaction

Represents transactions performed on a serialized product. This object is available in API version 57.0 and later.

Supported Calls

`describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`

Fields

Field	Details
LastReferencedDate	Type dateTime

Field	Details
	Properties Filter, Nillable, Sort Description The last date and time this record was referenced.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The last date and time this record was viewed.
Name	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description The name of the serialized product record. This is auto-generated.
RelatedRecordId	Type reference Properties Filter, Group, Sort Description The serialized product being transferred. This field is a polymorphic relationship field. Relationship Name SerializedProduct Relationship Type Lookup Refers To ProductConsumed, ProductTransfer
SerializedProductId	Type reference Properties Filter, Group, Sort Description The serialized product being transferred. This field is a relationship field. Relationship Name SerializedProduct

Field	Details
	Relationship Type Lookup Refers To SerializedProduct
TransactionType	Type picklist Properties Filter, Group, Nillable, Restricted picklist, Sort Description The action that the transaction tracks. Possible values are: - Adjusted - Consumed - Damaged - Found - Lost - Received - Repaired - Replenished - Sent - Withdrawn

Associated Objects

This object has these associated objects. Unless noted, they're available in the same API version as this object.

SerializedProductTransactionChangeEvent (API version 64.0)

Change events are available for the object.

SerializedProductTransactionFeed

Feed tracking is available for the object.

SerializedProductTransactionHistory

History is available for tracked fields of the object.

ServiceAppointment

Represents an appointment to complete work for a customer in Field Service, Lightning Scheduler, Intelligent Appointment Management, and Virtual Care. This object is available in API version 38.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field Name	Details
AccountId	Type reference Properties Filter, Group, Nillable, Sort Description (Read only) The account associated with the appointment. If the parent record is a work order or work order line item, this field's value is inherited from the parent. Otherwise, it remains blank. This is a relationship field. Relationship Name Account Relationship Type Lookup Refers To Account
ActualDuration	Type double Properties Create, Filter, Nillable, Sort, Update Description The number of minutes that it took the resource to complete the appointment after arriving at the address. When values are first added to the <code>Actual Start</code> and <code>Actual End</code> fields, the <code>Actual Duration</code> is automatically populated to list the difference between the <code>Actual Start</code> and <code>Actual End</code>. If the <code>Actual Start</code> and <code>Actual End</code> fields are subsequently updated, the <code>Actual Duration</code> field doesn't re-update, but you can manually update it.
ActualEndTime	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description The actual date and time the appointment ended.

Field Name	Details
ActualStartTime	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description The actual date and time the appointment started.
Address	Type address Properties Filter Description The address where the appointment is taking place. The address is inherited from the parent record if the parent record is a work order or work order line item.
AppointmentNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description An auto-assigned number that identifies the appointment.
ArrivalWindowEndTime	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description The end of the window of time in which the technician is scheduled to arrive at the site. This window is typically larger than the Scheduled Start and End window to allow time for delays and scheduling changes. You may choose to share the Arrival Window Start and End with the customer, but keep the Scheduled Start and End internal-only.
ArrivalWindowStartTime	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description The beginning of the window of time in which the technician is scheduled to arrive at the site. This window is typically larger than the Scheduled Start and End window to allow time for delays and scheduling changes. You may choose

Field Name	Details
	to share the Arrival Window Start and End with the customer, but keep the Scheduled Start and End internal-only.
BundlePolicyId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description Reference to the bundle policy associated with this service appointment. This is a relationship field. Relationship Name BundlePolicy Relationship Type Lookup Refers To ApptBundlePolicy
City	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The city where the appointment is completed. Maximum length is 40 characters.
ContactId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The contact associated with the parent record. If needed, you can manually update the service appointment contact. This is a relationship field. Relationship Name Contact Relationship Type Lookup Refers To Contact
Country	Type string

Field Name	Details
	Properties Create, Filter, Group, Nillable, Sort, Update Description The country where the work order is completed. Maximum length is 80 characters.
Description	Type textarea Properties Create, Nillable, Update Description The description of the appointment.
DueDate	Type dateTime Properties Create, Filter, Sort, Update Description The date by which the appointment must be completed. Earliest Start Permitted and Due Date typically reflect terms in the customer's service-level agreement.
Duration	Type double Properties Create, Nillable, Filter, Sort, Update Description The estimated length of the appointment. If the parent record is work order or work order line item, the appointment inherits its parent's duration, but it can be manually updated. The duration is in minutes or hours based on the value selected in the <code>Duration Type</code> field.
DurationType	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The unit of the Duration: Minutes or Hours.
EarliestStartTime	Type dateTime Properties Create, Filter, Sort, Update

Field Name	Details
	Description The date after which the appointment must be completed. Earliest Start Permitted and Due Date typically reflect terms in the customer's service-level agreement.
GeocodeAccuracy	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The level of accuracy of a location's geographical coordinates compared with its physical address. Usually provided by a geocoding service based on the address's latitude and longitude coordinates. This field is available in the API only.
IsAnonymousBooking	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether a service resource was automatically assigned to the appointment. The default value is false. This field is available in API version 49.0 and later.
IsBundle	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates if this service appointment is a bundle service appointment. The default value is false. This field is available in API version 54.0 and later.
IsBundleMember	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates if this service appointment is a bundle member service appointment. The default value is false. This field is available in API version 54.0 and later.
IsManuallyBundled	Type boolean

Field Name	Details
	Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates if this bundle was created manually. The default value is false. This field is available in API version 54.0 and later.
IsOffsiteAppointment	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Any type of work that can be done remotely. This field is available in API version 58.0 and later.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the service appointment was last modified. Its label in the user interface is <code>Last Modified Date</code>.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the service appointment was last viewed.
Latitude	Type double Properties Create, Filter, Nillable, Sort, Update Description Used with <code>Longitude</code> to specify the precise geolocation of the address where the service appointments is completed. Acceptable values are numbers between -90 and 90 with up to 15 decimal places. To integrate data from an external data source for latitude, map your data to the <code>ServiceAppointment.Latitude</code> and not the <code>ServiceAppointment.FSL__InternalSLRGeolocation__Latitude__s</code> field.

Field Name	Details
	This field is available in the API only.
Longitude	Type double Properties Create, Filter, Nillable, Sort, Update Description Used with <code>Latitude</code> to specify the precise geolocation of the address where the service appointment is completed. Acceptable values are numbers between -180 and 180 with up to 15 decimal places. To integrate data from an external data source for longitude, map your data to the <code>ServiceAppointment.Longitude</code> and not the <code>ServiceAppointment.FSL__InternalSLRGeolocation__Longitude__s</code> field. This field is available in the API only.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The owner of the service appointment. This is a polymorphic relationship field. Relationship Name Owner Relationship Type Lookup Refers To Group, User
ParentRecordId	Type reference Properties Create, Filter, Group, Nillable, Sort Description The parent record associated with the appointment. The parent record can't be updated after the service appointment is created. This is a polymorphic relationship field. Relationship Name ParentRecord

Field Name	Details
	Relationship Type Lookup Refers To Account, Asset, Lead, Opportunity, ServiceAppointmentGroup, WorkOrder, WorkOrderLineItem
ParentRecordStatusCategory	Type picklist Properties Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort Description (Read only) The <code>StatusCategory</code> of the parent record. If the parent record is a work order or work order line item, this field is populated; otherwise, it remains blank.
ParentRecordType	Type string Properties Filter, Group, Nillable, Sort Description (Read only) The type of parent record: Account, Asset, Lead, Opportunity, Work Order, or Work Order Line Item.
PostalCode	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The postal code where the work order is completed. Maximum length is 20 characters.
RelatedBundleId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The bundle that this service appointment is a member of. This is a relationship field. Relationship Name RelatedBundle Relationship Type Lookup

Field Name	Details
	Refers To ServiceAppointment
SchedEndTime	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description The time at which the appointment is scheduled to end. If you are using the Field Service managed package with the scheduling optimizer, this field is populated once the appointment is assigned to a resource. $\text{Scheduled End} - \text{Scheduled Start} = \text{Estimated Duration}$.
SchedStartTime	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description The time at which the appointment is scheduled to start. If you are using the Field Service managed package with the scheduling optimizer, this field is populated once the appointment is assigned to a resource.
ServiceDocumentTemplate	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The template ID which sets the template for each service document for the Document Builder feature.
ServiceTerritoryId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The service territory associated with the appointment. If the parent record is a work order or work order line item, the appointment inherits its parent's service territory. This is a relationship field. Relationship Name ServiceTerritory

Field Name	Details
	Relationship Type Lookup Refers To ServiceTerritory
State	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The state where the service appointment is completed. Maximum length is 80 characters.
Status	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description The status of the appointment. The picklist includes the following values, which can be customized: • <code>None</code>—Default value. • <code>Scheduled</code>—Appointment has been assigned to a service resource. • <code>Dispatched</code>—Assigned service resource has been notified about their assignment. • <code>In Progress</code>—Work has begun. • <code>Completed</code>—Work is complete. • <code>Cannot Complete</code>—Work could not be completed. • <code>Canceled</code>—Work is canceled, typically before any work began While you can set the status to null via the API, setting the status to null returns an error. To prevent errors, use one of the picklist values.
StatusCategory	Type picklist Properties Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort Description The category that each <code>Status</code> value falls into. The <code>Status Category</code> field's values are identical to the default <code>Status</code> values. If you create custom <code>Status</code> values, you must indicate which category it belongs to. For example, if you create a <code>Customer Absent</code> value, you may

Field Name	Details
	decide that it belongs in the <i>Cannot Complete</i> category. To learn which processes reference StatusCategory, see How are Status Categories Used?
Street	Type textarea Properties Create, Filter, Group, Nillable, Sort, Update Description The street number and name where the service appointment is completed.
Subject	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description A short phrase describing the appointment.
Transaction	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The last transaction ID of the scheduling and optimization request that updated this object. The transaction ID is automatically generated and populated by the Enhanced Scheduling and Optimization engine. Available in API version 63.0 and later.
WorkTypeId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The work type associated with the service appointment. The work type is inherited from the appointment's parent record if the parent is a work order or work order line item. If Lightning Scheduler is also in use, this field is editable. However, users see an error if they update it to list a different work type than the parent record's work type. This is a relationship field. Relationship Name WorkType Relationship Type Lookup

Field Name	Details
	Refers To WorkType

Usage

Service appointments always have a parent record, which can be a work order, work order line item, opportunity, account, or asset. The type of parent record tells you about the nature of the service appointment:

- Service appointments on *work orders* and *work order line items* offer a more detailed view of the work being performed. While work orders and work order line items let you enter general information about a task, service appointments are where you add the details about scheduling and ownership.
- Service appointments on *assets* represent work being performed on the asset.
- Service appointments on *accounts* represent work being performed for the account.
- Service appointments on *opportunities* represent work that is related to the opportunity.
- Service appointments on *leads* represent work that is related to lead—for example, a site visit to pursue a promising lead.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ServiceAppointmentChangeEvent (API version 48.0)

Change events are available for the object.

ServiceAppointmentFeed

Feed tracking is available for the object.

ServiceAppointmentHistory

History is available for tracked fields of the object.

ServiceAppointmentOwnerSharingRule

Sharing rules are available for the object.

ServiceAppointmentShare

Sharing is available for the object.

ServiceAppointmentStatus

Represents a possible status of a service appointment in field service.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

Field Service must be enabled.

Fields

Field Name	Details
ApiName	Type string Properties Filter, Group, idLookup, Sort Description The API name of the status value.
IsDefault	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Indicates that the status value is the default status on service appointments. Only one status value can be the default.
MasterLabel	Type string Properties Filter, Group, Nillable, Sort Description The label for the picklist value that appears in the UI.
SortOrder	Type int Properties Filter, Group, Nillable, Sort Description The value's position in the drop-down list of values in the UI.
StatusCode	Type picklist Properties Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort Description The status category that the value corresponds to. The Status Category field has seven values which are identical to the default Status values.

Usage

The Status field on service appointments comes with the following values:

- None—Default value.
- Scheduled—Appointment has been assigned to a service resource.
- Dispatched—Assigned service resource has been notified about their assignment.
- In Progress—Work has begun.
- Completed—Work is complete.
- Cannot Complete—Work could not be completed.
- Canceled—Work is canceled, typically before any work began
- CheckedIn—The customer has arrived for their scheduled appointment.

 **Important:** While you can set the status to null via the API, setting the status to null returns an error. To prevent errors, use one of the documented picklist values.

The ServiceAppointmentStatus object corresponds to the Status field. Adding a value to the Status field—for example, Waiting—creates a service appointment status record, and vice versa.

 **Note:** Service appointments also come with a StatusCategory field whose values are identical to the default Status values. If you create custom Status values, you must indicate which category it belongs to. For example, if you create a *Customer Absent* value, you may decide that it belongs in the *Cannot Complete* category. To learn which processes reference StatusCategory, see [How are Status Categories Used?](#)

ServiceContract

Represents a customer support contract (business agreement). This object is available in API version 18.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
AccountId	Type reference Properties Filter, Group, Nillable, Sort Description ID of the account associated with the service contract.
ActivationDate	Type dateTime Properties Filter, Nillable, Sort Description The initial day the service contract went into effect (whereas <code>StartDate</code> may include a renewal date).

Field	Details
AdditionalDiscount	Type percent Properties Create, Filter, Nillable, Sort, Update Description Extra discount percentage for the service contract. Available in API version 55.0 and later.
ApprovalStatus	Type picklist Properties Defaulted on create, Filter, Group, Nillable, Sort Description Approval status of the service contract.
BillingAddress (beta)	Type address Properties Filter, Nillable Description The compound form of the billing address. Read-only. See Address Compound Fields for details on compound address fields.
BillingCity	Type string Properties Filter, Group, Nillable, Sort Description Details for the billing address. Maximum size is 40 characters.
BillingCountry	Type string Properties Filter, Group, Nillable, Sort Description Details for the billing address. Maximum size is 40 characters.
BillingCountryCode	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The ISO country code for the service contract's billing address.
BillingLatitude	Type double Properties Create, Filter, Nillable, Sort, Update Description Used with <code>BillingLongitude</code> to specify the precise geolocation of a billing address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places.
BillingLongitude	Type double Properties Create, Filter, Nillable, Sort, Update Description Used with <code>BillingLatitude</code> to specify the precise geolocation of a billing address. Acceptable values are numbers between –180 and 180 with up to 15 decimal places.
BillingPostalCode	Type string Properties Filter, Group, Nillable, Sort Description Details for the billing address. Maximum size is 20 characters.
BillingState	Type string Properties Group, Sort, Filter, Nillable Description Details for the billing address. Maximum size is 20 characters.
BillingStateCode	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update Description The ISO state code for the service contract's billing address.

Field	Details
BillingStreet	Type textarea Properties Filter, Group, Nillable, Sort Description Street address for the billing address.
ContactId	Type reference Properties Filter, Group, Nillable, Sort Description Required. ID of the Contact associated with the service contract. Must be a valid ID.
ContractNumber	Type string Properties Autonumber, Defaulted on create, Filter, Sort Description Unique number automatically assigned to the service contract.
Description	Type textarea Properties Nillable Description Description of the service contract.
Discount	Type percent Properties Filter, Nillable, Sort Description Discount percentage for the service contract.
EndDate	Type date Properties Filter, Group, Nillable, Sort Description The last day the service contract is in effect.

Field	Details
GrandTotal	Type currency Properties Filter, Nillable, Sort Description Total price of the service contract plus shipping and taxes.
IsDeleted	Type boolean Properties Defaulted on create, Filter Description Indicates whether the object has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Label is Deleted.
LastReferencedDate	Type date Properties Filter, Nillable, Sort, Update Description The timestamp when the current user last interacted with this record, directly or indirectly. Some sample scenarios are:
LastViewedDate	Type date Properties Filter, Nillable, Sort, Update Description The timestamp when the current user last viewed this record or list view. If this value is null, it's possible that the user only accessed this record or list view (<code>LastReferencedDate</code>), but not viewed it.
LineItemCount	Type int Properties Filter, Nillable, Group, Sort Description Number of <code>ContractLineItem</code> records associated with the service contract.
Name	Type string

Field	Details
	Properties Create, Filter, Group, Sort, Update Description Name of the service contract.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The ID of the user who currently owns the service contract.
ParentServiceContractId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The service contract's parent service contract, if it has one.
Pricebook2Id	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description ID of the Pricebook2 associated with the service contract. Must be a valid ID.
RootServiceContractId	Type reference Properties Filter, Group, Nillable, Sort Description (Read only) The top-level service contract in a service contract hierarchy. Depending on where a service contract lies in the hierarchy, its root could be the same as its parent.
ShippingAddress (beta)	Type address Properties Filter, Nillable Description The compound form of the shipping address. Read-only. See Address Compound Fields for details on compound address fields.

Field	Details
ShippingCity	Type string Properties Filter, Group, Nillable, Sort Description Details of the shipping address. Maximum size is 40 characters.
ShippingCountry	Type string Properties Filter, Group, Nillable, Sort Description Details of the shipping address. Country maximum size is 40 characters.
ShippingCountryCode	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update Description The ISO country code for the service contract's shipping address.
ShippingLatitude	Type double Properties Create, Filter, Nillable, Sort, Update Description Used with <code>ShippingLongitude</code> to specify the precise geolocation of a shipping address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places.
ShippingLongitude	Type double Properties Create, Filter, Nillable, Sort, Update Description Used with <code>ShippingLatitude</code> to specify the precise geolocation of an address. Acceptable values are numbers between –180 and 180 with up to 15 decimal places.
ShippingPostalCode	Type string Properties Create, Filter, Nillable, Update

Field	Details
	Description Details of the shipping address. Postal code maximum size is 20 characters.
ShippingState	Type string Properties Create, Filter, Nillable, Update Description Details of the shipping address. State maximum size is 20 characters.
ShippingStateCode	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update Description The ISO state code for the service contract's shipping address.
ShippingStreet	Type textarea Properties Create, Filter, Nillable, Update Description The street address of the shipping address. Maximum of 255 characters.
SpecialTerms	Type textarea Properties Create, Nillable, Update Description Any terms specifically agreed to and tracked in the service contract.
StartDate	Type date Properties Create, Filter, Nillable, Update Description The first day the service contract is in effect.
Status	Type picklist Properties Filter, Nillable

Field	Details
	Description The status of the service contract, such as Inactive.
Subtotal	Type currency Properties Filter, Nillable Description Total of the service contract line items (products) before discounts, taxes, and shipping are applied.
Tax	Type currency Properties Create, Filter, Nillable, Update Description Total taxes for the service contract.
Term	Type int Properties Create, Filter, Nillable, Update Description Number of months that the service contract is valid.
TotalPrice	Type currency Properties Filter, Nillable Description Total of the contract line items (products) after discounts and before taxes and shipping.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ServiceContractChangeEvent (API version 44.0)

Change events are available for the object.

ServiceContractFeed (API version 23.0)

Feed tracking is available for the object.

ServiceContractHistory

History is available for tracked fields of the object.

ServiceContractOwnerSharingRule

Sharing rules are available for the object.

ServiceContractShare

Sharing is available for the object.

ServiceCrew

Represents a group of service resources who can be assigned to service appointments as a unit.

A service crew is a group of service resources whose combined skills and experience make them a good fit to work together on appointments. For example, a wellhead repair crew might include a hydrologist, a mechanical engineer, and an electrician.

Service appointments can only be assigned to service resources. To assign a service crew to service appointments, you must create a service resource with a resource type of Crew that represents the crew, then use the resource for assignment purposes.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field Name	Details
CrewSize	Type int Properties Create, Filter, Group, Sort, Update Description The number of members on the crew. This field is manual, so it doesn't auto-update when you add or remove members.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the service crew was last modified. Its label in the user interface is <code>Last Modified Date</code> .

Field Name	Details
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the service crew was last viewed.
Name	Type string Properties Create, Filter, Group, Sort, Update Description The name of the service crew. For example, Repair Crew.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The crew owner. By default, the owner is the person who created the service crew.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ServiceCrewChangeEvent (API version 48.0)

Change events are available for the object.

ServiceCrewFeed

Feed tracking is available for the object.

ServiceCrewHistory

History is available for tracked fields of the object.

ServiceCrewOwnerSharingRule

Sharing rules are available for the object.

ServiceCrewShare

Sharing is available for the object.

ServiceCrewMember

Represents a technician service resource that belongs to a service crew.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field Name	Details
EndDate	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description The last day that the service resource belongs to the crew. You can use this field to track employment dates for contractors.
IsLeader	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates that the member is the crew leader.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the service crew member was last modified. Its label in the user interface is <code>Last Modified Date</code>.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the service crew member was last viewed.
ServiceCrewId	Type reference

Field Name	Details
	Properties Create, Filter, Group, Sort Description The crew that the service resource belongs to.
ServiceCrewMemberNumber	Type string Properties Autonumber, Defaulted on create, Filter, Sort Description An auto-generated number identifying the service crew member.
ServiceResourceId	Type reference Properties Create, Filter, Group, Sort, Update Description The service resource that belongs to the crew. Only service resources whose resource type is Technician can be added to service crews.
StartDate	Type dateTime Properties Create, Filter, Sort, Update Description Required. The day the service resource joins the crew. Service resources can belong to multiple crews as long as their start and end dates don't overlap.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ServiceCrewMemberChangeEvent (API version 48.0)

Change events are available for the object.

ServiceCrewMemberFeed

Feed tracking is available for the object.

ServiceCrewMemberHistory

History is available for tracked fields of the object.

ServiceReport

Represents a report that summarizes a work order, work order line item, or service appointment.

The fields that appear on a service report are determined by its service report template. Service reports can be signed by the customer and shared as a PDF.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()` `update()`

Special Access Rules

Field Service must be enabled.

Fields

Field Name	Details
<code>ContentVersionDocumentId</code>	Type reference Properties Filter, Group, Nillable, Sort Description ID of the service report version, used for storage.
<code>DocumentBody</code>	Type base64 Properties Create, Nillable Description The report output. <code>DocumentBody</code> can't be retrieved via REST API.
<code>DocumentContentType</code>	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort Description The type of data used for the report output.. Possible values are: • audio/ogg • text/calendar • video/3gpp2 • video/3gpp • image/avif • text/calendar • audio/x-caf • image/webp

Field Name	Details
DocumentLength	Type int Properties Filter, Group, Nillable, Sort Description The length of the report output.
DocumentName	Type string Properties Create, Filter, Group, Nillable, Sort Description The name of the report output, always set to Service Report.
DocumentTemplate	Type string Properties Create, Filter, Group, Nillable, Sort Description The template used to generate service documents for the Document Builder feature.  Important: DocumentTemplate is different from Template. The document template needs to reference a flexipage that is of type serviceDocument and must target the object used to generate the service document. For example, you can't use an Account flexipage for a service report tied to a work order.
IsSigned	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort Description Indicates whether the service report contains one or more signatures. This field isn't supported for Document Builder.  Tip: Add this field to the Service Reports related list on work orders, work order line items, and service appointments.
ParentId	Type reference Properties Create, Filter, Group, Sort

Field Name	Details
	Description The ID of the service appointment, work order, or work order line item that the service report summarizes. For example, if you click Create Service Report on a service appointment, this field lists the service appointment's record ID.
ServiceReportLanguage	Type picklist Properties Create, Filter, Group, Nillable, Sort, Restricted picklist Description The language used for the service report. The language is selected in the <code>ServiceReportLanguage</code> field on the associated work order. If the work order doesn't specify a service report language, the report is translated in the default language in Salesforce of the person generating the report.
ServiceReportNumber	Type string Properties Autonumber, Defaulted on create, Filter, Sort Description An auto-generated number identifying the service report.
Status	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The status of the service report. Available in API version 53.0 and later. Possible values are: - Completed - Failed - Generating - In Progress - None - Queued The default value is <code>None</code>.
Template	Type string Properties Create, Filter, Group, Nillable, Sort

Field Name	Details
	Description The service report template used to generate the service report.
	 Note: If the person creating the service report doesn't have access to certain objects or fields that are included in the service report template, those fields aren't visible in the report they create.

Associated Objects

This object has the following associated objects. Unless noted, they are available in the same API version as this object.

ServiceReportChangeEvent

Change events are available for the object. Available in API version 55.0 and later.

ServiceReportHistory

History is available for tracked fields of the object.

ServiceReportLayout

Represents a service report template in field service.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

Field Service must be enabled. All users with Field Service Standard user permission can view the ServiceReportLayout object via the API.

Fields

Field Name	Details
DeveloperName	Type string
	Properties Filter, Group, Sort
	Description The developer name of the service report template.
	 Note: Only users with View DeveloperName OR View Setup and Configuration permission can view, group, sort, and filter this field.
Language	Type picklist

Field Name	Details
	Properties Filter, Group, Restricted picklist, Sort Description The language that the service report template uses.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date the service report template was last viewed.
MasterLabel	Type string Properties Filter, Group, Sort Description The name of the service report template. For example, Maintenance Report Template.
TemplateType	Type picklist Properties Defaulted on create, Filter, Group, Restricted picklist, Sort Description The type of the service report template. Available in API version 46.0 and later. Possible values are: • DigitalForm • ServiceReport The default value is <code>ServiceReport</code> .

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ServiceReportLayoutChangeEvent

Change events are available for the object. Available in API version 55.0 and later.

ServiceResource

Represents a service technician or service crew in Field Service and Salesforce Scheduler, or an agent in Workforce Engagement. This object is available in API version 38.0 and later.

Supported Calls

`create()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `update()`, `upsert()`

Special Access Rules

Field Service or Workforce Engagement must be enabled.

Fields

Field Name	Details
Description	Type textarea Properties Create, Nillable, Update Description The description of the resource.
IsActive	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description When selected, this option means that the resource can be assigned to work orders. For service tracking purposes, resources can't be deleted, so deactivating a resource is the best way to send them into retirement. Deactivating a user doesn't deactivate the related service resource. You can't create a service resource that is linked to an inactive user.
IsCapacityBased	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Capacity-based resources are limited to a certain number of hours or appointments in a specified time period.  Tip: The Capacities related list shows a resource's capacity.
IsOptimizationCapable	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update

Field Name	Details
	Description This field is reserved for Field Service and the managed package. Create a custom field instead of using this field to include a service resource in optimization.
LastKnownLatitude	Type double Properties Create, Filter, Nillable, Sort, Update Description The latitude of the last known location.
LastKnownLongitude	Type double Properties Create, Filter, Nillable, Sort, Update Description The longitude of the last known location.
LastKnownLocation	Type location Properties Nillable Description The service resource's last known location. You can configure this field to display data collected from a custom mobile app. This field isn't visible in the user interface, but you can expose it on service resource page layouts or set up field tracking to be able to view a resource's location history.
LastKnownLocationDate	Type dateTime Properties Filter, Nillable, Sort, Update Description The date and time of the last known location.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the service resource was last modified. Its label in the user interface is <code>Last Modified Date</code>.

Field Name	Details
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the service resource was last viewed.
LocationId	Type reference Properties Create, Filter, Group, Sort, Nillable, Update Description The location associated with the service resource. For example, a service vehicle driven by the service resource. LocationId is a relationship field. Relationship Name Location Relationship Type Lookup Refers To Location
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The resource's name, for example the name or title of the associated user or service crew.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The owner of the service resource. OwnerId is a polymorphic relationship field. Relationship Name Owner Relationship Type Lookup

Field Name	Details
	Refers To Group, User
RelatedRecordId	Type reference Properties Create, Filter, Group, Sort, Nillable, Update Description The associated user. Its label in the UI is <code>User</code> . If the service resource represents a service crew rather than a user, leave the <code>User</code> field blank and select the related crew in the <code>ServiceCrewId</code> field. RelatedRecordId is a relationship field. Relationship Name RelatedRecord Relationship Type Lookup Refers To User
ResourceType	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Indicates whether the resource is a Technician (T), Dispatcher (D), Crew (C), Asset (S), Agent (A), or Planner (P). The default value is Technician (T). Resources who are dispatchers can't be capacity-based or included in scheduling optimization. Only users with the Field Service Dispatcher permission-set license can be dispatchers. You can't add additional resource types. To create a dependent lookup filter with <code>ServiceResource.ResourceType</code> , use only the first letter of the picklist value, for example T for Technician.
ServiceCrewId	Type reference Properties Create, Filter, Group, Sort, Nillable, Update Description The associated service crew. If the service resource represents a crew, select the crew.  Note: This field is hidden for all users by default. To use it, update its field-level security settings in Setup and add it to your service resource page layouts.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ServiceResourceChangeEvent (API version 48.0)

Change events are available for the object.

ServiceResourceFeed

Feed tracking is available for the object.

ServiceResourceHistory

History is available for tracked fields of the object.

ServiceResourceOwnerSharingRule

Sharing rules are available for the object.

ServiceResourceShare

Sharing is available for the object.

ServiceResourceCapacity

Represents the maximum number of scheduled hours or number of service appointments that a capacity-based service resource can complete within a specific time period. This object is available in API version 38.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field Name	Details
<code>CapacityInHours</code>	Type double Properties Create, Filter, Nillable, Sort, Update Description The number of hours that the resource can work per time period. You must fill out this field, the <code>CapacityInWorkItems</code> field, or both.
<code>CapacityInWorkItems</code>	Type int Properties Create, Filter, Group, Nillable, Sort, Update

Field Name	Details
	Description The total number of service appointments that the resource can complete per time period. You must fill out this field, the <code>CapacityInHours</code> field, or both.
<code>CapacityNumber</code>	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description (Read only) An auto-generated number identifying the capacity record.
<code>EndDate</code>	Type date Properties Create, Filter, Group, Nillable, Sort, Update Description The date the capacity ends; for example, the end date of a contract.
<code>LastReferencedDate</code>	Type dateTime Properties Filter, Nillable, Sort Description The timestamp for when the current user last viewed a record related to this record.
<code>LastViewedDate</code>	Type dateTime Properties Filter, Nillable, Sort Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.
<code>ServiceResourceId</code>	Type reference Properties Create, Filter, Group, Sort Description The associated service resource. You can set multiple capacities for a resource as long as their start and end dates do not overlap.

Field Name	Details
StartDate	Type date Properties Create, Filter, Group, Sort Description The date the capacity goes into effect.
TimePeriod	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Days, Hours, or Months. For example, if a resource can work 80 hours per month, the capacity's <code>Time Period</code> would be <i>Month</i> and <code>Hours per Time Period</code> would be <i>80</i>.

Usage

Service resources who are capacity-based can only work a certain number of hours or complete a certain number of service appointments within a specified time period. Contractors tend to be capacity-based. To indicate that a service resource is capacity-based, select

Capacity-Based on the service resource record, then create a capacity record for the service resource.

You must fill out at least one of these fields: `CapacityInWorkItems` and `CapacityInHours`. If you're using the Field Service managed package and would like to measure capacity both in hours and in number of work items, enter a value for both. The resource is considered to reach their capacity based on whichever term is met first—hours or number of work items.



Important: If you aren't using the Field Service managed package, capacity serves more as a suggestion than a rule. Resources can still be as scheduled beyond their capacity, and you aren't notified when a resource exceeds their capacity.

Associated Objects

This object has the following associated objects. Unless noted, they are available in the same API version as this object.

ServiceResourceCapacityChangeEvent (API version 54.0)

Change events are available for the object.

ServiceResourceCapacityFeed

Feed tracking is available for the object.

ServiceResourceCapacityHistory

History is available for tracked fields of the object.

ServiceResourceSkill

Represents a skill that a service resource possesses in Field Service and Lightning Scheduler. This object is available in API version 38.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field Name	Details
<code>EffectiveEndDate</code>	Type datetime Properties Create, Filter, Nillable, Sort, Update Description The date when the skill expires. For example, if a service resource needs to be re-certified after six months, the end date would be the date their certification expires.
<code>EffectiveStartDate</code>	Type datetime Properties Create, Filter, Sort, Update Description The date when the service resource gains the skill. For example, if the skill represents a certification, the start date would be the date of certification.
<code>LastReferencedDate</code>	Type dateTime Properties Filter, Nillable, Sort Description The date when the resource skill was last modified. Its label in the user interface is <code>Last Modified Date</code>.
<code>LastViewedDate</code>	Type dateTime Properties Filter, Nillable, Sort Description The date when the resource skill was last viewed.

Field Name	Details
ServiceResourceId	Type reference Properties Create, Filter, Group, Sort Description The service resource who possesses the skill. This is a relationship field. Relationship Name ServiceResource Relationship Type Lookup Refers To ServiceResource
SkillId	Type reference Properties Create, Filter, Group, Sort, Update Description The skill the service resource possesses. This is a relationship field. Relationship Name Skill Relationship Type Lookup Refers To Skill
SkillLevel	Type double Properties Create, Defaulted on create, Filter, Nillable, Sort, Update Description The service resource's skill level. Skill level can range from zero to 99.99.
SkillNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description An auto-generated number identifying the resource skill assignment.

Usage

You can assign skills to all service resources in your org to indicate their certifications and areas of expertise, and specify each resource's skill level from 0 to 99.99. For example, you can assign Maria the "Welding" skill, level 50.

If you intend to use the skills feature, determine which skills you want to track and how skill level should be determined. For example, you may want the skill level to reflect years of experience, certification levels, or license classes.

Associated Objects

This object has the following associated objects. Unless noted, they are available in the same API version as this object.

ServiceResourceSkillChangeEvent (API version 54.0)

Change events are available for the object.

ServiceResourceSkillFeed

Feed tracking is available for the object.

ServiceResourceSkillHistory

History is available for tracked fields of the object.

ServiceTerritory

Represents a geographic or functional region in which work can be performed in Field Service, Salesforce Scheduler, or Workforce Engagement. This object is available in API version 38.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field Name	Details
Address	Type address Properties Filter Description An address to associate with the territory. For example, you can list the address of the territory's headquarters.
AvgTravelTime	Type int Properties Create, Filter, Group, Nillable, Sort, Update

Field Name	Details
	Description The average travel time for this service territory. The value is added to the Work Capacity Usage for each scheduled service appointment in the service territory. Available in API version 59.0 and later.
City	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The city of the associated address. Maximum length is 40 characters.
Country	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The country to associate with the territory. Maximum length is 80 characters.
Description	Type textarea Properties Create, Nillable, Update Description The description of the territory.
GeocodeAccuracy	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The level of accuracy of a location's geographical coordinates compared with its physical address. Usually provided by a geocoding service based on the address's latitude and longitude coordinates. This field is available in the API only.
IsActive	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update

Field Name	Details
	Description Indicates whether the service territory is meant to be used. If a territory is inactive, you can't add members to it or link it to work orders, work order line items, or service appointments.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the territory was last modified. Its label in the user interface is <code>Last Modified Date</code>.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the territory was last viewed.
Latitude	Type double Properties Create, Filter, Nillable, Sort, Update Description Used with <code>Longitude</code> to specify the precise geolocation of the address associated with the territory. Acceptable values are numbers between –90 and 90 with up to 15 decimal places. This field is available in the API only.
Longitude	Type double Properties Create, Filter, Nillable, Sort, Update Description Used with <code>Latitude</code> to specify the precise geolocation of the address associated with the territory. Acceptable values are numbers between –180 and 180 with up to 15 decimal places. This field is available in the API only.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update

Field Name	Details
	Description The name of the territory.
OperatingHoursId	Type reference Properties Create, Filter, Group, Sort, Update Description The territory's operating hours, which indicate when service appointments within the territory can occur. Service resources who are members of a territory automatically inherit the territory's operating hours unless different hours are specified on the resource record. This field is a relationship field. Relationship Name OperatingHours Relationship Type Lookup Refers To OperatingHours
ParentTerritoryId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The territory's parent service territory, if it has one. For example, a <i>Northern California</i> territory can have a <i>State of California</i> territory as its parent. A service territory hierarchy can contain up to 10,000 territories. This field is a relationship field. Relationship Name ParentTerritory Relationship Type Lookup Refers To ServiceTerritory
PostalCode	Type string Properties Create, Filter, Group, Nillable, Sort, Update

Field Name	Details
	Description The postal code of the address associated with the territory. Maximum length is 20 characters.
State	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The state of the address associated with the territory. Maximum length is 80 characters.
Street	Type textarea Properties Create, Filter, Group, Nillable, Sort, Update Description The street number and name of the address associated with the territory.
TopLevelTerritoryId	Type reference Properties Filter, Group, Nillable, Sort Description (Read only) The top-level territory in a hierarchy of service territories. Depending on where a territory lies in the hierarchy, its top-level territory can be the same as its parent. This field is a relationship field. Relationship Name TopLevelTerritory Relationship Type Lookup Refers To ServiceTerritory
TravelModeId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description ID of the TravelMode used for travel time calculations. The travel mode includes information about the type of transportation, such as a car or walking, whether

Field Name	Details
	a vehicle can take toll roads, and whether a vehicle is transporting hazardous materials. This field is a relationship field. Relationship Name TravelMode Relationship Type Lookup Refers To TravelMode
TravelTimeBuffer	Type int Properties Create, Filter, Group, Nillable, Sort, Update Description Add additional time to driving time, such as time to find parking or to walk to the site. This value overrides the Travel Time Buffer value defined in Field Service Settings Scheduling Routing.
TypicalInTerritoryTravelTime	Type double Properties Create, Filter, Nillable, Sort, Update Description Estimated number of minutes needed to travel from one location to another within the service territory. You can use this field in Apex customization.

Usage

If you want to use service territories, determine which territories to create. Depending on how your business works, you can create territories based on cities or counties, or on functional categories such as sales versus service. If you plan to build out a hierarchy of service territories, create the highest-level territories first.

For example, you can create a hierarchy of territories to represent the areas where your team works in California. Include a top-level territory named *California*, three child territories named *Northern California*, *Central California*, and *Southern California*, and a series of third-level territories corresponding to California counties. Assign service resources to each county territory to indicate who is available to work in that county.

Associated Objects

This object has these associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ServiceTerritoryChangeEvent (API version 48.0)

Change events are available for the object.

ServiceTerritoryFeed

Feed tracking is available for the object.

ServiceTerritoryHistory

History is available for tracked fields of the object.

ServiceTerritoryOwnerSharingRule

Sharing rules are available for the object.

ServiceTerritoryShare

Sharing is available for the object.

ServiceTerritoryLocation

Represents a location associated with a particular service territory in field service.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field Name	Details
LocationId	Type reference Properties Create, Filter, Group, Sort, Update Description The location that is associated with the service territory.
ServiceTerritoryId	Type reference Properties Create, Filter, Group, Sort Description The associated service territory.
ServiceTerritoryLocationNumber	Type string Properties Autonumber, Defaulted on create, Filter, Sort

Field Name	Details
	Description (Read only) Auto-generated number identifying the service territory location.

Associated Objects

This object has the following associated objects. Unless noted, they are available in the same API version as this object.

ServiceTerritoryLocationChangeEvent (API version 55.0)

Change events are available for the object.

ServiceTerritoryLocationFeed

Feed tracking is available for the object.

ServiceTerritoryLocationHistory

History is available for tracked fields of the object.

ServiceTerritoryMember

Represents a service resource who can be assigned in a service territory in Field Service, Salesforce Scheduler, or Workforce Engagement. This object is available in API version 38.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `update()`, `upsert()`

Special Access Rules

Field Service or Workforce Engagement must be enabled.

Fields

Field Name	Details
Address	Type address Properties Filter Description The member's address. You may want to list the related service resource's address in this field.
City	Type string Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update

Field Name	Details
	Description The city of the member's address. Maximum length is 40 characters.
Country	Type string Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description The country of the member's address. Maximum length is 80 characters.
EffectiveEndDate	Type datetime Properties Create, Filter, Nillable, Sort, Update Description The date when the service resource is no longer a member of the territory. If the resource will be working in the territory for the foreseeable future, leave this field blank. This field is mainly useful for indicating when a temporary relocation ends.
EffectiveStartDate	Type datetime Properties Create, Filter, Sort, Update Description The date when the service resource becomes a member of the service territory.
GeocodeAccuracy	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The level of accuracy of a location's geographical coordinates compared with its physical address. Usually provided by a geocoding service based on the address's latitude and longitude coordinates.  Note: This field is available in the API only.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort

Field Name	Details
	Description The date when the territory member was last modified. Its label in the user interface is <code>Last Modified Date</code> .
<code>LastViewedDate</code>	Type dateTime Properties Filter, Nillable, Sort Description The date when the territory member was last viewed.
<code>Latitude</code>	Type double Properties Create, Defaulted on create, Filter, Nillable, Sort, Update Description Used with <code>Longitude</code> to specify the precise geolocation of the member's address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places.  Note: This field is available in the API only.
<code>Longitude</code>	Type double Properties Create, Defaulted on create, Filter, Nillable, Sort, Update Description Used with <code>Latitude</code> to specify the precise geolocation of the member's address. Acceptable values are numbers between –180 and 180 with up to 15 decimal places.  Note: This field is available in the API only.
<code>MemberNumber</code>	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description (Read only) An auto-generated number identifying the service territory member.
<code>OperatingHoursId</code>	Type reference

Field Name	Details
	Properties Create, Filter, Group, Sort, Nillable, Update Description The operating hours assigned to the service territory member. If no operating hours are specified, the member is assumed to use their parent service territory's operating hours. If a member needs special operating hours, create them in Setup and select them in the <code>Operating Hours</code> lookup field on the member's detail page. This is a relationship field. Relationship Name OperatingHours Relationship Type Lookup Refers To OperatingHours
PostalCode	Type string Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description The postal code of the member's address. Maximum length is 20 characters.
ServiceResourceId	Type reference Properties Create, Filter, Group, Sort, Update Description The service resource assigned to the service territory. This is a relationship field. Relationship Name ServiceResource Relationship Type Lookup Refers To ServiceResource
ServiceTerritoryId	Type reference Properties Create, Filter, Group, Sort

Field Name	Details
	Description The service territory that the service resource is assigned to. This is a relationship field. Relationship Name ServiceTerritory Relationship Type Lookup Refers To ServiceTerritory
State	Type string Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description The state of the member's address. Maximum length is 80 characters.
Street	Type textarea Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description The street number and name of the member's address.
TerritoryType	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Primary, Secondary, or Relocation. - The primary territory is typically the territory where the resource works most often—for example, near their home base. Service resources can only have one primary territory. - Secondary territories are territories where the resource can be assigned to appointments if needed. Service resources can have multiple secondary territories. - Relocation territories represent temporary moves for service resources. If you're using the Field Service managed packages with the scheduling optimizer, resources with relocation territories are always assigned to services within their relocation territories during the specified relocation dates; if they don't have a relocation territory, the primary territories are favored over the secondary.

Field Name	Details
	For example, a service resource might have the following territories: • Primary territory: <i>West Chicago</i> • Secondary territories: – <i>East Chicago</i> – <i>South Chicago</i> • Relocation territory: <i>Manhattan</i>, for a three-month period
TravelModeId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description ID of the TravelMode used for travel time calculations. The travel mode includes information about the type of transportation, such as a car or walking, whether a vehicle can take toll roads, and whether a vehicle is transporting hazardous materials. This field is a relationship field. Relationship Name TravelMode Relationship Type Lookup Refers To TravelMode

Usage

If you delete a service territory with members, the service resources who were members no longer have any connection to the territory.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ServiceTerritoryMemberChangeEvent (API version 48.0)

Change events are available for the object.

ServiceTerritoryMemberFeed

Feed tracking is available for the object.

ServiceTerritoryMemberHistory

History is available for tracked fields of the object.

Shift

Represents a shift for service resource scheduling. Available in API versions 46.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `upsert()`

Special Access Rules

Field Service, Service Engagement, or Workforce Engagement must be enabled. For Field Service, users must have Field Service permissions. For Service Engagement, users must have the Service Engagement Planner permission set. For Workforce Engagement, users must have the Workforce Engagement Admin or Planner permission set.

Fields

Field	Details
BackgroundColor	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description Sets a background color when shifts are displayed in the UI. Use a 3- or 6-digit hexadecimal format, for example #FF00FF. Available in API version 54.0 and later.
EndTime	Type dateTime Properties Create, Filter, Sort, Update Description The date and time that the shift ends.
IsHolidayShift	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates a shift that overlaps with holiday hours. The default value is false. Available in API version 55.0 and later.
IsNonStandard	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether the shift is nonstandard, such as overtime or on-call shifts. The default value is false. Available in API version 54.0 and later.

Field	Details
JobProfileId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The job profile associated with the shift. Available in API versions 47.0 and later.
Label	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The label that a shift is given.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date and time when the current user last viewed a related record.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date and time when the current user last viewed this record.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The owner of the shift. This is a polymorphic relationship field. Relationship Name Owner Relationship Type Lookup Refers To Group, User

Field	Details
RecordsetFilterCriteriaId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the recordset filter criteria selected for the shift. Available in API version 49.0 and later. This is a relationship field. Relationship Name RecordsetFilterCriteria Relationship Type Lookup Refers To RecordsetFilterCriteria
ServiceResourceId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the service resource the shift belongs to. Available in API versions 47.0 and later. This is a relationship field. Relationship Name ServiceResource Relationship Type Lookup Refers To ServiceResource
ServiceTerritoryId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the service territory the shift belongs to. Available in API versions 47.0 and later. This is a relationship field. Relationship Name ServiceTerritory Relationship Type Lookup

Field	Details
	Refers To ServiceTerritory
ShiftNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description The number automatically given to the shift upon creation.
ShiftTemplateId	Type reference Properties Create, Filter, Group, Nillable, Sort Description The shift template ID, if the shift was created from a shift template. Available in API version 53.0 and later. This is a relationship field. Relationship Name ShiftTemplate Relationship Type Lookup Refers To ShiftTemplate
StartTime	Type dateTime Properties Create, Filter, Sort, Update Description The date and time that the shift starts.
Status	Type picklist Properties Create, Filter, Group, Sort, Update Description Describes the status of the shift. Users can create custom values. Default values are: • Tentative • Published • Confirmed

Field	Details
StatusCategory	Type picklist Properties Filter, Group, Nillable, Restricted picklist, Sort Description Describes the status of the shift using static values. This field is derived from <code>Status</code> using the mapping defined in setup. Possible values are: • Tentative • Published • Confirmed
TimeSlotType	Type picklist Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update Description Type of time slot for the shift. The same setup values as the <code>TimeSlot</code> field in the <code>OperatingHours</code> object. Possible values are: • Normal (default value) • Extended

Usage

Scheduling and dispatching service resources using shift data is not supported in API version 46.0, and is a pilot feature in API version 47.0.

Associated Objects

This object has the following associated objects. Unless noted, they are available in the same API version as this object.

ShiftChangeEvent (API version 54.0)

Change events are available for the object.

ShiftFeed

Feed tracking is available for the object.

ShiftHistory

History is available for tracked fields of the object.

ShiftOwnerSharingRule

Sharing rules are available for the object.

ShiftShare

Sharing is available for the object.

ShiftPattern

Represents a pattern of templates for creating shifts. This object is available in API version 51.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled. Users must have Field Service permission.

Fields

Field	Details
Description	Type textarea Properties Create, Nillable, Update Description A short description of the shift pattern to help users identify the pattern.
IsActive	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates that the shift pattern can be used to create shifts. The default value is 'false'.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date that the shift pattern was last used.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date that the shift pattern was last viewed.

Field	Details
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description A short, descriptive name of the shift pattern.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The ID of the owner of the shift pattern. Default is the user who creates the shift pattern. This is a polymorphic relationship field. Relationship Name Owner Relationship Type Lookup Refers To Group, User
PatternLength	Type int Properties Create, Filter, Group, Sort, Update Description The duration in days of the shift pattern.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ShiftPatternChangeEvent (API version 54.0)

Change events are available for the object.

ShiftPatternFeed

Feed tracking is available for the object.

ShiftPatternHistory

History is available for tracked fields of the object.

ShiftPatternShare

Sharing is available for the object.

ShiftPatternEntry

ShiftPatternEntry links a shift template to a shift pattern. This object is available in API version 51.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled. Users must have Field Service permission.

Fields

Field	Details
DayOrder	Type int Properties Create, Filter, Group, Sort, Update Description DayOrder links the shift template to the specific day within the shift pattern duration that the template. For example, if the DayOrder is 2 then a shift from the associated template is created on the second day of the pattern.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date that the shift pattern entry was last used.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date that the shift pattern entry was last viewed.
Name	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description An auto-generated reference number for the shift pattern entry.

Field	Details
ShiftPatternId	Type reference Properties Create, Filter, Group, Sort Description The ID of the shift pattern that the shift pattern entry is linked to. This is a relationship field. Relationship Name ShiftPattern Relationship Type Lookup Refers To ShiftPattern
ShiftTemplateId	Type reference Properties Create, Filter, Group, Sort, Update Description The ID of the shift template that's used to create shifts for this shift pattern entry. This is a relationship field. Relationship Name ShiftTemplate Relationship Type Lookup Refers To ShiftTemplate

Associated Objects

This object has the following associated objects. Unless noted, they are available in the same API version as this object.

ShiftPatternEntryChangeEvent (API version 54.0)

Change events are available for the object.

ShiftTemplate

Represents a template for creating shifts. This object is available in API version 51.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `update()`, `upsert()`

Special Access Rules

Field Service or Workforce Engagement must be enabled. For Field Service, users must have Field Service permission. For Workforce Engagement, the user needs to have a Workforce Engagement Admin or Planner permission set.

Fields

Field	Details
BackgroundColor	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description Sets a background color when shifts are displayed in the UI. Use a 3- or 6-digit hexadecimal format, for example #FF00FF. Available in API version 54.0 and later.
Description	Type textarea Properties Create, Filter, Group, Nillable, Sort, Update Description Additional information about the shift like number of breaks or activities.
Duration	Type double Properties Create, Filter, Sort, Update Description How long the shift lasts. The unit of measurement for this field is determined by <code>ShiftTemplateDurationType</code>.
IsActive	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether the shift is active or inactive.
IsNonStandard	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether the shift is nonstandard, such as overtime or on-call shifts.

Field	Details
	The default value is false. Available in API version 54.0 and later.
JobProfileId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The Job Profile record. This field is optional. This is a relationship field. Relationship Name JobProfile Relationship Type Lookup Refers To JobProfile
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the shift template was last modified. Its label in the user interface is Last Modified Date.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the shift template was last viewed.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The shift template record name.
OwnerId	Type reference

Field	Details
	Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The owner of the shift template. This is a polymorphic relationship field. Relationship Name Owner Relationship Type Lookup Refers To Group, User
RecordsetFilterCriteriaId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the recordset filter criteria selected for the shift template. Available in API version 53.0 and later. This is a relationship field. Relationship Name RecordsetFilterCriteria Relationship Type Lookup Refers To RecordsetFilterCriteria
ShiftTemplateDurationType	Type picklist Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update Description The unit of measurement for the shift template duration. Possible values are: • H—Hours • M—Minutes The default value is H.
StartTime	Type time

Field	Details
	Properties Create, Filter, Sort, Update Description The time of day when the shift starts.
TimeSlotType	Type picklist Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update Description The type of time slot. Possible values are: • Normal • Extended You can use <i>Extended</i> to represent overtime shifts. Available in API version 55.0 and later.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ShiftTemplateOwnerSharingRule

Sharing rules are available for the object.

ShiftTemplateShare

Sharing is available for the object.

ShiftTemplateChangeEvent

Change Data Capture events are available for the object. Available in API version 54.0 and later.

Shipment

Represents the transport of inventory in field service or a shipment of order items in Order Management.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

At least one of these features must be enabled:

- Order Management
- Field Service
- B2B Commerce
- Health Cloud Visit Inventory

- Consumer Goods Cloud Retail Execution

Fields

Field Name	Details
ActualDeliveryDate	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description Date the product was delivered.
DeliveredToId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The person or entity the product was delivered to. This is a polymorphic relationship field. Relationship Name DeliveredTo Relationship Type Lookup Refers To Group, User
DeliveryMethodId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The delivery method used for the shipment. This field is available in API version 51.0 and later.
Description	Type textarea Properties Create, Nillable, Update Description Details not recorded in the provided fields

Field Name	Details
DestinationLocationId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The place the product is to be delivered. This is a relationship field. Relationship Name DestinationLocation Relationship Type Lookup Refers To Location
ExpectedDeliveryDate	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description Date the product is expected to be delivered.
FulfillmentOrderId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The fulfillment order that the shipment belongs to. This field is available in API version 51.0 and later.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last interacted with this record, directly or indirectly.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort

Field Name	Details
	Description The timestamp when the current user last viewed this record or list view. If this value is null, it's possible that the user only accessed this record or list view (<code>LastReferencedDate</code>), but not viewed it.
OrderSummaryId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The order summary associated with the shipment. This field is available in API version 51.0 and later.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The owner of the shipment. This is a polymorphic relationship field. Relationship Name Owner Relationship Type Lookup Refers To Group, User
Provider	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update Description The company or person making the transfer.
ReturnOrderId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description For a return Shipment, the associated ReturnOrder. This field is available in API version 53.0 and later.

Field Name	Details
ShipFromAddress	Type address Properties Filter, Nillable Description The place the product is coming from. The compound form of the ship to address. Read-only. For details on compound address fields, see Address Compound Fields.
ShipFromCity	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The city of the address where the shipment originates.
ShipFromCountry	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The country of the address where the shipment originates.
ShipFromGeocodeAccuracy	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Accuracy level of the geocode for the address where the shipment originates. See Compound Field Considerations and Limitations for details on geolocation compound fields.  Note: This field is available in the API only.
ShipFromLatitude	Type double Properties Create, Filter, Nillable, Sort, Update Description Used with Longitude to specify the precise geolocation of the address where the shipment originates. Acceptable values are numbers between –90 and 90 with

Field Name	Details
	up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields.  Note: This field is available in the API only.
ShipFromLongitude	Type double Properties Create, Filter, Nillable, Sort, Update Description Used with Latitude to specify the precise geolocation of the address where the shipment originates. Acceptable values are numbers between –180 and 180 with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields.  Note: This field is available in the API only.
ShipFromPostalCode	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The postal code of the address where the shipment originates.
ShipFromState	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The state of the address where the shipment originates.
ShipFromStreet	Type textarea Properties Create, Filter, Group, Nillable, Sort, Update Description The street of the address where the shipment originates.
ShipToAddress	Type address Properties Filter, Nillable

Field Name	Details
	Description The physical address where the shipment is delivered. The compound form of the ship to address. Read-only. For details on compound address fields, see Address Compound Fields.
ShipToCity	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The city of the address where the shipment is delivered.
ShipToCountry	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The country of the address where the shipment is delivered.
ShipToGeocodeAccuracy	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Accuracy level of the geocode for the address where the shipment is delivered. See Compound Field Considerations and Limitations for details on geolocation compound fields.  Note: This field is available in the API only.
ShipToLatitude	Type double Properties Create, Filter, Nillable, Sort, Update Description Used with Longitude to specify the precise geolocation of the address where the shipment is delivered. Acceptable values are numbers between –90 and 90 with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields.  Note: This field is available in the API only.

Field Name	Details
ShipToLongitude	Type double Properties Create, Filter, Nillable, Sort, Update Description Used with Latitude to specify the precise geolocation of the address where the shipment is delivered. Acceptable values are numbers between –180 and 180 with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields.  Note: This field is available in the API only.
ShipToName	Type string Properties Create, Filter, Group, Sort, Update Description The shipment recipient.
ShipToPostalCode	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The postal code of the address where the shipment is delivered.
ShipToState	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The state of the address where the shipment is delivered.
ShipToStreet	Type textarea Properties Create, Filter, Group, Nillable, Sort, Update Description The street of the address where the shipment is delivered.
ShipmentNumber	Type string

Field Name	Details
	Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description An auto-generated number identifying the shipment.
SourceLocationId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The field service location where the shipment originates. This is a relationship field. Relationship Name SourceLocation Relationship Type Lookup Refers To Location
Status	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update Description The status of the shipment. The picklist includes the following values, which can be customized: • <i>Created</i>—Shipment has been created. • <i>Delivered</i>—Shipment has been delivered. • <i>In Transit</i>—Shipment is in transit. • <i>Shipped</i>—Order has been shipped. • <i>Voided</i>—Shipment has been cancelled.
TotalItemsQuantity	Type double Properties Filter, Nillable, Sort Description The total quantity of items included in the shipment. This value is calculated as the sum of the quantities of the shipment items in the shipment. This field is available in API version 51.0 and later.

Field Name	Details
TrackingNumber	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description Tracking number for the shipment.
TrackingUrl	Type url Properties Create, Filter, Group, Nillable, Sort, Update Description URL of website used for tracking the shipment.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ShipmentChangeEvent (API version 48.0)

Change events are available for the object.

ShipmentFeed

Feed tracking is available for the object.

ShipmentHistory

History is available for tracked fields of the object.

ShipmentOwnerSharingRule

Sharing rules are available for the object.

ShipmentShare

Sharing is available for the object.

Skill

Represents a category or group of Chat users or service resources in Field Service or Workforce Engagement. This object is available in API version 24.0 and later.



Note: For information about WDC skills on a user's profile, see the ProfileSkill topic.

Supported Calls

`create()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field Name	Details
Description	Type textarea Properties Create, Nillable, Update Description The description of the skill.
DeveloperName	Type string Properties Create, Filter, Group, Sort, Update Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. When creating large sets of data, always specify a unique <code>DeveloperName</code> for each record. If no <code>DeveloperName</code> is specified, performance slows down while Salesforce generates one for each record.
Language	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The language of the skill.
LastViewedDate	Type datetime Properties Filter, Nillable, Sort Description The timestamp for when the current user last viewed the skill.
MasterLabel	Type string

Field Name	Details
	Properties Create, Filter, Group, idLookup, Sort, Update Description The name of the skill.
TypeId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The skill type associated with the skill. This field is a relationship field. This field is available in API version 58.0 and later. Relationship Name Type Refers To SkillType

Usage

Chat

Use this object to assign Chat users to groups based on their abilities. The skills associated with a LiveChatButton determine which agents receive chat requests that come in through that button.

Field Service

Use this object to track certifications and areas of expertise in your workforce. After you create a skill, you can:

- Assign it to a service resource via the Skills related list on the resource's detail page. When you assign a skill to a service resource, you can specify their skill level and the duration of the skill.
- Add it as a required skill via the Skill Requirements related list on any work type, work order, or work order line item. When you add a required skill to a work record, you can specify the skill level.

Workforce Engagement

Use this object to specify areas of expertise in your workforce. After you create a skill, you can:

- Assign it to a service resource via the Skills related list on the resource's detail page.
- Add it as a required skill via the Skill Requirements related list on a job profile.

SkillRequirement

Represents a skill that is required to complete a particular task in Field Service, Omni-Channel, Salesforce Scheduler, or Workforce Engagement. Skill requirements can be added to pending service routing objects in Omni-Channel. They can be added to work types, work orders, and work order line items in Field Service and Lightning Scheduler. And they can be added to job profiles in Workforce Engagement. This object is available in API version 38.0 and later. You also can add skill requirements to work items in Omni-Channel skills-based routing using API version 42.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `update()`, `upsert()`

Special Access Rules

If you want to use SkillRequirement for Field Service use cases, then Field Service must be enabled.

If you want to use SkillRequirement only for Omni-Channel skills-based routing use cases, then you don't need Field Service to be enabled.

If you want to use SkillRequirement for Workforce Engagement use cases, then Workforce Engagement must be enabled.

Fields

Field Name	Details
<code>IsAdditionalSkill</code>	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates that a skill is additional. After a designated timeout period, a skill marked as additional is dropped from Omni-Channel routing. The case is then routed to the best-matched agent even if they don't have all the skills.
<code>LastReferencedDate</code>	Type dateTime Properties Filter, Nillable, Sort Description The timestamp for when the current user last viewed a record related to this record.
<code>LastViewedDate</code>	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.
<code>RelatedRecordId</code>	Type reference Properties Create, Filter, Group, Sort

Field Name	Details
	Description The record that the skill is required for. The related record can be a work order, work order line item, work type, or pending service routing record. This is a polymorphic relationship field. Relationship Name RelatedRecord Relationship Type Lookup Refers To WorkOrder, WorkOrderLineItem, WorkType
SkillId	Type reference Properties Create, Filter, Group, Sort, Update Description The skill that is required. This is a relationship field. Relationship Name Skill Relationship Type Lookup Refers To Skill
SkillLevel	Type double Properties Create, Defaulted on create, Filter, Nillable, Sort, Update Description The level of the skill required. Skill levels can range from zero to 99.99. Depending on your business needs, you can have the skill level to reflect years of experience, certification levels, or license classes.
SkillNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description An auto-generated number identifying the skill requirement.

Field Name	Details
SkillPriority	Type int Properties Aggregatable, Create, Filter, Group, Nillable, Sort, Update Description For additional skills, specify the order in which skills are dropped if after the specified timeout no agent with that skill is available. Higher priority-value skills are dropped first. Lower priority-value skills, for example 0, are dropped last. Skills with the same priority value are dropped as a group. You can set skill priority using skills-based routing rules or Apex code.

Usage

Field Service

Skill requirements help dispatchers assign work orders to service resources with the proper expertise. You can still assign a work order, work order line item, or related service appointment to a service resource that does *not* have the specified skills, so skill requirements serve more as a suggestion than a rule.



Note: If you're using the Field Service managed package, use matching rules to ensure that appointments are only assigned to service resources who possess the skills listed on the parent work order.

If many of your work orders require the same skills, add skill requirements to work types to save time and keep your processes consistent. When you add a skill requirement to a work type, work orders and work order line items that use that type automatically inherit the skill requirement. For example, if all annual maintenance visits for your Classic Refrigerator product require a Refrigerator Maintenance skill level of at least 50, add that skill requirement to the Annual Maintenance Visit work type. When you create a work order for a customer's annual fridge maintenance, applying that work type adds the skill requirement as well.

Omni-Channel

We recommend that you use Omni-Channel flow or skills-based routing rules to create skills-based routing requests. When you do so, work items are routed by creating a PendingServiceRouting object. The PendingServiceRouting object can have multiple SkillRequirements objects associated with it. When a work item requires multiple skills, it's routed to an agent who has all of the required skills. The PendingServiceRouting object adds attributes to the work item that represent the skill (skill id), priority, skill proficiency, and timestamp.

Workforce Engagement

Workforce Engagement uses skill requirements to assign shifts to agents who have the right skills. You can still assign shifts to service resources if they don't have those skills.

In a non-Omni workflow, create a scheduling rule that matches agents to shifts based on their skills and the job profile's skill requirements. Shift scheduling tools can then assign agents with the right skills.

Associated Objects

This object has the following associated objects. Unless noted, they're available in the same API version as this object.

SkillRequirementChangeEvent (API version 54.0)

Change events are available for the object.

SkillRequirementFeed

Feed tracking is available for the object.

SkillRequirementHistory

History is available for tracked fields of the object.

TimeSheet

Represents a schedule of a service resource's time in Field Service or Workforce Engagement. This object is available in API v47.0 and later.

Time sheets are composed of time sheet entries, which typically track individual tasks like travel or asset repair.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service or Workforce Engagement must be enabled.

Fields

Field Name	Details
CurrencyIsoCode	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Available only if the multicurrency feature is enabled. Contains the ISO code for any currency allowed by the organization. The label in the user interface is <code>Currency ISO Code</code> .
EndDate	Type date Properties Create, Filter, Group, Sort, Update Description The last day the time sheet covers.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last interacted with this record, directly or indirectly.

Field Name	Details
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last viewed this record or list view. If this value is null, it's possible that the user only accessed this record or list view (LastReferencedDate), but not viewed it.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The owner of the time sheet.
ServiceResourceId	Type reference Properties Create, Filter, Group, Sort, Update Description The service resource whose time is being tracked with the time sheet.
StartDate	Type date Properties Create, Filter, Group, Sort, Update Description The first day the time sheet covers.
Status	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description The status of the time sheet. The picklist includes the following values, which can be customized: • New • Submitted • Approved

Field Name	Details
TimeSheetEntryCount	Type int Properties Filter, Group, Nillable, Sort Description (Read Only) The number of related time sheet entries.
TimeSheetNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description An auto-generated number identifying the time sheet.
TotalDurationInHours	Type double Properties Filter, Nillable, Sort Description Represents the sum total of the duration field of all the time sheet entries related to the time sheet object in hours.
TotalDurationInMinutes	Type int Properties Filter, Group, Nillable, Sort Description Represents the sum total of the duration field of all the time sheet entries related to the time sheet object in minutes.

Associated Objects

This object has the following associated objects. Unless noted, they are available in the same API version as this object.

TimeSheetChangeEvent (API version 48.0)

Change events are available for the object.

TimeSheetFeed

Feed tracking is available for the object.

TimeSheetHistory

History is available for tracked fields of the object.

TimeSheetOwnerSharingRule

Sharing rules are available for the object.

TimeSheetShare

Sharing is available for the object.

TimeSheetEntry

Represents a span of time that a service resource spends on a field service task. This object is available in API version 47.0 and later.

Time sheets are composed of time sheet entries. Time sheet entries typically track individual tasks like travel or asset repair.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field Name	Details
CurrencyIsoCode	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Available only if the multicurrency feature is enabled. Contains the ISO code for any currency allowed by the organization. The label in the user interface is <code>Currency ISO Code</code>. Time sheet entries inherit their time sheet's currency code. Updates to a time sheet's currency code aren't reflected in existing time sheet entries' currency code.
Description	Type textarea Properties Create, Nillable, Update Description Notes on how the time was spent. For example, "This service took longer than normal because the machine was jammed."
DurationInMinutes	Type int Properties Filter, Group, Nillable, Sort

Field Name	Details
	Description Minutes recorded on the time sheet entry.
EndTime	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description The date and time the activity finished.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last interacted with this record, directly or indirectly.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last viewed this record or list view. If this value is null, it's possible that the user only accessed this record or list view (LastReferencedDate), but not viewed it.
LocationTimeZone	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Time zone of the location where the activity occurred. This field is available in API version 50.0 and later.
StartTime	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description The date and time the activity began.

Field Name	Details
Status	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description The status of the time sheet entry. The picklist includes the following values, which can be customized: • New • Submitted • Approved
Subject	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description Activity performed; for example, repair, lunch, or travel.
TimeSheetEntryNumber	Type string Properties Autonumber, Defaulted on create, Filter, Sort Description An auto-generated number identifying the time sheet entry.
TimeSheetId	Type reference Properties Create, Filter, Group, Sort Description The time sheet associated with the time sheet entry.
Type	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description The type of work performed. The picklist includes the following values, which can be customized: • Direct • Indirect

Field Name	Details
WorkOrderId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The work order related to the time sheet entry. Work orders are searchable by their content.
WorkOrderLineItemId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The work order line item related to the time sheet entry. Work order line items are searchable by their content.

Associated Objects

This object has the following associated objects. Unless noted, they are available in the same API version as this object.

TimeSheetEntryChangeEvent (API version 48.0)

Change events are available for the object.

TimeSheetEntryFeed

Feed tracking is available for the object.

TimeSheetEntryHistory

History is available for tracked fields of the object.

TimeSlot

Represents a period of time on a specified day of the week during which work can be performed in Field Service, Salesforce Scheduler, or Workforce Engagement. Operating hours consist of one or more time slots. This object is available in API version 38.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field Name	Details
DayOfWeek	Type picklist

Field Name	Details
	Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update Description The day of the week when the time slot takes place.
EndTime	Type time Properties Create, Filter, Sort, Update Description The time when the time slot ends.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp for when the current user last viewed a record related to this record.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (<code>LastReferencedDate</code>) and not viewed.
MaxAppointments	Type int Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description Maximum number of appointments for a single time slot. Available in API version 47.0 and later.
OperatingHoursId	Type reference Properties Create, Filter, Group, Sort

Field Name	Details
	Description The operating hours that the time slot belongs to. An operating hours' time slots appear in the Operating Hours related list. This is a relationship field. Relationship Name OperatingHours Relationship Type Lookup Refers To OperatingHours
StartTime	Type time Properties Create, Filter, Sort, Update Description The time when the time slot starts.
RecordSetFilterCriteriaId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the recordset filter criteria selected for the time slot. This is a relationship field. Relationship Name RecordsetFilterCriteria Relationship Type Lookup Refers To RecordsetFilterCriteria
TimeSlotNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description The name of the time slot. The name is auto-populated to a day and time format—for example, Monday 9:00 AM – 10:00 PM—but you can manually update it if you wish.

Field Name	Details
Type	Type picklist Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update Description The type of time slot. Possible values are <i>Normal</i> and <i>Extended</i>. You may choose to use <i>Extended</i> to represent overtime shifts.
WorkTypeGroupId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description Work type group assigned to the time slot. Available in API version 47.0 and later. This is a relationship field. Relationship Name WorkTypeGroup Relationship Type Lookup Refers To WorkTypeGroup

Usage

Operating hours are composed of time slots, which indicate the hours of operation for a particular day. After you create operating hours, create time slots for each day. For example, if the operating hours should be 8 AM to 5 PM Monday through Friday, create five time slots, one per day. To reflect breaks such as lunch hours, create multiple time slots in a day: for example, *Monday 8:00 AM – 12:00 PM* and *Monday 1:00 PM – 5:00 PM*.



Tip: Time slots don't come with any built-in rules, but you can create Apex triggers that limit time slot settings in your org. For example, you may want to restrict the start and end times on time slots to half-hour increments, or to prohibit end times later than 8 PM.

Associated Objects

This object has the following associated objects. Unless noted, they are available in the same API version as this object.

TimeSlotChangeEvent (API version 54.0)

Change events are available for the object.

TimeSlotHistory (API version 62.0)

History is available for tracked fields of the object.

TravelMode

Represents a travel mode used for travel time calculations. The records include information about the type of transportation (such as Car or Walking), whether a vehicle can take toll roads, and whether a vehicle is transporting hazardous materials. This object is available in API version 54.0 and later.

Fields

Field	Details
CanUseTollRoads	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates if the vehicle is allowed to drive on toll roads. The default value is <code>false</code> .
IsLocked	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Indicates whether the travel model record is locked or not. The default value is <code>false</code> .
IsTransportingHazmat	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates if the vehicle is transporting hazardous materials. The default value is <code>false</code> .
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last accessed this record, a record related to this record, or a list view.
LastViewedDate	Type dateTime

Field	Details
	Properties Filter, Nillable, Sort Description The timestamp when the current user last viewed this record or list view. If this value is null, the user might have only accessed this record or list view (<code>LastReferencedDate=</code>) but not viewed it.
MayEdit	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Indicates whether the travel model record can be edited or not. The default value is <code>false</code>.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description Name of the travel mode.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description ID of the owner of this object. This field is a polymorphic relationship field. Relationship Name Owner Relationship Type Lookup Refers To Group, User
TransportType	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details
	Description Type of transportation. Possible values are: • Bicycle • Car-Default. • Heavy Truck • Light Truck • Walking

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

TravelModeFeed

Feed tracking is available for the object.

TravelModeOwnerSharingRule

Sharing rules are available for the object.

TravelModeShare

Sharing is available for the object.

WarrantyTerm

Represents warranty terms defining the labor, parts, and expenses covered, along with any exchange options, provided to rectify issues with products. This object is available in API version 50.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
Code	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description A code or other identifier associated with this warranty term.

Field	Details
Description	Type textarea Properties Create, Nillable, Update Description Description of the warranty term.
EffectiveStartDate	Type picklist Properties Create, Filter, Group, Sort, Update Description Date on which the warranty term became available for use. Possible values are: • InstallDate • ManufactureDate • PurchaseDate
ExchangeType	Type picklist Properties Create, Filter, Group, Nillable, Sort, Update Description The type of exchange offered. Possible values are: • AdvanceExchange • Loaner • ReturnExchange
Exclusions	Type textarea Properties Create, Nillable, Update Description Description of any exclusions.
ExpensesCovered	Type percent Properties Create, Filter, Nillable, Sort, Update

Field	Details
	Description The percentage of expenses covered.
ExpensesCoveredDuration	Type int Properties Create, Filter, Group, Nillable, Sort, Update Description The duration for which expenses are covered.
ExpensesCoveredUnitOfTime	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The unit in which expenses covered duration is measured. Possible values are: • Days • Months • Weeks • Years
IsActive	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Defines whether the warranty term is active.
IsTransferable	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Defines whether the warranty can be transferred to a new owner.
LaborCovered	Type percent Properties Create, Filter, Nillable, Sort, Update

Field	Details
	Description The percentage of labor covered.
LaborCoveredDuration	Type int Properties Create, Filter, Group, Nillable, Sort, Update Description The duration for which labor is covered.
LaborCoveredUnitOfTime	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The unit in which labor covered duration is measured. Possible values are: • Days • Months • Weeks • Years
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the warranty term was last modified. Its label in the user interface is <code>Last Modified Date</code> .
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the warranty term was last viewed.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update

Field	Details
	Description The warranty term's assigned owner.
PartsCovered	Type percent Properties Create, Filter, Nillable, Sort, Update Description The percentage of parts covered.
PartsCoveredDuration	Type int Properties Create, Filter, Group, Nillable, Sort, Update Description The duration for which parts are covered.
PartsCoveredUnitOfTime	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The unit in which parts covered duration is measured. Possible values are: • Days • Months • Weeks • Years
Pricebook2Id	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the price book item associated with this warranty term.
WarrantyDuration	Type int Properties Create, Filter, Group, Sort, Update

Field	Details
	Description The duration of the warranty offered by this term.
WarrantyTermName	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The name of the warranty term.
WarrantyType	Type picklist Properties Create, Filter, Group, Sort, Update Description The type of warranty. Possible values are: • Repair • Standard • Supplier
WarrantyUnitOfTime	Type picklist Properties Create, Filter, Group, Restricted picklist, Sort, Update Description The unit in which the warranty duration is measured. Possible values are: • Days • Months • Weeks • Years

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

WarrantyTermChangeEvent

Change events are available for the object.

WorkCapacityAvailability

Represents the available work capacity for a specific time and service territory. This object is available in API version 59.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
AvailCapacityHours	Type double Properties Create, Filter, Nillable, Sort, Update Description The number of available capacity in hours in the time frame the user defined for a service territory.
AvailCapacityMinutes	Type double Properties Create, Filter, Nillable, Sort, Update Description The number of available capacity in minutes in the time frame the user defined for a service territory.
EndDate	Type date Properties Create, Filter, Group, Nillable, Sort, Update Description The end date of the total available capacity.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last accessed this record indirectly, for example, through a list view or related record.
LastViewedDate	Type dateTime

Field	Details
	Properties Filter, Nillable, Sort Description The timestamp when the current user last viewed this record or list view. If this value is null, and <code>LastReferenceDate</code> isn't null, the user accessed this record or list view indirectly.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The ID of the owner of this object. This field is a polymorphic relationship field. Relationship Name Owner Relationship Type Lookup Refers To Group, User
ServiceTerritoryId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the service territory of the work capacity availability calculation. This field is a relationship field. Relationship Name ServiceTerritory Relationship Type Lookup Refers To ServiceTerritory
StartDate	Type date Properties Create, Filter, Group, Nillable, Sort, Update Description The start date of the total available capacity.

Field	Details
TimePeriod	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The time period the user selected when creating the work capacity limit. The value is copied from the <code>TimePeriod</code> field of the <code>WorkCapacityLimit</code> object. Possible values are: • Day The default value is <code>Day</code>.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

WorkCapacityAvailabilityFeed

Feed tracking is available for the object.

WorkCapacityAvailabilityShare

Sharing is available for the object.

WorkCapacityLimit

Represents the capacity limit in a specific service territory for a workstream or for the whole service territory in a given period. This object is available in API version 59.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
CapacityLimitRelaxation	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description Defines the relaxation behavior that determines the limit override policy for this limit if the limit override policy is set per limit in Field Service Settings. If the limit override policy isn't set per limit, this value is ignored. Valid strings are:

Field	Details
	• Empty value - no limit override • 0 - limit override starts at midnight on the day of service • Positive integer - limit override starts this number of hours after midnight. the maximum value is 23. • Negative integer - limit override starts this number of hours before midnight. The maximum value is 336.
Description	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description Description of the work capacity limit.
EndDate	Type date Properties Create, Filter, Group, Nillable, Sort, Update Description End date of the work capacity limit. If no <code>EndDate</code> is set this work capacity limit is without an expiration date.
IsActive	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether the work capacity limit is active or inactive. When creating a record, save the record, and then activate it. You can't update fields in an active record. The default value is <code>false</code>.
IsFriday	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether the limitation is applied on Fridays. The default value is <code>false</code>.
IsMonday	Type boolean

Field	Details
	Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether the limitation is applied on Mondays. The default value is <code>false</code>.
IsSaturday	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether the limitation is applied on Saturdays. The default value is <code>false</code>.
IsSunday	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether the limitation is applied on Sundays. The default value is <code>false</code>.
IsSvcTerrOnlyLimit	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort Description Apply this work capacity limit to the entire service territory. The default value is <code>false</code>.
IsThursday	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether the limitation is applied on Thursdays. The default value is <code>false</code>.
IsTuesday	Type boolean

Field	Details
	Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether the limitation is applied on Tuesdays. The default value is <code>false</code>.
IsWednesday	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether the limitation is applied on Wednesdays. The default value is <code>false</code>.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last accessed this record indirectly, for example, through a list view or related record.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last viewed this record or list view. If this value is null, and <code>LastReferenceDate</code> isn't null, the user accessed this record or list view indirectly.
LimitationUnits	Type picklist Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update Description Unit of the <code>LimitationValue</code>. Possible values are: - Hours - Percentage The default value is <code>Hours</code>.

Field	Details
LimitationValue	Type double Properties Create, Filter, Sort, Update Description If the <code>LimitationUnits</code> is <code>Hours</code>, <code>LimitationValue</code> is the threshold that represents how many hours of total work capacity can be scheduled for a specific workstream in a service territory. Enter the number of hours for the daily limitation as a whole number. If the <code>LimitationUnits</code> is <code>Percentage</code> this threshold represents the percentage of the total work capacity that can be scheduled for a specific workstream in a service territory. Enter the percentage for the daily limitation as a whole number.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description ID of the creator of the work capacity limit. This field is a polymorphic relationship field. Relationship Name Owner Relationship Type Lookup Refers To Group, User
ServiceTerritoryId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description ID of the service territory of the work capacity workstream for which the limit is defined. This field is a relationship field. Relationship Name ServiceTerritory Relationship Type Lookup Refers To ServiceTerritory

Field	Details
StartDate	Type date Properties Create, Filter, Group, Sort, Update Description The start date of the Work Capacity Limit.
SvcApptField	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Work-specific criteria used to define the capacity limit for the workstream. The service appointment field is set for the organization when the first work capacity limit instance is created. Possible values are: • <code>ServiceAppointment.AccountId</code> • <code>ServiceAppointment.ActualDuration</code>—Actual duration (in minutes) • <code>ServiceAppointment.Duration</code>—Duration • <code>ServiceAppointment.DurationInMinutes</code> • <code>ServiceAppointment.FSL__Appointment_Grade__ce</code> • <code>ServiceAppointment.FSL__Auto_Schedule__c</code> • <code>ServiceAppointment.FSL__Duration_In_Minutes__c</code>—Scheduled duration • <code>ServiceAppointment.FSL__Emergency__c</code> • <code>ServiceAppointment.FSL__GanttColor__c</code> • <code>ServiceAppointment.FSL__GanttLabel__c</code> • <code>ServiceAppointment.FSL__InJeopardyReason__c</code> • <code>ServiceAppointment.FSL__InJeopardy__c</code> • <code>ServiceAppointment.FSL__IsFillInCandidate__c</code> • <code>ServiceAppointment.FSL__IsMultiDay__c</code> • <code>ServiceAppointment.FSL__Last_Updated_Epoch__c</code> • <code>ServiceAppointment.FSL__MDS_Calculated_length__c</code>—Multiday work calculated length • <code>ServiceAppointment.FSL__Pinned__c</code> • <code>ServiceAppointment.FSL__Prevent_Geocoding_For_Chatter_Actions__c</code> • <code>ServiceAppointment.FSL__Related_Service__c</code> • <code>ServiceAppointment.FSL__Same_Day__c</code> • <code>ServiceAppointment.FSL__Same_Resource__c</code>

Field	Details
	• <code>ServiceAppointment.FSL__Schedule_Mode__c</code> • <code>ServiceAppointment.FSL__Schedule_over_lower_priority_appointment__c</code> • <code>ServiceAppointment.FSL__Scheduling_Policy_Used__c</code> • <code>ServiceAppointment.FSL__Time_Dependency__c</code> • <code>ServiceAppointment.FSL__UpdatedByOptimization__c</code> • <code>ServiceAppointment.FSL__Use_Async_Logic__c</code> • <code>ServiceAppointment.FSL__Virtual_Service_For_Chatter_Action__c</code> • <code>ServiceAppointment.IsOffsiteAppointment</code> • <code>ServiceAppointment.Subject</code> • <code>ServiceAppointment.WorkTypeId</code>—Work Type ID
<code>SvcApptFieldValDplyNm</code>	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The workstream display name of <code>SvcApptFieldValue</code>. If <code>SvcApptField</code> is a lookup to a service appointment, <code>SvcApptFieldValue</code> is an ID and the display name describes the value for the user.
<code>SvcApptFieldValue</code>	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The value of <code>SvcApptField</code>, the work-specific criteria of the capacity limit.
<code>TimePeriod</code>	Type picklist Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update Description Duration for defining the capacity limitation on the workstream in the service territory. Possible values are: • Day The default value is Day.
<code>WorkCapacityLimitNumber</code>	Type string

Field	Details
	Properties Autonumber, Defaulted on create, Filter, idLookup, Sort
	Description Read-only. Auto populated, unique identifying number.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

WorkCapacityLimitChangeEvent (API version 62.0)

Change events are available for the object.

WorkCapacityLimitFeed

Feed tracking is available for the object.

WorkCapacityLimitHistory

History is available for tracked fields of the object.

WorkCapacityLimitShare

Sharing is available for the object.

WorkCapacityUsage

Represents the capacity usage in a specific service territory for a workstream or for the whole service territory in a given period. This object is available in API version 59.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
AvailCapacityHours	Type double
	Properties Create, Defaulted on create, Filter, Nillable, Sort, Update
	Description This value is copied from the <code>AvailCapacityHours</code> of the <code>WorkCapacityAvailability</code> object for the service territory on the same date.
CapacityLimitRelaxation	Type string

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update Description Defines the relaxation behavior that determines the limit override policy for this limit if the limit override policy is set per limit in Field Service Settings. If the limit override policy isn't set per limit, this value is ignored. Valid strings are: • Empty value - no limit override • 0 - limit override starts at midnight on the day of service • Positive integer - limit override starts this number of hours after midnight. the maximum value is 23. • Negative integer - limit override starts this number of hours before midnight. The maximum value is 336.
ConsumptionToLimitRatio	Type double Properties Create, Defaulted on create, Filter, Nillable, Sort Description (Time consumed in hours / Limit in hours) * 100 Note the following exceptions. • If a limit isn't defined (-1) the ratio is -1 (even if consumption is 0 or higher). • If consumption is 0, and the limit is a number greater than 0, then the ration is 0. • If consumption is 0 and the limit is 0, the ration is 100% hard-coded. • If consumption is greater than 0 and the limit is 0, the ration is calculated as if the limit = 0.99 in order to get a result that's higher than 100%.
EndDate	Type date Properties Create, Filter, Group, Sort Description End date of the time period for which the capacity usage is accumulated.
IsSvcTerrOnlyLimit	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort Description Applies this work capacity limit to the entire service territory. The default value is <code>false</code>.

Field	Details
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last accessed this record indirectly, for example, through a list view or related record.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last viewed this record or list view. If this value is null, and LastReferenceDate isn't null, the user accessed this record or list view indirectly.
LimitationPercentage	Type double Properties Create, Defaulted on create, Filter, Nillable, Sort Description If the LimitationUnits is Percentage this value is copied from the LimitationValue field of the WorkCapacityLimit object.
LimitationUnits	Type picklist Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort Description Defines whether the limitation for the workstream in the service territory is in hours or as a percentage of all the available hours for all the workstreams for which limitations exist in the service territory on a specific day. Possible values are: - Hours - Percentage The default value is Hours.
LimitationValue	Type double Properties Create, Filter, Sort

Field	Details
	Description The <code>LimitationValue</code> depends on the <code>LimitationUnit</code>. If the <code>LimitationUnit</code> is <code>Hours</code> the value is copied from <code>LimitationValue</code> in the <code>WorkCapacityLimit</code> object. If the <code>LimitationUnit</code> is <code>Percentage</code>, the percentage is calculated relative to the availability in the <code>WorkCapacityAvailability</code> object.
<code>OriginalLimit</code>	Type double Properties Create, Defaulted on create, Filter, Nillable, Sort, Update Description If the limit value is changed after the work capacity usage was created, this parameter is the original value.
<code>OwnerId</code>	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description ID of the owner of this object. This field is a polymorphic relationship field. Relationship Name Owner Relationship Type Lookup Refers To Group, User
<code>ServiceTerritoryId</code>	Type reference Properties Create, Filter, Group, Nillable, Sort Description ID of the service territory of the work capacity workstream for which usage is accumulated. This field is a relationship field. Relationship Name ServiceTerritory Relationship Type Lookup Refers To ServiceTerritory

Field	Details
StartDate	Type date Properties Create, Filter, Group, Sort Description Start date of the time period for which the capacity usage is accumulated.
SvcAptField	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort Description Work-specific criteria used to define the capacity limit for the workstream. Possible values are: • <code>ServiceAppointment.AccountId</code> • <code>ServiceAppointment.ActualDuration</code>—Actual duration (in minutes) • <code>ServiceAppointment.Duration</code>—Duration • <code>ServiceAppointment.DurationInMinutes</code> • <code>ServiceAppointment.FSL__Appointment_Grade__c</code> • <code>ServiceAppointment.FSL__Auto_Schedule__c</code> • <code>ServiceAppointment.FSL__Duration_In_Minutes__c</code>—Scheduled duration • <code>ServiceAppointment.FSL__Emergency__c</code> • <code>ServiceAppointment.FSL__GanttColor__c</code> • <code>ServiceAppointment.FSL__GanttLabel__c</code> • <code>ServiceAppointment.FSL__InJeopardyReason__c</code> • <code>ServiceAppointment.FSL__InJeopardy__c</code> • <code>ServiceAppointment.FSL__IsFillInCandidate__c</code> • <code>ServiceAppointment.FSL__IsMultiDay__c</code> • <code>ServiceAppointment.FSL__Last_Updated_Epoch__c</code> • <code>ServiceAppointment.FSL__MDS_Calculated_length__c</code>—Multiday work calculated length • <code>ServiceAppointment.FSL__Pinned__c</code> • <code>ServiceAppointment.FSL__Prevent_Geocoding_For_Chatter_Actions__c</code> • <code>ServiceAppointment.FSL__Related_Service__c</code> • <code>ServiceAppointment.FSL__Same_Day__c</code> • <code>ServiceAppointment.FSL__Same_Resource__c</code> • <code>ServiceAppointment.FSL__Schedule_Mode__c</code> • <code>ServiceAppointment.FSL__Schedule_over_lower_priority_appointment__c</code>

Field	Details
	• <code>ServiceAppointment.FSL__Scheduling_Policy_Used__c</code> • <code>ServiceAppointment.FSL__Time_Dependency__c</code> • <code>ServiceAppointment.FSL__UpdatedByOptimization__c</code> • <code>ServiceAppointment.FSL__Use_Async_Logic__c</code> • <code>ServiceAppointment.FSL__Virtual_Service_For_Chatter_Action__c</code> • <code>ServiceAppointment.IsOffsiteAppointment</code> • <code>ServiceAppointment.Subject</code> • <code>ServiceAppointment.WorkTypeId</code>—Work Type ID
<code>SvcApptFieldValDplyNm</code>	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description Workstream display name of <code>SvcApptFieldValue</code>.
<code>SvcApptFieldValue</code>	Type string Properties Create, Filter, Group, Nillable, Sort Description Value of <code>SvcApptField</code>, the work-specific criteria of the capacity limit.
<code>TimeConsumedInHours</code>	Type double Properties Create, Filter, Nillable, Sort Description Time consumed in hours by the workstream in the service territory for the defined period. This value is calculated by dividing <code>TimeConsumedInMinutes</code> by 60.
<code>TimeConsumedInMinutes</code>	Type double Properties Create, Filter, Sort Description Time consumed in minutes by the workstream in the service territory for the defined period.
<code>TimePeriod</code>	Type picklist

Field	Details
	Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort Description Duration for defining the capacity limitation on the workstream in the service territory. Possible values are: • Day The default value is Day.
WcuUniqueField1	Type string Properties Create, Filter, Group, idLookup, Nillable, Sort, Update Description Read-only. Auto populated, unique identifying number.
WcuUniqueField2	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description Read-only. Auto populated, unique identifying number.
WorkCapacityUsageNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description Read-only. Auto populated, unique identifying number.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

WorkCapacityUsageFeed

Feed tracking is available for the object.

WorkCapacityUsageShare

Sharing is available for the object.

WorkOrder

Represents field service work to be performed for a customer. This object is available in API version 36.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

- Work orders or Field Service must be enabled.
- The following fields can't be edited, regardless of your field-level security settings:
 - Discount
 - GrandTotal
 - IsGeneratedFromMaintenancePlan
 - RootWorkOrderId

Fields

Field Name	Details
AccountId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The account associated with the work order. This is a relationship field. Relationship Name Account Relationship Type Lookup Refers To Account
Address	Type address Properties Filter, Nillable Description The compound form of the address where the work order is completed.
AssetId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update

Field Name	Details
	Description The asset associated with the work order. This is a relationship field. Relationship Name Asset Relationship Type Lookup Refers To Asset
AssetWarrantyId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The asset warranty term associated with the work order. This field is available in API version 50.0 and above.
BusinessHoursId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The business hours associated with the work order. This is a relationship field. Relationship Name BusinessHours Relationship Type Lookup Refers To BusinessHours
CaseId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The case associated with the work order. This is a relationship field. Relationship Name Case

Field Name	Details
	Relationship Type Lookup
	Refers To Case
City	Type string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The city where the work order is completed. Maximum length is 40 characters.
ContactId	Type reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The contact associated with the work order. This is a relationship field.
	Relationship Name Contact
	Relationship Type Lookup
	Refers To Contact
Country	Type string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The country where the work order is completed. Maximum length is 80 characters.
CurrencyIsoCode	Type picklist
	Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description Available only for orgs with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the organization. The label in the user interface is <code>Currency ISO Code</code> .

Field Name	Details
Description	Type textarea Properties Create, Nillable, Update Description The description of the work order. Try to include the steps needed to change the work order's status to Completed.
Discount	Type percent Properties Filter, Nillable, Sort Description Read only. The weighted average of the discounts on all line items in the work order. It can be any positive number up to 100.
Duration	Type double Properties Create, Filter, Nillable, Sort, Update Description The estimated time required to complete the work order. Specify the duration unit in the <code>Duration Type</code> field. If the <code>Duration</code> field on a Work Order is null, it adopts the duration value from the Work Type object when the work type is updated or inserted. Work order duration and work order line item duration are independent of each other. If you want work order duration to automatically show the sum of the work order line items' duration, replace the Duration field on work orders with a custom roll-up summary field.
DurationInMinutes	Type double Properties Filter, Nillable, Sort Description The estimated duration in minutes. For internal use only.
DurationType	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field Name	Details
	Description The unit of the duration: Minutes or Hours.
EndDate	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description The date when the work order is completed. This field is blank unless you set up an Apex trigger or quick action to populate it. For example, you can create a quick action that sets the <code>EndDate</code> to 365 days after the <code>StartDate</code> .
EntitlementId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The entitlement associated with the work order.
GeocodeAccuracy	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Accuracy level of the geocode for the address. See Compound Field Considerations and Limitations for details on geolocation compound fields.
GrandTotal	Type currency Properties Filter, Nillable, Sort Description Read only. The total price of the work order with tax added.
IsClosed	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Indicates whether the work order is closed (<code>true</code>) or open (<code>false</code>).

Field Name	Details
IsGeneratedFromMaintenancePlan	Type boolean Properties Defaulted on create, Filter, Group, Sort Description (Read Only) Indicates that the work order was generated from a maintenance plan (<code>true</code>), rather than manually created (<code>false</code>).
IsStopped	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether a milestone is paused (<code>true</code>) or counting down (<code>false</code>). This field is available only if Enable stopped time and actual elapsed time is selected on the Entitlement Settings page.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the work order was last modified. Its label in the user interface is <code>Last Modified Date</code>.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the work order was last viewed.
Latitude	Type double Properties Create, Filter, Nillable, Sort, Update Description Used with Longitude to specify the precise geolocation of the address where the work order is completed. Acceptable values are numbers between <code>-90</code> and <code>90</code> with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields.

Field Name	Details
LineItemCount	Type int Properties Filter, Group, Nillable, Sort Description The number of work order line items in the work order. Its label in the user interface is <code>Line Items</code>.
LocationId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The location associated with the work order. For example, a work site. This is a relationship field. Relationship Name Location Relationship Type Lookup Refers To Location
Longitude	Type double Properties Create, Filter, Nillable, Sort, Update Description Used with Latitude to specify the precise geolocation of the address where the work order is completed. Acceptable values are numbers between –180 and 180 with up to 15 decimal places. See Compound Field Considerations and Limitations for details on geolocation compound fields.
MaintenancePlanId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The maintenance plan associated with the work order. When the work order is auto-generated from a maintenance plan, this field automatically lists the related plan.

Field Name	Details
MaintenanceWorkRuleId	Type reference Properties Filter, Group, Nillable, Sort Description ID of the maintenance work rule that generated this work order. This field is available in API version 50.0 and above.
MilestoneStatus	Type string Properties Group, Nillable, Sort Description Indicates the status of a milestone. This field is visible if an entitlement process is applied to a work order.
MinimumCrewSize	Type int Properties Create, Filter, Group, Nillable, Sort, Update Description The minimum crew size allowed for a crew assigned to the work order. If you're not using the Field Service managed package, this field serves as a suggestion rather than a rule. If you are using the managed package, the scheduling optimizer counts the number of service crew members on a service crew to determine whether it fits a work order's minimum crew size requirement.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The work order's assigned owner. This is a polymorphic relationship field. Relationship Name Owner Relationship Type Lookup Refers To Group, User

Field Name	Details
ParentWorkOrderId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The work order's parent work order, if it has one. Create a custom report to view a work order's child work orders. This is a relationship field. Relationship Name ParentWorkOrder Relationship Type Lookup Refers To WorkOrder
PostWorkSummary	Type textarea Properties Create, Nillable, Update Description The summary of a completed work order that's either entered manually or created by an AI agent.
PostalCode	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The postal code where the work order is completed. Maximum length is 20 characters.
PreWorkBriefPromptTemplate	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the activated Pre-Work Brief prompt template.
Pricebook2Id	Type reference

Field Name	Details
	Properties Create, Filter, Group, Nillable, Sort, Update Description The price book associated with the work order. Adding a price book to the work order lets you assign different price book entries to the work order's line items. This is only available if Product2 is enabled. This is a relationship field. Relationship Name Pricebook2 Relationship Type Lookup Refers To Pricebook2
Priority	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description The priority of the work order. The picklist includes the following values, which can be customized: • Low • Medium • High • Critical
ProductServiceCampaignId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The product service campaign associated with the work order.
ProductServiceCampaignItemId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The product service campaign item associated with the work order.

Field Name	Details
RecommendedCrewSize	Type int Properties Create, Filter, Group, Nillable, Sort, Update Description The recommended number of people on the service crew assigned to the work order. For example, you might have a Minimum Crew Size of 2 and a Recommended Crew Size of 3.
ReturnOrderId	Type reference Properties Filter, Group, Nillable, Sort Description The return order associated with the work order.
ReturnOrderLineItemId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The return order line item associated with the work order.
RootWorkOrderId	Type reference Properties Filter, Group, Nillable, Sort Description (Read only) The top-level work order in a work order hierarchy. Depending on where a work order lies in the hierarchy, its root could be the same as its parent. View a work order's child work order in the Child Work Orders related list. This is a relationship field. Relationship Name RootWorkOrder Relationship Type Lookup Refers To WorkOrder
ServiceAppointmentCount	Type int

Field Name	Details
	Properties Filter, Group, Nillable, Sort Description The number of service appointments on the work order.
ServiceContractId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The service contract associated with the work order.
ServiceDocumentTemplate	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The template ID which sets the template for each service document for the Document Builder feature.
ServiceReportLanguage	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The language used for all service reports and service report previews created for the work order, its service appointments, and its work order line items and their service appointments. If the field is blank, service reports are generated in the default language in Salesforce of the person creating the report. To appear as an option in the ServiceReportLanguage field, a language must be set up in Translation Workbench or be one of Salesforce's 18 fully supported languages. Rich text fields and service report section names aren't translated.
ServiceReportTemplateId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The service report template that the work order uses. If you don't specify a service report template on a work order, it uses the service report template listed on its work type. If the work type doesn't list a template or no work type is specified, the work order uses the default service report template.

Field Name	Details
ServiceTerritoryId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The service territory where the work order is taking place. This is a relationship field. Relationship Name ServiceTerritory Relationship Type Lookup Refers To ServiceTerritory
SlaExitDate	Type dateTime Properties Filter, Nillable, Sort Description The time that the work order exits the entitlement process.
SlaStartDate	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description The time that the work order enters the entitlement process. You can update or reset the time if you have “Edit” permission on work orders.
StartDate	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description The date when the work order goes into effect. This field is blank unless you set up an Apex trigger or quick action to populate it. For example, you can create a quick action that sets the StartDate to the date when the Status changes to In Progress.
State	Type string

Field Name	Details
	Properties Create, Filter, Group, Nillable, Sort, Update Description The state where the work order is completed. Maximum length is 80 characters.
Status	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description The status of the work order. The picklist includes the following values, which can be customized: • New—Work order was created, but there hasn't yet been any activity. • In Progress—Work has begun. • On Hold—Work is paused. • Completed—Work is complete. • Cannot Complete—Work could not be completed. • Closed—All work and associated activity is complete. • Canceled—Work is canceled, typically before any work began. Changing a work order's status does not affect the status of its work order line items or associated service appointments.
StatusCategory	Type picklist Properties Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort Description The category that each Status value falls into. The Status Category field has eight default values: seven values which are identical to the default Status values, and a None value for statuses without a status category. If you create custom Status values, you must indicate which category it belongs to. For example, if you create a <i>Waiting for Response</i> value, you may decide that it belongs in the <i>On Hold</i> category. To learn which processes reference StatusCategory, see How are Status Categories Used?
StopStartDate	Type dateTime Properties Filter, Nillable, Sort

Field Name	Details
	Description Indicates when the milestone was paused. The label in the user interface is Stopped Since.
Street	Type textarea Properties Create, Filter, Group, Nillable, Sort, Update Description The street number and name where the work order is completed.
Subject	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The subject of the work order. Try to describe the nature and purpose of the job to be completed. For example, "Annual On-Site Well Maintenance." Maximum length is 255 characters.
Subtotal	Type currency Properties Filter, Nillable, Sort Description Read only. The total of the work order line items' subtotals before discounts and taxes are applied.
SuggestedMaintenanceDate	Type date Properties Create, Filter, Group, Nillable, Sort, Update Description The suggested date that the work order is completed. When the work order is auto-generated from a maintenance plan, this field is automatically populated based on the maintenance plan's settings.
Tax	Type currency Properties Create, Filter, Nillable, Sort, Update

Field Name	Details
	Description The total tax on the work order. You can enter a number with or without the currency symbol and use up to two decimal places. For example, in a work order whose total price is \$100, enter \$10 to apply a 10% tax.
TotalPrice	Type currency Properties Filter, Nillable, Sort Description Read only. The total of the work order line items' prices. This value has discounts applied but not tax.
WorkOrderNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description An eight-digit, auto-generated number that identifies the work order.
WorkTypeId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The work type associated with the work order. When a work type is selected, the work order automatically inherits the work type's <code>Duration</code>, <code>Duration Type</code>, and required skills. If the <code>Duration</code> field for the work type is null, enter the duration value. This is a relationship field. Relationship Name WorkType Relationship Type Lookup Refers To WorkType

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

WorkOrderChangeEvent (API version 48.0)

Change events are available for the object.

WorkOrderFeed

Feed tracking is available for the object.

WorkOrderHistory

History is available for tracked fields of the object.

WorkOrderOwnerSharingRule

Sharing rules are available for the object.

WorkOrderShare

Sharing is available for the object.

WorkOrderLineItem

Represents a subtask on a work order in field service. This object is available in API version 36.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Work orders or Field Service must be enabled.

Fields

Field Name	Details
Address	Type address Properties Filter, Nillable Description The compound form of the address where the line item is completed.
AssetId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The asset associated with the work order line item. The asset is not automatically inherited from the parent work order. This is a relationship field. Relationship Name Asset

Field Name	Details
	Relationship Type Lookup
	Refers To Asset
AssetWarrantyId	Type reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The asset warranty term associated with the work order line item. This field is available in API version 50.0 and above.
City	Type string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The city where the line item is completed. Maximum length is 40 characters.
Country	Type string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The country where the line item is completed. Maximum length is 80 characters.
CurrencyIsoCode	Type picklist
	Properties Defaulted on create, Filter, Group, Restricted picklist, Sort
	Description Available only for orgs with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the organization. The label in the user interface is <code>Currency ISO Code</code> .
Description	Type textarea
	Properties Create, Nillable, Update

Field Name	Details
	Description The description of the work order line item. Try to describe the steps needed to mark the line item Completed.
Discount	Type percent Properties Create, Filter, Nillable, Sort, Update Description The percent discount to apply to the line item. You can enter a number with or without the percent symbol, and you can use up to two decimal places.
Duration	Type double Properties Create, Filter, Nillable, Sort, Update Description The estimated time required to complete the line item. Specify the duration unit in the <code>Duration Type</code> field. If the <code>Duration</code> field on a Work Order is null, it adopts the duration value from the Work Type object when the work type is updated or inserted.  Note: Work order duration and work order line item duration are independent of each other. If you want work order duration to automatically show the sum of the work order line items' duration, replace the <code>Duration</code> field on work orders with a custom roll-up summary field.
DurationInMinutes	Type double Properties Filter, Nillable, Sort Description The estimated duration in minutes. For internal use only.
DurationType	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The unit of the duration: Minutes or Hours.
EndDate	Type dateTime

Field Name	Details
	Properties Create, Filter, Nillable, Sort, Update Description The date on which the line item is completed. This field is blank unless you set up an Apex trigger or quick action to populate it. For example, you can create a quick action that sets the EndDate to 365 days after the StartDate.
GeocodeAccuracy	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The level of accuracy of a location's geographical coordinates compared with its physical address. Usually provided by a geocoding service based on the address's latitude and longitude coordinates.  Note: This field is available in the API only. Possible values are: • Address • Block • City • County • ExtendedZip • NearAddress • Neighborhood • State • Street • Unknown • Zip
IsClosed	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Indicates whether the line item has been closed. Changing the line item's status to Closed causes this checkbox to be selected in the user interface (sets IsClosed to true).  Tip: Use this field to report on closed versus open work order line items.

Field Name	Details
IsGeneratedFromMaintenancePlan	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Identifies whether the work order line item is generated from a maintenance plan.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the line item was last modified. Its label in the user interface is <code>Last Modified Date</code>.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the line item was last viewed.
Latitude	Type double Properties Create, Filter, Nillable, Sort, Update Description Used with <code>Longitude</code> to specify the precise geolocation of the address where the line item is completed. Acceptable values are numbers between <code>-90</code> and <code>90</code> with up to 15 decimal places.  Note: This field is available in the API only.
LineItemNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description An auto-generated number that identifies the work order line item. Each work order's line items start at 1.

Field Name	Details
ListPrice	Type currency Properties Filter, Nillable, Sort Description The price of the line item (product) as listed in its corresponding price book entry. If a price book entry isn't specified, the list price defaults to zero.
LocationId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description A location associated with the work order line item. For example, a work site. This is a relationship field. Relationship Name Location Relationship Type Lookup Refers To Location
Longitude	Type double Properties Create, Filter, Nillable, Sort, Update Description Used with <code>Latitude</code> to specify the precise geolocation of the address where the line item is completed. Acceptable values are numbers between –180 and 180 with up to 15 decimal places.  Note: This field is available in the API only.
MaintenancePlanId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The maintenance plan associated with the work order line item.

Field Name	Details
MaintenanceWorkRuleId	Type reference Properties Filter, Group, Nillable, Sort Description ID of the maintenance work rule that generated this line item. This field is available in API version 50.0 and above.
MinimumCrewSize	Type int Properties Create, Filter, Group, Nillable, Sort, Update Description The minimum crew size allowed for a crew assigned to the line item. If you're not using the Field Service managed package, this field serves as a suggestion rather than a rule. If you are using the managed package, the scheduling optimizer counts the number of service crew members on a service crew to determine whether it fits a work order line item's minimum crew size requirement.
OrderId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The order associated with the line item. For example, you may need to order replacement parts before you can complete the line item. This is a relationship field. Relationship Name Order Relationship Type Lookup Refers To Order
ParentWorkOrderLineItemId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The line item's parent work order line item, if it has one.

Field Name	Details
	 Tip: Create a custom report to view a line item's child line items. This is a relationship field. Relationship Name ParentWorkOrderLineItem Relationship Type Lookup Refers To WorkOrderLineItem
PostalCode	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The postal code where the line item is completed. Maximum length is 20 characters.
PricebookEntryId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The price book entry (product) associated with the line item. The label in the user interface is <code>Product</code>. This field's lookup search only returns products that are included in the work order's price book. This is a relationship field. Relationship Name PricebookEntry Relationship Type Lookup Refers To PricebookEntry
Priority	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description The priority of the line item. The picklist includes the following values, which can be customized: • Low

Field Name	Details
	• Medium • High • Critical
Product2Id	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description (Read only) The product associated with the price book entry. This field is not available in the user interface. For best results, use the <code>PricebookEntryId</code> field in any custom code or layouts. This is a relationship field. Relationship Name Product2 Relationship Type Lookup Refers To Product2
ProductServiceCampaignId	Type reference Properties Filter, Group, Nillable, Sort Description The product service campaign associated with the work order line item.
ProductServiceCampaignItemId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The product service campaign item associated with the work order line item.
Quantity	Type double Properties Create, Filter, Nillable, Sort, Update Description Number of units of the line item included in the associated work order.

Field Name	Details
RecommendedCrewSize	Type int Properties Create, Filter, Group, Nillable, Sort, Update Description The recommended number of people on the service crew assigned to the line item. For example, you might have a Minimum Crew Size of 2 and a Recommended Crew Size of 3.
ReturnOrderId	Type reference Properties Filter, Group, Nillable, Sort Description The return order associated with the work order line item.
ReturnOrderLineItemId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The return order line item associated with the work order line item.
RootWorkOrderLineItemId	Type reference Properties Filter, Group, Nillable, Sort Description (Read only) The top-level line item in a work order line item hierarchy. Depending on where a line item lies in the hierarchy, its root could be the same as its parent.  Note: View a line item's child line items in the Child Work Order Line Items related list. This is a relationship field. Relationship Name RootWorkOrderLineItem Relationship Type Lookup Refers To WorkOrderLineItem

Field Name	Details
ServiceAppointmentCount	Type int Properties Filter, Group, Nillable, Sort Description The number of service appointments on the work order line item.
ServiceDocumentTemplate	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The template ID which sets the template for each service document for the Document Builder feature.
ServiceReportTemplateId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The service report template that the line item uses. If you don't specify a service report template on a work order line item, it uses the service report template listed on its work type. If the work type doesn't list a template or no work type is specified, the line item uses the default service report template.
ServiceTerritoryId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The service territory where the line item is completed. This is a relationship field. Relationship Name ServiceTerritory Relationship Type Lookup Refers To ServiceTerritory
StartDate	Type dateTime

Field Name	Details
	Properties Create, Filter, Nillable, Sort, Update Description The date on which the line item goes into effect. This field is blank unless you set up an Apex trigger or quick action to populate it. For example, you can create a quick action that sets the <code>StartDate</code> to the date when the <code>Status</code> changes to <code>In Progress</code>.
State	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The state where the line item is completed. Maximum length is 80 characters.
Status	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description The status of the line item. The picklist includes the following values, which can be customized: • <code>New</code>—Line item was created, but there hasn't yet been any activity. • <code>In Progress</code>—Work has begun. • <code>On Hold</code>—Work is paused. • <code>Completed</code>—Work is complete. • <code>Cannot Complete</code>—Work could not be completed. • <code>Closed</code>—All work and associated activity is complete. • <code>Canceled</code>—Work is canceled, typically before any work began.
StatusCategory	Type picklist Properties Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort Description The category that each <code>Status</code> value falls into. The <code>Status Category</code> field has eight default values: seven values which are identical to the default <code>Status</code> values, and a <code>None</code> value for statuses without a status category. If you create custom <code>Status</code> values, you must indicate which category it belongs to. For example, if you create a <i>Waiting for Response</i> value,

Field Name	Details
	you may decide that it belongs in the <i>On Hold</i> category. To learn which processes reference StatusCategory, see How are Status Categories Used?
Street	Type textarea Properties Create, Filter, Group, Nillable, Sort, Update Description The street number and name where the line item is completed.
Subject	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description A word or phrase describing the line item.
Subtotal	Type currency Properties Filter, Nillable, Sort Description (Read only) The line item's unit price multiplied by the quantity.
SuggestedMaintenanceDate	Type date Properties Create, Filter, Group, Nillable, Sort, Update Description Date when maintenance work is planned.
TotalPrice	Type currency Properties Filter, Nillable, Sort Description Read only. The line item's subtotal with discounts applied.
UnitPrice	Type currency Properties Create, Filter, Nillable, Sort, Update

Field Name	Details
	Description Initially, the unit price for a work order line item is the line item's list price from the price book, but you can change it.
WorkOrderId	Type reference Properties Create, Filter, Group, Sort Description The line item's parent work order. Because work order line items must be associated with a work order, this is a required field. This is a relationship field. Relationship Name WorkOrder Relationship Type Lookup Refers To WorkOrder
WorkTypeId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The work type associated with the line item. When a work type is selected, the line item automatically inherits the work type's <code>Duration</code>, <code>Duration Type</code>, and required skills. If the <code>Duration</code> field for the work type is null, enter the duration value. This is a relationship field. Relationship Name WorkType Relationship Type Lookup Refers To WorkType

Usage

A work order line item is a child record of a work order. It represents a specific subtask on a work order.

For example, suppose a customer purchased a truck from you. The truck is represented as an asset in your Salesforce org. After some time, the truck needs both headlight bulbs replaced. Here's one way that you can use work orders and work order line items to track the repair.

1. Create a work order named "Replace Headlight Bulbs" from the asset record detail page.
2. Add three work order line items to the work order: "Replace Left Headlight Bulb," "Replace Right Headlight Bulb," and "Test Headlights."
3. Assign the work order to a technician via a queue.
4. As the technician completes each line item, he or she marks the item `Completed`.
5. When all the line items are complete, the technician marks the work order `Completed`.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

WorkOrderLineItemChangeEvent (API version 48.0)

Change events are available for the object.

WorkOrderLineItemFeed

Feed tracking is available for the object.

WorkOrderLineItemHistory

History is available for tracked fields of the object.

WorkOrderLineItemStatus

Represents a possible status of a work order line item in field service.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

Field Service must be enabled.

Fields

Field Name	Details
<code>ApiName</code>	Type string Properties Filter, Group, idLookup, Sort Description The API name of the status value.
<code>IsDefault</code>	Type boolean

Field Name	Details
	Properties Defaulted on create, Filter, Group, Sort Description Indicates that the status value is the default status on work orders. Only one status value can be the default.
MasterLabel	Type string Properties Filter, Group, Nillable, Sort Description The label for the picklist value that appears in the UI.
SortOrder	Type int Properties Filter, Group, Nillable, Sort Description The value's position in the drop-down list of values in the UI.
StatusCode	Type picklist Properties Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort Description The status category that the value corresponds to. The Status Category field has seven values which are identical to the default Status values.

Usage

The Status field on work order line items comes with the following values:

- New—Line item was created, but there hasn't yet been any activity.
- In Progress—Work has begun.
- On Hold—Work is paused.
- Completed—Work is complete.
- Cannot Complete—Work could not be completed.
- Closed—All work and associated activity is complete.
- Canceled—Work is canceled, typically before any work began.

The WorkOrderLineItemStatus object corresponds to the Status field. Adding a value to the Status field—for example, Canceled By Customer—creates a work order line item status record, and vice versa.

 **Note:** Work order line items also come with a `StatusCategory` field whose values are identical to the default `Status` values. If you create custom `Status` values, you must indicate which category it belongs to. For example, if you create a *Customer Absent* value, you may decide that it belongs in the *Cannot Complete* category. To learn which processes reference `StatusCategory`, see [How are Status Categories Used?](#)

WorkPlan

Represents a work plan for a work order or work order line item. This object is available in API version 52.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field	Details
Description	Type textarea Properties Create, Nillable, Update Description The description of the work plan.
ExecutionOrder	Type int Properties Create, Filter, Group, Nillable, Sort, Update Description The order in which the work plan is executed. Only positive values or null are supported.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last interacted with this record, directly or indirectly.
LastViewedDate	Type dateTime

Field	Details
	Properties Filter, Nillable, Sort Description The timestamp when the current user last viewed this record or list view. If this value is null, it's possible that the user only accessed this record or list view (<code>LastReferencedDate</code>), but not viewed it.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The name of the work plan.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The ID of the user who created the work plan. Relationship Name Owner Relationship Type Lookup Refers To Group, User
ParentRecordId	Type reference Properties Create, Filter, Group, Sort Description The ID of the work order, work order line item, or change request that the work plan is associated with. Available in API version 54.0 and later. This field is a polymorphic relationship field. Relationship Name ParentRecord Relationship Type Lookup Refers To ChangeRequest, WorkOrder, WorkOrderLineItem

Field	Details
ParentRecordType	Type string Properties Filter, Group, Nillable, Sort Description Describes whether the parent record is a work order, work order line item, or change request. Available in API version 54.0 and later.
WorkOrderId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description Required. The ID of the work order. Relationship Name WorkOrder Relationship Type Lookup Refers To WorkOrder
WorkOrderLineItemId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the work order line item. Relationship Name WorkOrderLineItem Relationship Type Lookup Refers To WorkOrderLineItem
WorkPlanTemplateId	Type reference Properties Filter, Group, Nillable, Sort Description The ID of the work plan template record. Available in API version 54.0 and later. This field is a relationship field.

Field	Details
	Relationship Name WorkPlanTemplate
	Relationship Type Lookup
	Refers To WorkPlanTemplate

Associated Objects

This object has these associated objects. Unless noted, they're available in the same API version as this object.

WorkPlanChangeEvent

Change events are available for the object. Available in API version 54.0 and later.

WorkPlanFeed

Feed tracking is available for the object.

WorkPlanHistory

History is available for tracked fields of the object.

WorkPlanOwnerSharingRule

Sharing rules are available for the object.

WorkPlanShare

Sharing is available for the object.

WorkPlanSelectionRule

Represents a rule that selects a work plan for a work order or work order line item. This object is available in API version 52.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field	Details
AssetId	Type reference
	Properties Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The ID of the asset.
Description	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The description of the selection rule.
IsActive	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Controls whether this selection rule is active (<code>true</code>) or not (<code>false</code>). Default is <code>false</code> . Label is Active.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last interacted with this record, directly or indirectly.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last viewed this record or list view. If this value is null, it's possible that the user only accessed this record or list view (<code>LastReferencedDate</code>), but not viewed it.
LocationId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the location.

Field	Details
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The ID of the owner.
Product2Id	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the product. Label is Product.
ServiceTerritoryId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the service territory.
WorkPlanSelectionRuleNumber	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description The auto-generated number of the work plan selection rule, for example, WPSR-0001.
WorkPlanTemplateId	Type reference Properties Create, Filter, Group, Sort, Update Description Required. The ID of the work plan template.
WorkTypeId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The ID of the work type.

Associated Objects

This object has these associated objects. Unless noted, they're available in the same API version as this object.

WorkPlanSelectionRuleChangeEvent

Change events are available for the object.

WorkPlanSelectionRuleFeed

Feed tracking is available for the object.

WorkPlanSelectionRuleHistory

History is available for tracked fields of the object.

WorkPlanSelectionRuleOwnerSharingRule

Sharing rules are available for the object.

WorkPlanSelectionRuleShare

Sharing is available for the object.

WorkPlanTemplate

Represents a template for a work plan. This object is available in API version 52.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field	Details
Description	Type textarea Properties Create, Nillable, Update Description The description of the work plan template.
IsActive	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Controls whether the specific template is available for application (<code>true</code>) or not (<code>false</code>). Default is <code>false</code> . Label is Active.

Field	Details
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last interacted with this record, directly or indirectly.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last viewed this record or list view. If this value is null, it's possible that the user only accessed this record or list view (LastReferencedDate), but not viewed it.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The user-defined name of the work plan template.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The ID of the owner who created the work plan template.
RelativeExecutionOrder	Type int Properties Create, Filter, Group, Nillable, Sort, Update Description The relative execution order for sorting the work plan when it's applied to the work order or work order line item. Only positive integers are supported.

Associated Objects

This object has these associated objects. Unless noted, they're available in the same API version as this object.

WorkPlanTemplateChangeEvent

Change events are available for the object.

WorkPlanTemplateFeed

Feed tracking is available for the object.

WorkPlanTemplateHistory

History is available for tracked fields of the object.

WorkPlanTemplateOwnerSharingRule

Sharing rules are available for the object.

WorkPlanTemplateShare

Sharing is available for the object.

WorkPlanTemplateEntry

Represents an object that associates a work step template with a work plan template. This object is available in API version 52.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field	Details
ExecutionOrder	Type int Properties Create, Filter, Group, Nillable, Sort, Update Description The sequence number of when this entry is executed. Only positive values are supported.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last interacted with this record, directly or indirectly.
LastViewedDate	Type dateTime

Field	Details
	Properties Filter, Nillable, Sort Description The timestamp when the current user last viewed this record or list view. If this value is null, it's possible that the user only accessed this record or list view (<code>LastReferencedDate</code>), but not viewed it.
<code>WorkPlanTemplateEntryNumber</code>	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description The auto-generated number of the work plan template entry, for example, WPTE-0001.
<code>WorkPlanTemplateId</code>	Type reference Properties Create, Filter, Group, Sort Description Required. The ID of the work plan template.
<code>WorkStepTemplateId</code>	Type reference Properties Create, Filter, Group, Sort, Update Description Required. The ID of the work step template.

Associated Objects

This object has these associated objects. Unless noted, they're available in the same API version as this object.

WorkPlanTemplateEntryChangeEvent

Change events are available for the object.

WorkPlanTemplateEntryFeed

Feed tracking is available for the object.

WorkPlanTemplateEntryHistory

History is available for tracked fields of the object.

WorkOrderStatus

Represents a possible status of a work order in field service.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

Field Service must be enabled.

Fields

Field Name	Details
<code>ApiName</code>	Type string Properties Filter, Group, idLookup, Sort Description The API name of the status value.
<code>IsDefault</code>	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Indicates that the status value is the default status on work orders. Only one status value can be the default.
<code>MasterLabel</code>	Type string Properties Filter, Group, Nillable, Sort Description The label for the picklist value that appears in the UI.
<code>SortOrder</code>	Type int Properties Filter, Group, Nillable, Sort Description The value's position in the drop-down list of values in the UI.
<code>StatusCode</code>	Type picklist Properties Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort

Field Name	Details
	Description The status category that the value corresponds to. The Status Category field has seven values which are identical to the default Status values.

Usage

The Status field on work orders comes with the following values:

- New—Work order was created, but there hasn't yet been any activity.
- In Progress—Work has begun.
- On Hold—Work is paused.
- Completed—Work is complete.
- Cannot Complete—Work could not be completed.
- Closed—All work and associated activity is complete.
- Canceled—Work is canceled, typically before any work began.

The WorkOrderStatus object corresponds to the Status field. Adding a value to the Status field—for example, Canceled By Customer—creates a work order status record, and vice versa.

 **Note:** Work orders also come with a StatusCategory field whose values are identical to the default Status values. If you create custom Status values, you must indicate which category it belongs to. For example, if you create a *Customer Absent* value, you may decide that it belongs in the *Cannot Complete* category. To learn which processes reference StatusCategory, see [How are Status Categories Used?](#)

WorkStep

Represents a work step in a work plan. This object is available in API version 52.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field	Details
ActionDefinition	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details
	Description The platform action that the work step executes. The possible values are the names of the flow and quick actions configured in your org. To launch Lightning Web Components from Work Steps, you must use <code>QuickAction</code> on the action definition.
ActionType	Type picklist Properties Filter, Group, Nillable, Restricted picklist, Sort Description The type of platform action that the work step is associated with. Possible values are: - Flow - QuickAction
Description	Type textarea Properties Create, Filter, Nillable, Sort, Update Description The description of the work step.
EndTime	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description The date and time the work step ends. The value must be greater than or equal to <code>StartTime</code>.
ExecutionOrder	Type int Properties Create, Filter, Group, Nillable, Sort, Update Description The order in which the work step is executed. Only positive integer values or null are supported.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort

Field	Details
	Description The timestamp when the current user last interacted with this record, directly or indirectly.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last viewed this record or list view. If this value is null, it's possible that the user only accessed this record or list view (<code>LastReferencedDate</code>), but not viewed it.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description Required. The user-defined name of the work step.
PausedFlowInterviewId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The auto-populated ID of the flow interview paused by a user.
ProcessType	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The flow process type launched from the work step. Possible values are: • <code>DataCaptureFlow</code>—Data Capture Flow • <code>DiscoveryFrameworkFlow</code>—Discovery Framework Data Capture Flow (Beta) • <code>FieldServiceMobileFlow</code>—Field Service Mobile Flow The default value is <code>DataCaptureFlow</code> .
StartTime	Type dateTime

Field	Details
	Properties Create, Filter, Nillable, Sort, Update Description The date and time the work step starts.
Status	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Sort, Update Description The customizable status of the work order. Every status must be mapped to a status category, but there can be status categories not mapped to a status. Possible values are: - Completed - In Progress - New - Not Applicable - Paused
StatusCategory	Type picklist Properties Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort Description The category that each status value belongs to. Each default status category is mapped to the corresponding default status. If you create a custom status, you must indicate which status category it belongs to. To learn which processes reference StatusCategory, see How are Status Categories Used?. Possible values are: - Completed - InProgress - New - NotApplicable - Paused
WorkOrderId	Type reference Properties Filter, Group, Sort

Field	Details
	Description The ID of the work order.
WorkOrderLineItemId	Type reference Properties Filter, Group, Nillable, Sort Description The ID of the work order line item.
WorkPlanExecutionOrder	Type int Properties Filter, Group, Nillable, Sort Description The ID of the plan execution order.
WorkPlanId	Type reference Properties Create, Filter, Group, Sort Description The ID of the work plan.

Associated Objects

This object has these associated objects. Unless noted, they're available in the same API version as this object.

WorkStepChangeEvent

Change events are available for the object.

WorkStepFeed

Feed tracking is available for the object.

WorkStepHistory

History is available for tracked fields of the object.

WorkStepStatus

Represents a picklist for a status category on a work step. This object is available in API version 52.0 and later.

Supported Calls

`describeSObjects()`, `query()`, `retrieve()`

Special Access Rules

Field Service must be enabled.

Fields

Field	Details
ApiName	Type string Properties Filter, Group, idLookup, Sort Description Required. The name of the work step status.
IsDefault	Type boolean Properties Defaulted on create, Filter, Group, Sort Description Controls whether this status is the default value of the picklist of the corresponding status category (<code>true</code>) or not (<code>false</code>). Default is <code>false</code>.
MasterLabel	Type string Properties Filter, Group, Nillable, Sort Description Required. The label of the work step status.
SortOrder	Type int Properties Filter, Group, Nillable, Sort Description Required. The order in which the work step statuses are displayed in the status category's picklist.
StatusCode	Type picklist Properties Required. Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort Description The status category that this status belongs to.

Field	Details
	Possible values are: - Completed - InProgress - New - NotApplicable - Paused

WorkStepTemplate

Represents a template for a work step. This object is available in API version 52.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field	Details
ActionDefinition	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The platform action that the work step executes. The possible values are the names of the flow and quick actions configured in your org.
ActionType	Type picklist Properties Filter, Group, Nillable, Restricted picklist, Sort Description The type of platform action that the work step is associated with. Possible values are: - Flow - QuickAction

Field	Details
Description	Type textarea Properties Create, Filter, Nillable, Sort, Update Description The description of the work step template.
IsActive	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Controls whether this work step template is active <code>true</code> or not <code>false</code>. Default is <code>false</code>.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last interacted with this record, directly or indirectly.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp when the current user last viewed this record or list view. If this value is null, it's possible that the user only accessed this record or list view (<code>LastReferencedDate</code>), but not viewed it.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The user-defined name of the work step template.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update

Field	Details
	Description The ID of the owner who created the work step template.

Associated Objects

This object has these associated objects. Unless noted, they're available in the same API version as this object.

WorkStepTemplateChangeEvent

Change events are available for the object.

WorkStepTemplateFeed

Feed tracking is available for the object.

WorkStepTemplateHistory

History is available for tracked fields of the object.

WorkStepTemplateOwnerSharingRule

Sharing rules are available for the object.

WorkStepTemplateShare

Sharing is available for the object.

WorkType

Represents a type of work to be performed in Field Service and Lightning Scheduler. Work types are templates that can be applied to work order or work order line items. This object is available in API version 38.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field Name	Details
Description	Type textarea Properties Create, Nillable, Update Description The description of the work type. Try to add details about the task or tasks that this work type represents.

Field Name	Details
DurationType	Type picklist Properties Create, Filter, Group, Defaulted on create, Restricted picklist, Sort, Update Description The unit of the <code>Estimated Duration</code>: Minutes or Hours.
EstimatedDuration	Type double Properties Create, Filter, Sort, Update Description The estimated length of the work. The estimated duration is in minutes or hours based on the value selected in the <code>Duration Type</code> field.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the work type was last modified. Its label in the user interface is <code>Last Modified Date</code>.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The date when the work type was last viewed by the current user.
MinimumCrewSize	Type int Properties Create, Filter, Group, Nillable, Sort, Update Description The minimum crew size allowed for a crew assigned to the work. Work orders and work order line items inherit their work type's minimum crew size. If you're not using the Field Service managed package, this field serves as a suggestion rather than a rule. If you are using the managed package, the scheduling optimizer counts the number of service crew members on a service crew to determine whether it fits the minimum crew size requirement.

Field Name	Details
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The name of the work type. Try to use a name that helps users quickly understand the type of work orders that can be created from the work type. For example, "Annual Refrigerator Maintenance" or "Valve Replacement."
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The work type's owner. This is a polymorphic relationship field. Relationship Name Owner Relationship Type Lookup Refers To Group, User
RecommendedCrewSize	Type int Properties Create, Filter, Group, Nillable, Sort, Update Description The recommended number of people on the service crew assigned to the work. For example, you might have a Minimum Crew Size of 2 and a Recommended Crew Size of 3. Work orders and work order line items inherit their work type's recommended crew size.
SaDocumentTemplate	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The document template ID. If <code>ServiceDocumentTemplateId</code> isn't specified, this document template ID determines which service document template is used for service documents generated from a service appointment. The ID is 15 to 18 characters long.

Field Name	Details
ServiceReportTemplateId	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The service report template associated with the work type. When users create service reports from a work order or work order line item that uses this work type, the reports use this template.
ShouldAutoCreateSvcAppt	Type boolean Properties Create, Filter, Group, Defaulted on create, Sort, Update Description Select this option to have a service appointment automatically created on work orders and work order line items that use the work type.  Note: - By default, the Due Date on auto-created service appointments is seven days after the created date. Admins can adjust this offset from the Field Service Settings page in Setup. - If a work type with the Auto-Create Service Appointment option selected is added to an existing work order or work order line item, a service appointment is only created for the work order or work order line item if it doesn't yet have one. - If someone updates an existing work type by selecting the Auto-Create Service Appointment option, service appointments aren't created on work orders and work order line items that were already using the work type.
WoDocumentTemplate	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The document template ID. If ServiceDocumentTemplateId isn't specified, this document template ID determines which service document template is used for service documents generated from a work order. The ID is 15 to 18 characters long.
WoliDocumentTemplate	Type string

Field Name	Details
	Properties Create, Filter, Group, Nillable, Sort, Update Description The document template ID. If <code>ServiceDocumentTemplateId</code> isn't specified, this document template ID determines which service document template is used for service documents generated from a work order line item. The ID is 15 to 18 characters long.

Usage

Adding a work type to a work order or work order line item causes the record to inherit the work type's duration values and required skills and products.



Note:

- If needed, you can update the duration values and required skills and products on a work order or work order line item after they're inherited from the work type.
- If a work order or work order line item already has required skills or products, associating it with a work type doesn't cause it to inherit the work type's requirements.
- If a work order or work order line item already has a duration value in its `Duration` field, associating it with a work type doesn't cause it to inherit the work type's duration value.
- Customizations to required skills or products, such as validation rules or Apex triggers, are not carried over from work types to work orders and work order line items.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

WorkTypeChangeEvent (API version 48.0)

Change events are available for the object.

WorkTypeFeed

Feed tracking is available for the object.

WorkTypeHistory

History is available for tracked fields of the object.

WorkTypeOwnerSharingRule

Sharing rules are available for the object.

WorkTypeShare

Sharing is available for the object.

WorkTypeGroup

Represents a grouping of work types used to categorize types of appointments available in Lightning Scheduler, or to define scheduling limits in Field Service. This object is available in API version 45.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
AdditionalInformation	Type multipicklist Properties Create, Filter, Nillable, Update Description Additional information about the types of appointments this work type group represents.
Description	Type textarea Properties Create, Nillable, Update Description A description of this work type group.
GroupType	Type picklist Properties Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update Description The category of this work type group. Possible values are: • Capacity—A group of work types used to define a work capacity limit in Field Service. • Default—A non-capacity group of work types used in Lightning Scheduler.
IsActive	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether this work type group can be used for appointment scheduling or work capacity limits. A work type can belong to only one active work type group of type Capacity.
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort

Field	Details
	Description The date and time that the current user last viewed a record related to this object.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp for when the current user last viewed this object.
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The name of this work type group.
OwnerId	Type reference Properties Create, Defaulted on create, Filter, Group, Sort, Update Description The ID of the user who created this record. This is a polymorphic relationship field. Relationship Name Owner Relationship Type Lookup Refers To Group, User

Associated Objects

This object has the following associated objects. Unless noted, they are available in the same API version as this object.

WorkTypeGroupFeed

Feed tracking is available for the object.

WorkTypeGroupHistory

History is available for tracked fields of the object.

WorkTypeGroupOwnerSharingRule

Sharing rules are available for the object.

WorkTypeGroupShare

Sharing is available for the object.

WorkTypeGroupMember

Represents the relationship between a work type and the work type group it belongs to. This object is available in API version 45.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
LastReferencedDate	Type dateTime Properties Filter, Nillable, Sort Description The date and time that the current user last viewed a record related to this object.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort Description The timestamp for when the current user last viewed this object.
Name	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description Autogenerated number identifying the work type group membership. It uses the format #####.
WorkTypeGroupId	Type reference Properties Create, Filter, Group, Sort Description The ID of the work type group that this record belongs to.

Field	Details
	This is a relationship field. Relationship Name WorkTypeGroup Relationship Type Lookup Refers To WorkTypeGroup
WorkTypeId	Type reference Properties Create, Filter, Group, Sort Description The ID of the work type that this record corresponds to. This is a relationship field. Relationship Name WorkType Relationship Type Lookup Refers To WorkType

Associated Objects

This object has the following associated objects. Unless noted, they are available in the same API version as this object.

WorkTypeGroupMemberFeed

Feed tracking is available for the object.

WorkTypeGroupMemberHistory

History is available for tracked fields of the object.

Field Service Custom Fields on Standard Objects

A list of custom fields on standard Salesforce objects installed with the Field Service managed package.

See the [Field Service Apex Namespace](#) section for more API references related to the managed package.

IN THIS SECTION:

[AssignedResource Custom Fields](#)

Custom fields associated with a service resource who is assigned to a service appointment in Field Service. Assigned resources appear in the Assigned Resources related list on service appointments. This object is available in API version 38.0 and later.

[ResourceAbsence Custom Fields](#)

Custom fields associated with a time period in which a service resource is unavailable to work in Field Service.

[ServiceAppointment Custom Fields](#)

Custom fields associated with an appointment to complete work for a customer in Field Service.

[ServiceResource Custom Fields](#)

Custom fields associated with a field service technician or crew in Field Service.

[ServiceResourceCapacity Custom Fields](#)

Custom fields associated with the maximum number of scheduled hours or number of service appointments that a capacity-based service resource can complete within a specific time period.

[ServiceTerritory Custom Fields](#)

Custom fields associated with a geographic or functional region in which field service work can be performed in Field Service.

[TimeSlot Custom Fields](#)

Custom fields associated with a period of time on a specified day of the week during which field service work can be performed in Field Service. Operating hours consist of one or more time slots.

[WorkOrder Custom Fields](#)

Custom fields associated with field service work to be performed for a customer.

[WorkOrderLineItem Custom Fields](#)

Custom fields associated with a subtask on a work order in field service.

AssignedResource Custom Fields

Custom fields associated with a service resource who is assigned to a service appointment in Field Service. Assigned resources appear in the Assigned Resources related list on service appointments. This object is available in API version 38.0 and later.

The standard fields are documented in the [AssignedResource](#) object reference.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service managed package must be installed.

Fields

Field	Details
<code>FSL__EstimatedTravelDistanceFrom__c</code>	Type double Properties Create, Filter, Nillable, Sort, Update Description The estimated travel distance from the service resource's assigned appointment location to their home base. For this field, the assigned appointment location must have latitude and longitude coordinates and be the last location of the day. If it doesn't have coordinates, isn't the last location of the day, or the travel calculations are disabled, this field is 0.

Field	Details
FSL__EstimatedTravelDistanceTo__c	Type double Properties Create, Filter, Nillable, Sort, Update Description The estimated travel distance to the service resource's assigned appointment location from a service appointment, another resource absence location, or their home base. For this field, the assigned appointment location must have latitude and longitude coordinate. If it doesn't have latitude and longitude coordinates or the travel calculations are disabled, this field is 0.
FSL__EstimatedTravelTimeFrom__c	Type double Properties Create, Filter, Nillable, Sort, Update Description The estimated travel time from the service resource's assigned appointment location to their home base. For this field, the assigned appointment location must have latitude and longitude coordinates and be the last location of the day. If it doesn't have coordinates, isn't the last location of the day, or the travel calculations are disabled, this field is 0.
FSL__Estimated_Travel_Time_From_Source__c	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The method used to calculate the travel time from the service resource's assigned appointment location to their home base. For this field, the assigned appointment location must have latitude and longitude coordinates and be the last location of the day. If it doesn't have coordinates, isn't the last location of the day, or the travel calculations are disabled, this field is None. Possible values are: • Aerial • None • Predictive • SLR
FSL__Estimated_Travel_Time_To_Source__c	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details
	Description The method used to calculate the travel time to the service resource's assigned appointment location from a service appointment, another resource absence location, or their home base. For this field, the assigned appointment location must have latitude and longitude coordinates. If it doesn't have latitude and longitude coordinates or the travel calculations are disabled, this field is <code>None</code>. Possible values are: • <code>Aerial</code> • <code>None</code> • <code>Predictive</code> • <code>SLR</code>

Internal Fields

These internal fields are used by the Field Service managed package. Although they're publicly accessible, they must only be updated by the managed package.

- `FSL__Last_Updated_Epoch__c`—Used to prevent the overlapping of multiple concurrent scheduling requests.
- `FSL__UpdatedByOptimization__c`—Equals `true` if the record was updated by the optimization engine.
- `FSL__calculated_duration__c`—Indicates the duration (start to end time) of the service appointment assigned to the resource in minutes.

SEE ALSO:

[AssignedResource](#)

ResourceAbsence Custom Fields

Custom fields associated with a time period in which a service resource is unavailable to work in Field Service.

The standard fields are documented in the [ResourceAbsence](#) object reference.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `update()`, `upsert()`

Special Access Rules

Field Service managed package must be installed.

Fields

Field	Details
FSL__Approved__c	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description When this field is <code>true</code> and the Activate approval confirmation on resource absences setting is enabled, the resource absence appears in the dispatcher console's Gantt chart and is considered by the scheduler. When this field is <code>false</code> and the setting is enabled, the resource absence doesn't appear in the Gantt chart and is ignored by the scheduler. The default value is <code>false</code>. For more info, view the View Resource Absences on the Gantt and Map Help article.
FSL__Duration_In_Minutes__c	Type double Properties Filter, Nillable, Sort Description The duration of the resource absence in minutes. This field is automatically populated after the resource absence is created. This is a calculated field. Formula <pre>IF (ISBLANK(Start), 0, (End - Start)*24*60)</pre>
FSL__EstTravelTimeFrom__c	Type double Properties Create, Filter, Nillable, Sort, Update Description The estimated travel time from the service resource's absence location to their home base. For this field, the absence location must have latitude and longitude coordinates and be the last location of the day. If it doesn't have coordinates, isn't the last location of the day, or the travel calculations are disabled, this field is 0.
FSL__EstTravelTime__c	Type double Properties Create, Filter, Nillable, Sort, Update

Field	Details
	Description The estimated time to the service resource's absence location from a service appointment or another resource absence location. For this field, the absence location must have latitude and longitude coordinates. If it doesn't have latitude and longitude coordinates or the travel calculations are disabled, this field is 0.
FSL__EstimatedTravelDistanceFrom__c	Type double Properties Create, Filter, Nillable, Sort, Update Description The estimated travel distance from the service resource's absence location to their home base. For this field, the absence location must have latitude and longitude coordinates and be the last location of the day. If it doesn't have coordinates, isn't the last location of the day, or the travel calculations are disabled, this field is 0.
FSL__EstimatedTravelDistanceTo__c	Type double Properties Create, Filter, Nillable, Sort, Update Description The estimated travel distance to service resource's absence location from a service appointment or another resource absence location. For this field, the absence location must have latitude and longitude coordinate. If it doesn't have latitude and longitude coordinates or the travel calculations are disabled, this field is 0.
FSL__Estimated_Travel_Time_From_Source__c	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The method used to calculate the travel time from the service resource's absence location to a service appointment or another resource absence location. For this field, the absence location must have latitude and longitude coordinates and be the last location of the day. If it doesn't have coordinates, isn't the last location of the day, or the travel calculations are disabled, this field is <code>None</code>. Possible values are: • Aerial • None • Predictive • SLR

Field	Details
FSL__Estimated_Travel_Time_To_Source__c	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The method used to calculate the travel time to this service resource's absence location from a service appointment or another resource absence location. For this field, the absence location must have latitude and longitude coordinates. If it doesn't have latitude and longitude coordinates or the travel calculations are disabled, this field is None. Possible values are: • Aerial • None • Predictive • SLR
FSL__GanttLabel__c	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The label of the resource absence in the Field Service dispatcher console's Gantt chart and resource calendar. This replaces the resource absence's number in the chart.
FSL__Gantt_Color__c	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The Hex color of the resource absence in Field Service dispatcher console's Gantt chart and resource calendar.
FSL__Scheduling_Policy_Used__c	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description A scheduling policy used by the scheduler for the service appointment. This policy overrides the default one in the Field Service Settings page or the one that the scheduler would otherwise select. This is used for travel calculations. This is a relationship field.

Field	Details
	Relationship Name FSL__Scheduling_Policy_Used__r
	Relationship Type Lookup
	Refers To FSL__Scheduling_Policy__c

Internal Fields

These internal fields are used by the Field Service managed package for Street Level Routing calculations. Although they're publicly accessible, they must only be updated by the managed package.

- FSL__InternalSLRGeolocation__Latitude__s
- FSL__InternalSLRGeolocation__Longitude__s
- FSL__InternalSLRGeolocation__c

SEE ALSO:

[Salesforce Object Reference: ResourceAbsence](#)

[Salesforce Help: View Resource Absences on the Gantt and Map](#)

ServiceAppointment Custom Fields

Custom fields associated with an appointment to complete work for a customer in Field Service.

The standard fields are documented in the [ServiceAppointment](#) object reference.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service managed package must be installed.

Fields

Field	Details
FSL__Appointment_Grade__c	Type double
	Properties Create, Filter, Nillable, Sort, Update

Field	Details
	Description The appointment grade of the scheduled appointment using the Appointment Booking feature.
FSL__Auto_Schedule__c	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates if the appointment is created and scheduled in the same action. The default value is <code>false</code>. For more info, view the Schedule an Appointment Automatically Help article.
FSL__Duration_In_Minutes__c	Type double Properties Filter, Nillable, Sort Description The duration in minutes of the scheduled appointment. It calculates the time between the scheduled start and end times in minutes. This is a calculated field. Formula <pre>IF (ISBLANK(SchedStartTime), 0, (SchedEndTime - SchedStartTime)*24*60)</pre>
FSL__Emergency__c	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates if the Emergency Wizard global action schedules the appointment. If true, the service appointment has an emergency icon in the Field Service dispatcher console's Gantt chart. The default value is <code>false</code>.
FSL__GanttColor__c	Type string Properties Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The Hex color of the service appointment in the Field Service dispatcher console's Gantt chart and the resource calendar.
FSL__GanttIcon__c	Type textarea Properties Create, Nillable, Update Description A custom icon for the service appointment that appears in the Gantt chart, map, and appointment list. This helps dispatchers quickly identify appointment characteristics. For example, use a custom icon to indicate that an appointment is for a VIP or first-time customer. The format is a URL ending in an image suffix, such as .png or .gif. The image is scaled to 16 x 16 pixels. For more info, view the Create Custom Appointment Icons Help article.
FSL__GanttLabel__c	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The label of the scheduled service appointment in the Field Service dispatcher console's Gantt chart. This replaces the service appointment number in the chart.
FSL__Gantt_Display_Date__c	Type dateTime Properties Create, Filter, Nillable, Sort, Update Description The Gantt Display Date filter in the date field dropdown menu in the Field Service dispatcher console to control which appointments are visible in the appointment list. When a service appointment's Gantt Display Date falls within the Gantt time frame, the appointment is visible on the Gantt. For example, if a maintenance appointment must be completed within the next six months, you can set the date so that you see it on the Gantt every day. You can set up this field to update an important appointment's Gantt Display Date to today's date on a daily basis. For more info, view the Control Which Appointments Appear in the Dispatcher Console Help article.
FSL__InJeopardyReason__c	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details
	Description The reason for when the service appointment is in jeopardy. Use this field only when the <code>FSL__InJeopardy__c</code> status is <code>true</code>. You can add custom picklist values. Possible values are: • Delayed Finish • Delayed Start • Due Date Approaching • No Response • Rejected by Contractor
<code>FSL__InJeopardy__c</code>	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Determines if a service appointment is in jeopardy. This helps dispatchers gain visibility to service appointments at risk. A user can manually set the service appointment status to In Jeopardy or this can be done automatically using, for example, process builders or triggers. The default value is <code>false</code>.
<code>FSL__IsFillInCandidate__c</code>	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Determines if this service appointment is considered as a candidate when filling in a schedule with the Fill-In Schedule feature. If a service appointment's parent record is a work order or work order line item, the parent record's <code>FSL__IsFillInCandidate__c</code> field must also be set to <code>true</code> for the appointment to be a candidate. Alternatively, instead of using this field, you can create a custom checkbox field, including formula fields, to evaluate whether this appointment is considered as a candidate. This can be done through the Field Service Settings page. The default value is <code>true</code>. For more info, view the Fill Schedule Gaps Help article.
<code>FSL__IsMultiDay__c</code>	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update

Field	Details
	Description Determines if this service appointment spans over multiple days. Alternatively, instead of using this field, you can create a checkbox formula field through the Field Service Settings page to evaluate whether it spans over multiple days or not. The default value is <code>false</code>. For more info, view the Enable Multiday Service Appointments Help article.
FSL__Pinned__c	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Determines if this service appointment is pinned to the Field Service dispatcher console's Gantt chart. Pinned service appointments can't be manually dragged or automatically scheduled by any scheduling operation. Pinned service appointments have a lock icon in the Field Service dispatcher console's Gantt chart. The default value is <code>false</code>.
FSL__Prevent_Geocoding_For_Chatter_Actions__c	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Determines if the service appointment's BeforeUpdate Platform Apex trigger disables the Chatter Actions's geolocation cleanup on address change. When this field is set to <code>true</code>, it prevents Chatter Actions to geocode the address and waits until Field Service does it after the address changes. This field is set to <code>false</code> after the cleanup completes. The default value is <code>false</code>.
FSL__Schedule_Mode__c	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Unrestricted picklist, Sort, Update Description The type of the scheduling operation. For example, when not using Enhanced Scheduling and Optimization, if the service appointment is scheduled using drag and drop, the value is <code>Manual</code>. If the service appointment is scheduled using the Appointment Booking feature, the value is <code>Automatic</code>. When using Enhanced Scheduling and Optimization, if the service appointment is scheduled using drag and drop, the value is <code>Drag and Drop</code>. If the service appointment is scheduled using the Appointment Booking feature, the value is <code>Schedule</code>. This field is populated by the system. Don't edit this field. Possible values are:

Field	Details
	• Automatic • Manual • None • Optimization Additional values for each scheduling operation, available only with Enhanced Scheduling and Optimization: • Drag and Drop • Schedule • Global Optimization • In-Day Optimization • Resource Optimization The default value is 'None'.
FSL__Schedule_over_lower_priority_appointment__c	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Determines whether you can schedule critical service appointments over lower priority appointments. The default value is <code>false</code>. For more info, view the Schedule Appointments Using Priorities Help article.
FSL__Scheduling_Policy_Used__c	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description A scheduling policy used by the scheduler for the service appointment. If you edit this field, the policy overrides the default one on the Field Service Settings page. If this field is empty, the field populates with the policy used by the scheduler after the service appointment gets scheduled. This is a relationship field. Relationship Name FSL__Scheduling_Policy_Used__r Relationship Type Lookup Refers To FSL__Scheduling_Policy__c

Field	Details
FSL__UpdatedBy Optimization__c	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Determines if the service appointment is updated by the optimizer. This field is populated by the system. Don't edit this field. The default value is <code>false</code>.
FSL__Use_Async_Logic__c	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Determines if the scheduling features related to the service appointment run asynchronously. If you use UI features, such as the Appointment Booking global action, the managed package takes care of this async response for you. The default value is <code>false</code>.
FSL__Virtual_Service_ For_Chatter_Action__c	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Determines if the service appointment is a candidate or dummy appointment. If the value is <code>true</code>, the appointment is ignored by your custom triggers. This field is populated by the system. Don't edit this field. The default value is <code>false</code>.

Internal Fields

These internal fields are used by the Field Service managed package for Street Level Routing calculations. Although they're publicly accessible, they must only be updated by the managed package.

- FSL__InternalSLRGeolocation__Latitude__s
- FSL__InternalSLRGeolocation__Longitude__s

- `FSL__InternalSLRGeolocation__c`

SEE ALSO:

- [Salesforce Object Reference: ServiceAppointment](#)
- [Salesforce Help: Schedule an Appointment Automatically](#)
- [Salesforce Help: Create Custom Appointment Icons](#)
- [Salesforce Help: Control Which Appointments Appear in the Dispatcher Console](#)
- [Salesforce Help: Fill Schedule Gaps](#)
- [Salesforce Help: Enable Multiday Service Appointments](#)
- [Salesforce Help: Schedule Appointments Using Priorities](#)

ServiceResource Custom Fields

Custom fields associated with a field service technician or crew in Field Service.

The standard fields are documented in the [ServiceResource](#) object reference.

Supported Calls

`create()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `update()`, `upsert()`

Special Access Rules

Field Service managed package must be installed.

Fields

Field	Details
<code>FSL__Efficiency__c</code>	Type double Properties Create, Filter, Nillable, Sort, Update Description The efficiency score or the work pace of the service resource. Enter a value from 0.1 to 10. An efficiency of 1 (default) means that the mobile worker works at a typical or average speed. An efficiency greater than 1 means that the mobile worker works faster than average. Less than 1 means that the mobile worker works slower than average. For more info, view the Estimate a Service Resource's Efficiency Help article.
<code>FSL__GanttLabel__c</code>	Type string Properties Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The description of the service resource in the Field Service dispatcher console's Gantt chart. This is shown under the service resource's name in the chart.
FSL__Online_Offset__c	Type double Properties Create, Filter, Nillable, Sort, Update Description The offset of how long the service resource is considered online since they last used or logged into the mobile app. This overrides the default value in the Field Service Settings page.
FSL__Picture_Link__c	Type url Properties Create, Filter, Group, Nillable, Sort, Update Description The URL link to the customer's picture used as the avatar in the Field Service dispatcher console's Gantt chart. If no URL is provided here, the Gantt chart uses the user avatar.
FSL__Priority__c	Type double Properties Create, Filter, Nillable, Sort, Update Description The priority of the service resource used to rank their appointments. The lower the number the higher the priority.
FSL__Travel_Speed__c	Type double Properties Create, Filter, Nillable, Sort, Update Description The average aerial travel speed of the service resource used to calculate the aerial travel time. This field overrides the default value in the Field Service Settings page. The units, selected in the Field Service Settings page, are KPH or MPH. Street level routing and predictive travel calculations don't use this field. They have their own settings.

SEE ALSO:

[Salesforce Object Reference: ServiceResource](#)

[Salesforce Help: Estimate a Service Resource's Efficiency](#)

ServiceResourceCapacity Custom Fields

Custom fields associated with the maximum number of scheduled hours or number of service appointments that a capacity-based service resource can complete within a specific time period.

The standard fields are documented in the [ServiceResourceCapacity](#) object reference.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `update()`, `upsert()`

Special Access Rules

Field Service managed package must be installed.

Fields

Field	Details
<code>FSL__HoursInUse__c</code>	Type double Properties Filter, Nillable, Sort Description The total number of hours of scheduled services occupied by the service resource. This is used by the Field Service dispatcher console to show how much resource capacity is used up and by the scheduler to prevent offering resources whose capacity is full. This field is updated when the scheduler runs and updates the <code>FSL__MinutesUsed__c</code> field. This is a calculated field. Formula <code>FSL__MinutesUsed__c / 60</code>
<code>FSL__Last_Updated_Epoch__c</code>	Type double Properties Create, Filter, Nillable, Sort, Update Description The date and time, in Epoch format, that the Capacity object was last updated. The Capacity object is updated when a service is scheduled on a capacity-based resource. This field is used by the Field Service dispatcher console.
<code>FSL__MinutesUsed__c</code>	Type double Properties Create, Filter, Nillable, Sort, Update

Field	Details
	Description The total number of minutes of scheduled services occupies by the service resource. This is used by the Field Service dispatcher console to show how much resource capacity is used up and by the scheduler to prevent offering resources whose capacity is full.
FSL__Work_Items_Allocated__c	Type double Properties Create, Filter, Nillable, Sort, Update Description The number of scheduled service appointments that fill the capacity. This is used by the Field Service dispatcher console to show how much resource capacity is used up.

SEE ALSO:

[Salesforce Object Reference: ServiceResourceCapacity](#)

ServiceTerritory Custom Fields

Custom fields associated with a geographic or functional region in which field service work can be performed in Field Service.

The standard fields are documented in the [ServiceTerritory](#) object reference.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service managed package must be installed.

Fields

Field	Details
FSL__Hide_Emergency_Map__c	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Controls if the Emergency Wizard map used by the Emergency Wizard global action is hidden for a service territory. This is for privacy purposes. If this is <code>true</code> , it shows a list of appointments with estimated time of arrivals instead. The default value is <code>false</code> .

Field	Details
FSL__NumberOfServices ToDripFeed__c	Type double Properties Create, Filter, Nillable, Sort, Update Description The drip feed rate to dispatch appointments. This is part of the drip feed dispatching feature. This value overrides the default value in the Field Service Settings page. For more info, view the Drip Feed Service Appointments Help article.
FSL__System_Jobs__c	Type multipicklist Properties Create, Filter, Nillable, Update Description The list of automators for scheduling jobs associated with a single territory. Possible values are the default or custom automator names configured in the Field Service Settings page. For example, if you create an optimization automator for Los Angeles called "LA_Optimize_1", this field is populated with LA_Optimize_1 for the LA service territory. This field is populated by the system. Don't edit this field.
FSL__TerritoryLevel__c	Type double Properties Create, Filter, Nillable, Sort, Update Description The territory hierarchy level of the polygon defining the service territory. A polygon is a custom shape drawn on the map to define the area of the territory. It can be nested inside another polygon creating a hierarchy. This field is populated by the system. Don't edit this field.

Internal Fields

These internal fields are used by the Field Service managed package for Street Level Routing calculations. Although they're publicly accessible, they must only be updated by the managed package.

- FSL__Internal_SLRGeolocation__Latitude__s
- FSL__Internal_SLRGeolocation__Longitude__s
- FSL__Internal_SLRGeolocation__c

SEE ALSO:

[Salesforce Object Reference: ServiceTerritory](#)

[Salesforce Help: Drip Feed Service Appointments](#)

TimeSlot Custom Fields

Custom fields associated with a period of time on a specified day of the week during which field service work can be performed in Field Service. Operating hours consist of one or more time slots.

The standard fields are documented in the [TimeSlot](#) object reference.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

Field Service managed package must be installed.

Fields

Field	Details
<code>FSL__Designated_Work_Boolean_Fields__c</code>	Type multipicklist Properties Create, Filter, Nillable, Update Description The type of designated work time slot. To convert a regular time slot to a designated one, use the calendar editor in the Visualforce page instead of editing this field manually in the record page. Possible values are: • None—Default Additional values are added by the managed package when a time slot is converted to a designated work type. The values are the API names of the designated work boolean fields on the service appointment.
<code>FSL__Slot_Color__c</code>	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The time slot color as it appears on the calendar editor in the Visualforce page. This field is populated by the system. Possible values are: • Amber • Asphalt • Black • Blue

Field	Details
	• Brown • Cyan • Green • Grey • Indigo • Lime • Orange • Pink • Purple • Red • Teal • Yellow

SEE ALSO:

[Salesforce Object Reference: TimeSlot](#)

WorkOrder Custom Fields

Custom fields associated with field service work to be performed for a customer.

The standard fields are documented in the [WorkOrder](#) object reference.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service managed package must be installed.

Fields

Field	Details
FSL__IsFillInCandidate__c	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Determines if this work order is considered as a candidate when filling in a schedule with the Fill-In Schedule feature. If a service appointment's parent record is a work order, this field must also be set to true for the appointment to be a candidate. Alternatively, you can create

Field	Details
	a custom checkbox field through the Field Service Settings page, instead of using this field, to evaluate whether this appointment is considered as a candidate. The custom checkbox field includes formula fields. The default value is <code>true</code>. For more info, view the Fill Schedule Gaps Help article.
FSL__Prevent_Geocoding_For_Chatter_Actions__c	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Determines if the work order's BeforeUpdate trigger disables the Chatter Action's geolocation cleanup on address change. This field is set to <code>false</code> after the cleanup completes. The default value is <code>false</code>.
FSL__Scheduling_Priority__c	Type double Properties Filter, Nillable, Sort Description The work order priority. The lower the value, the higher the priority. For more info, view the Schedule Appointments Using Priorities Help article. This is a calculated field. Formula <pre>CASE (TEXT(Priority), 'Critical' , 1 , 'High' ,2,'Medium' ,3 , 'Low', 4, null)</pre>
FSL__VisitingHours__c	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The visiting hours that define when service appointments associated with the work order can be scheduled. The visiting hours are enforced as long as the Visiting Hours work rule complies with the scheduling policy. Visiting hours are enforced only if the Visiting Hours work rule is part of the scheduling policy. For more info, view the Work Rule Type: Service Appointment Visiting Hours Help article. This is a relationship field. Relationship Name FSL__VisitingHours__r

Field	Details
	Relationship Type Lookup
	Refers To OperatingHours

SEE ALSO:

[Salesforce Object Reference: WorkOrder](#)

[Salesforce Help: Fill Schedule Gaps](#)

[Salesforce Help: Schedule Appointments Using Priorities](#)

[Salesforce Help: Work Rule Type: Service Appointment Visiting Hours](#)

WorkOrderLineItem Custom Fields

Custom fields associated with a subtask on a work order in field service.

The standard fields are documented in the [WorkOrderLineItem](#) object reference.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Field Service managed package must be installed.

Fields

Field	Details
FSL__IsFillInCandidate__c	Type boolean
	Properties Create, Defaulted on create, Filter, Group, Sort, Update
	Description Determines if this work order line item is considered as a candidate when filling in a schedule with the Fill-In Schedule feature. If a service appointment's parent record is a work order line item, this field must also be set to true for the appointment to be a candidate. Alternatively, you can create a custom checkbox field through the Field Service Settings page, instead of using this field, to evaluate whether this appointment is considered as a candidate. The custom checkbox field includes formula fields. The default value is <code>true</code> . For more info, view the Fill Schedule Gaps Help article.

Field	Details
FSL__VisitingHours__c	Type reference Properties Create, Filter, Group, Nillable, Sort, Update Description The visiting hours that define when service appointments associated with the work order line item can be scheduled. Visiting hours are enforced only if the Visiting Hours work rule is part of the scheduling policy. For more info, view the Work Rule Type: Service Appointment Visiting Hours Help article. This is a relationship field. Relationship Name FSL__VisitingHours__r Relationship Type Lookup Refers To OperatingHours

SEE ALSO:

- [Salesforce Object Reference: WorkOrderLineItem](#)
- [Salesforce Help: Fill Schedule Gaps](#)
- [Salesforce Help: Work Rule Type: Service Appointment Visiting Hours](#)

Supplementary Field Service Objects

A list of Field Service objects that support history tracking or sharing.

The following list links to reference information for these supplementary objects.



Note: Most objects are available only if Field Service is enabled. Objects not tied to Field Service enablement are shown with an asterisk (*).

- [AssetOwnerSharingRule*](#)
- [AssetShare*](#)
- [LinkedArticleHistory](#)
- [MaintenanceWorkRuleFeed](#)
- [MaintenanceWorkRuleHistory](#)
- [MaintenanceWorkRuleOwnerSharingRule](#)
- [MaintenanceWorkRuleShare](#)
- [OperatingHoursHistory](#)
- [ProductRequestHistory](#)
- [ProductRequestOwnerSharingRule](#)
- [ProductRequestShare](#)

- [ProductServiceCampaignFeed](#)
- [ProductServiceCampaignHistory](#)
- [ProductServiceCampaignOwnerSharingRule](#)
- [ProductServiceCampaignShare](#)
- [ProductServiceCampaignItemFeed](#)
- [ProductServiceCampaignItemHistory](#)
- [ProductServiceCampaignItemOwnerSharingRule](#)
- [ProductServiceCampaignItemShare](#)
- [ProductTransferHistory](#)
- [ProductTransferOwnerSharingRule](#)
- [ProductTransferShare](#)
- [ResourceAbsenceHistory](#)
- [ResourcePreferenceHistory](#)
- [ReturnOrderHistory](#)
- [ReturnOrderLineItemHistory](#)
- [ReturnOrderOwnerSharingRule](#)
- [ReturnOrderShare](#)
- [ServiceAppointmentHistory](#)
- [ServiceAppointmentOwnerSharingRule](#)
- [ServiceAppointmentShare](#)
- [ServiceCrewHistory](#)
- [ServiceCrewMemberHistory](#)
- [ServiceCrewOwnerSharingRule](#)
- [ServiceCrewShare](#)
- [ServiceResourceCapacityHistory](#)
- [ServiceResourceHistory](#)
- [ServiceResourceOwnerSharingRule](#)
- [ServiceResourceShare](#)
- [ServiceResourceSkillHistory](#)
- [ServiceTerritoryHistory](#)
- [ServiceTerritoryMemberHistory](#)
- [SkillRequirementHistory](#)
- [TimeSheetEntryHistory](#)
- [TimeSheetHistory](#)
- [TimeSheetOwnerSharingRule](#)
- [TimeSheetShare](#)
- [TimeSlotHistory](#)
- [WorkOrderHistory*](#)
- [WorkOrderLineItemHistory*](#)
- [WorkOrderShare*](#)

- [WorkTypeGroupHistory](#)
- [WorkTypeGroupMemberHistory](#)
- [WorkTypeGroupShare](#)
- [WorkTypeHistory](#)
- [WorkTypeOwnerSharingRule](#)
- [WorkTypeShare](#)

Field Service REST API

The following endpoints can be used to manage your field service implementation.

IN THIS SECTION:

[Field Service Flow](#)

Returns information corresponding to a field service flow. This resource is available in REST API version 42.0 and later.

[Field Service Mobile Settings](#)

Returns information about the Field Service mobile app settings assigned to the profile of the context user. This resource is available in REST API version 42.0 and later.

[Service Report Template](#)

Returns information corresponding to one or more service report templates in field service. This resource is available in REST API version 40.0 and later.

[Appointment Bundling REST APIs](#)

Use the Appointment Bundling REST APIs to create bundles, remove bundle members, start the automated process for creating bundles, unbundle one or more bundles, and update a bundle.

Field Service Flow

Returns information corresponding to a field service flow. This resource is available in REST API version 42.0 and later.

This resource accepts the API name of a flow definition and returns information about the active flow version. If no flow version is activated, the most recent version is returned. Flow elements can be translated to the language of the user invoking this API by using Translation Workbench. If a particular flow version's elements have been translated, the data returned uses the language of the query. To learn more, see [Flow Components in Translation Workbench](#).

This resource is available to API users with Field Service mobile app user permissions and the Run Flows user permission.

URI

`/services/data/vXX.X/support/fieldservice/Flow?developerNames=Flow Unique Name`

Formats

JSON, XML

HTTP Method

GET

Authentication

Authorization: Bearer **token**

Parameters

Parameter	Description
developerNames	The unique name for the flow definition. Currently, you can specify only one unique name using this parameter.

Response Body

The following table describes the main elements of the response body.

Field	Type	Description
Flows	Array	List of flows
CreatedDate	String	Date the flow was created
Description	String	Description of the flow
FullName	String	Full name for the flow
Id	String	Unique ID for the flow
LastModifiedDate	String	Date the flow was last modified
Metadata		
	Choices	Array
	ChoiceText	String
	DataType	String
	Name	String
	Screens	Array of objects
	AllowBack	Boolean
	AllowFinish	Boolean
	AllowPause	Boolean
	Fields	Array of objects
	HelpText	String
	Label	String
	LocationX	Number
	LocationY	Number
	Name	String
ProcessType	String	Type of process for the flow; in this case, it should be "FieldServiceMobile"

Field	Type	Description
Status	String	Status of the flow
VersionNumber	Number	Version number for the flow

 **Example:** This sample request uses the unique name “FS_Flow” for the flow definition.

The response contains two screens:

- Screen 1
- Screen 2

Screen 1 has two fields:

- Field 1: a text field
- RadioButton: a field with two choices

Screen 2 has one field:

- Field 2: a text field

Request

GET

/services/data/v42.0/support/fieldservice/Flow?developerNames=FS_Flow

Response

```
{
  "flows" : [ {
    "Id" : "301R000000008grIAA",
    "DefinitionId" : "300R00000004OLFIA2",
    "MasterLabel" : "FS_Flow",
    "ManageableState" : "unmanaged",
    "VersionNumber" : 1,
    "Status" : "Draft",
    "Description" : "FS Flow",
    "ProcessType" : "FieldServiceMobile",
    "CreateDate" : "2017-12-05T19:22:16.000+0000",
    "CreatedById" : "005R0000000J2glIAC",
    "LastModifiedDate" : "2017-12-05T19:22:16.000+0000",
    "LastModifiedById" : "005R0000000J2glIAC",
    "Metadata" : {
      "actionCalls" : null,
      "apexPluginCalls" : null,
      "assignments" : null,
      "choices" : [ {
        "choiceText" : "Choice A",
        "dataType" : "String",
        "description" : null,
        "name" : "Choice_A",
        "processMetadataValues" : null,
        "userInput" : null,
        "value" : null
      }, {
        "choiceText" : "Choice B",
```



```

        "dataType" : "String",
        "description" : null,
        "name" : "Choice_B",
        "processMetadataValues" : null,
        "userInput" : null,
        "value" : null
    } ],
    "constants" : null,
    "decisions" : null,
    "description" : "FS Flow",
    "dynamicChoiceSets" : null,
    "formulas" : null,
    "fullName" : "FS_Flow-1",
    "interviewLabel" : "FS_Flow {!$Flow.CurrentDateTime}",
    "label" : "FS Flow",
    "loops" : null,
    "module" : null,
    "processMetadataValues" : null,
    "processType" : "FieldServiceMobile",
    "recordCreates" : null,
    "recordDeletes" : null,
    "recordLookups" : null,
    "recordUpdates" : null,
    "screens" : [ {
        "allowBack" : true,
        "allowFinish" : true,
        "allowPause" : true,
        "connector" : {
            "processMetadataValues" : null,
            "targetReference" : "Screen_2"
        },
        "description" : null,
        "fields" : [ {
            "choiceReferences" : null,
            "dataType" : "String",
            "defaultSelectedChoiceReference" : null,
            "defaultValue" : null,
            "description" : null,
            "extensionName" : null,
            "fieldText" : "Field 1",
            "fieldType" : "InputField",
            "helpText" : "Help Text 1",
            "inputParameters" : [ ],
            "isRequired" : false,
            "isVisible" : null,
            "name" : "Field_1",
            "outputParameters" : [ ],
            "processMetadataValues" : null,
            "scale" : null,
            "validationRule" : null
        }, {
            "choiceReferences" : [ "Choice_A", "Choice_B" ],
            "dataType" : "String",
            "defaultSelectedChoiceReference" : null,

```

```

        "defaultValue" : null,
        "description" : null,
        "extensionName" : null,
        "fieldText" : "Pick a choice",
        "fieldType" : "RadioButtons",
        "helpText" : "Pick a Choice help text",
        "inputParameters" : [ ],
        "isRequired" : false,
        "isVisible" : null,
        "name" : "Pick_a_choice",
        "outputParameters" : [ ],
        "processMetadataValues" : null,
        "scale" : null,
        "validationRule" : null
    } ],
    "helpText" : null,
    "label" : "Screen 1",
    "locationX" : 189,
    "locationY" : 178,
    "name" : "Screen_1",
    "pausedText" : null,
    "processMetadataValues" : null,
    "rules" : null,
    "showFooter" : true,
    "showHeader" : true
}, {
    "allowBack" : true,
    "allowFinish" : true,
    "allowPause" : true,
    "connector" : null,
    "description" : null,
    "fields" : [ {
        "choiceReferences" : null,
        "dataType" : "String",
        "defaultSelectedChoiceReference" : null,
        "defaultValue" : null,
        "description" : null,
        "extensionName" : null,
        "fieldText" : "Field 1",
        "fieldType" : "InputField",
        "helpText" : null,
        "inputParameters" : [ ],
        "isRequired" : false,
        "isVisible" : null,
        "name" : "Field_1",
        "outputParameters" : [ ],
        "processMetadataValues" : null,
        "scale" : null,
        "validationRule" : null
    } ],
    "helpText" : null,
    "label" : "Screen 2",
    "locationX" : 437,
    "locationY" : 289,

```

```

        "name" : "Screen_2",
        "pausedText" : null,
        "processMetadataValues" : null,
        "rules" : null,
        "showFooter" : true,
        "showHeader" : true
      } ],
      "startElementReference" : "Screen_1",
      "steps" : null,
      "subflows" : null,
      "textTemplates" : null,
      "urls" : null,
      "variables" : null,
      "waits" : null
    },
    "FullName" : "FS_Flow-1"
  } ]
}

```

Field Service Mobile Settings

Returns information about the Field Service mobile app settings assigned to the profile of the context user. This resource is available in REST API version 42.0 and later.

The Field Service mobile app for Android and iOS comes with a variety of customizable settings, including branding colors, geolocation accuracy, app extensions, and date picker display. You can assign unique configurations of those settings to user profiles for a customized Field Service mobile experience. For example, configure separate settings to address the mobile needs of contractors, technicians of varying levels, and service crew leaders.

The `FieldServiceMobileSettings` object represents a configuration of the mobile app settings. The assignment of a settings configuration to a user profile is stored in a child object, `MobileSettingsAssignment`. A `FieldServiceMobileSettings` record can also have zero or more child `AppExtension` records. The `userSettings` extension enables the return of user profile settings information.

Each user profile can be associated with only one `FieldServiceMobileSettings` record. If a user's profile isn't explicitly assigned a settings record, the profile uses the default settings.

This resource returns information about the `FieldServiceMobileApp` settings assigned to the profile of the user in the context of which the call is made. It is available in orgs where Field Service is enabled, and can be used for users who have one or more of the following:

- Field Service Mobile permission set license
- View Setup user permission
- Customize Application user permission
- System Administrator profile

This resource can also be executed in the context of an Experience Builder site user with the Field Service Mobile permission set license.

URI

`/services/data/vXX.X/objects/FieldServiceMobileSettings/userSettings`

Formats

JSON, XML

HTTP Method

GET

Authentication

Authorization: Bearer ***token***

Parameters

None.

Response Body

A FieldServiceMobileSettings record contains the following properties. All string properties have a maximum length of 7 characters unless otherwise noted. For descriptions, see [FieldServiceMobileSettings](#) in the *Salesforce Object Reference*.

Field	Type
BgGeoLocationAccuracy	Enum Values: Medium, Coarse, VeryCoarse
BgGeoLocationMinUpdateFreqMins	Integer
BrandInvertedColor	String
ContrastInvertedColor	String
ContrastPrimaryColor	String
ContrastQuaternaryColor	String
ContrastQuinaryColor	String
ContrastSecondaryColor	String
ContrastTertiaryColor	String
DefaultListViewDeveloperName	String (maximum length: 255)
FeedbackPrimaryColor	String
FeedbackSecondaryColor	String
FeedbackSelectedColor	String
FutureDaysInDatePicker	Integer
GeoLocationAccuracy	Enum Values: Fine, Medium, Coarse
GeoLocationMinUpdateFreqMins	Integer
Id	String
IsAssignmentNotification	Boolean
IsDefault	Boolean
IsDispatchNotification	Boolean
IsScheduleViewResourceAbsences	Boolean Available in API version 55.0 and later.

Field	Type
IsSendLocationHistory	Boolean
IsShowEditFullRecord	Boolean
IsUseSalesforceMobileActions	Boolean
MetadataCacheTimeDays	Integer
NavbarBackgroundColor	String
NavbarInvertedColor	String
PastDaysInDatePicker	Integer
PrimaryBrandColor	String
RecordDataCacheTimeMins	Integer
SecondaryBrandColor	String
TimeIntervalSetupMins	Enum Values: 1, 5, 15, 20, 30, 60
UpdateScheduleTimeMins	Integer

If any of the following is true, an error message is returned:

- The user does not have access to FieldServiceMobileSettings
- Field Service is not enabled in the org
- An internal server error occurs



Example:

Request

GET

/services/data/v56.0/objects/FieldServiceMobileSettings/userSettings

Response

```
{
  "attributes" : {
    "type" : "FieldServiceMobileSettings",
    "url" :
"/services/data/v46.0/objects/FieldServiceMobileSettings/0MfRM0000000rpR0AQ"
  },
  "Id" : "0MfRM0000000rpR0AQ",
  "IsDeleted" : false,
  "DeveloperName" : "Field_Service_Mobile_Settings",
  "Language" : "en_US",
  "MasterLabel" : "Field Service Mobile Settings",
  "CreateDate" : "2019-02-17T05:20:47.000+0000",
  "CreatedById" : "005RM000001hTsLYAU",
  "LastModifiedDate" : "2019-03-22T22:48:17.000+0000",
```

```

    "LastModifiedById" : "005RM000001k2kpYAA",
    "SystemModstamp" : "2019-03-22T22:48:17.000+0000",
    "NavbarBackgroundColor" : "#803ABE",
    "BrandInvertedColor" : "#FFFFFF",
    "FeedbackPrimaryColor" : "#C23934",
    "FeedbackSecondaryColor" : "#13C4A3",
    "PrimaryBrandColor" : "#803ABE",
    "SecondaryBrandColor" : "#2A7AB0",
    "ContrastPrimaryColor" : "#000000",
    "ContrastSecondaryColor" : "#444444",
    "ContrastTertiaryColor" : "#9FAAB5",
    "ContrastQuaternaryColor" : "#E6E6EB",
    "ContrastQuinaryColor" : "#EEEEEE",
    "ContrastInvertedColor" : "#FFFFFF",
    "IsScheduleViewResourceAbsences" : true,
    "IsSendLocationHistory" : false,
    "GeoLocationMinUpdateFreqMins" : 10,
    "GeoLocationAccuracy" : "Medium",
    "RecordDataCacheTimeMins" : 240,
    "MetadataCacheTimeDays" : 7,
    "UpdateScheduleTimeMins" : 30,
    "IsShowEditFullRecord" : false,
    "TimeIntervalSetupMins" : "15",
    "DefaultListViewDeveloperName" : null,
    "NavbarInvertedColor" : "#FFFFFF",
    "FeedbackSelectedColor" : "#FFFFFF",
    "FutureDaysInDatePicker" : 45,
    "PastDaysInDatePicker" : 45,
    "IsDefault" : true,
    "BgGeoLocationMinUpdateFreqMins" : 60,
    "BgGeoLocationAccuracy" : "Coarse",
    "IsUseSalesforceMobileActions" : false,
    "IsAssignmentNotification" : false,
    "IsDispatchNotification" : true
  }

```

Service Report Template

Returns information corresponding to one or more service report templates in field service. This resource is available in REST API version 40.0 and later.

This resource allows users to input one or more service report template IDs, and responds with information about the templates. Service reports can be generated for work orders, work order line items, and service appointments.

ServiceReportTemplate is available to API users in orgs where Field Service is enabled. It is also enabled for Experience Builder sites.

URI

/services/data/vXX.X/support/fieldservice/ServiceReportTemplate

Formats

JSON

HTTP Method

GET

Authentication

Authorization: Bearer *token*

Parameters

Parameter	Description
<code>templateIds</code>	The list of service report template IDs, provided as comma-separated string values. The API response shows a failure if any template ID is invalid.
<code>showDefault</code>	A boolean parameter indicating whether the default service report template is returned in the response. If <code>true</code> , the default template is included; if <code>false</code> (default), the default template is not included. For example, if a request lists the IDs of two non-default templates and <code>showDefault=true</code> , the response will include information for three templates.
<code>templateTypes</code>	The type of record that the report template can be used for: ServiceAppointment, WorkOrder, or WorkOrderLineItem. Service report templates include four customizable sub-templates: • <code>WO</code>—Work Order • <code>WOLI</code>—Work Order Line Item • <code>WO_SA</code>—Service Appointment for Work Order • <code>WOLI_SA</code>—Service Appointment for Work Order Line Item

Response Body

A service report has three main areas which cannot be deleted: **Header**, **Body** and **Footer**. Each area must contain at least one component—a **section**, **related list**, or **signature**—though the component can be empty. The Header and Footer come with a predefined section, and additional components cannot be added to them. The Body can contain one or more of each component.

Section components have the following properties:

- `title`: String
- `hideTitle`: Boolean
- `hideFieldLabels`: Boolean
- `columns`: Enum {1, 2}
- `rightAlignment`: Enum {true,false}
- `fields`: Array of {Field}

A **field** can be added to a section, related list, or signature. Columns in related lists are transformed into fields in the API. Field components have the following properties:

- `fieldType`: Enum {rta, entityField, blank}
- `Attributes`:
 - If the field type is `rta`, the field also contains `{ "richTextValue" : "<html value>" }`
 - If the field type is `entityField`, the field also contains `{ "entityName" : "<WorkOrder>", "fieldName" : "Account" }`
 - If the field type is `blank`, no other information is required

- Position:
 - row
 - column

Related list components have the following properties:

- title: String
- hideTitle: Boolean
- entityName: String
- relatedListName: String
- relatedEntityName: String
- fields: Array of {Field Name (String), Column Position (Integer)}

Signature components have the following properties:

- title: String
- hideTitle: Boolean
- hideFieldLabels: Boolean
- columns: Enum {1, 2}
- rightAlignment: Enum {true,false}
- signatureType: Dynamic Enum {<technician 1>, <dispatcher 1>}
- fields: Array of {Field Name (String), Position (Row,Column)}



Example: This example request uses two service report template IDs.

Request

GET

/services/data/v42.0/support/fieldservice/ServiceReportTemplate?templateIds=0SLxx0000000ABC,
0SLR000000001QtOAI&showDefault=false&templateTypes=ServiceAppointment

Response

```
{
  "serviceReportTemplates": [
    {
      "defaultTemplate": false,
      "error": {
        "errorCode": "INVALID_TEMPLATE_ID",
        "errorMessage": "The Service Report Template ID is invalid."
      },
      "subTemplates": [],
      "templateId": "0SLxx0000000ABC"
    },
    {
      "defaultTemplate": false,
      "error": null,
      "subTemplates": [
        {
          "regions": [
            {
              "sections": [
                {

```



```

        "columns": 2,
        "hideFieldLabels": false,
        "hideTitle": false,
        "items": [
            {
                "position": {
                    "column": 0,
                    "row": 0
                },
                "richText": "<img alt=\"User-added image\"
src=\"https://mobile1.file.force.com/servlet/rtaImage?eid=0QR000000008oZ&ap;feid=Data&ap;refid=0EMF00000000DG\"></img>",
                "type": "rta"
            },
            {
                "position": {
                    "column": 1,
                    "row": 0
                },
                "richText": "<u><i>Salesforce.com</i></u>",
                "type": "rta"
            }
        ],
        "position": 0,
        "rightAlign": false,
        "title": "Service Report",
        "type": "section"
    }
],
"type": "HEADER"
},
{
    "sections": [
        {
            "columns": 2,
            "hideFieldLabels": false,
            "hideTitle": true,
            "items": [],
            "position": 0,
            "rightAlign": false,
            "showPageNumber": false,
            "title": "Footer Section",
            "type": "section"
        }
    ],
    "type": "FOOTER"
},
{
    "sections": [
        {
            "columns": 2,
            "hideFieldLabels": false,
            "hideTitle": false,
            "items": [

```

```

    {
      "entityName": "ServiceAppointment",
      "label": "Account",
      "name": "AccountId",
      "position": {
        "column": 0,
        "row": 0
      },
      "type": "entityField"
    },
    ...
    {
      "position": {
        "column": 1,
        "row": 0
      },
      "richText": "Prepared By:",
      "type": "rta"
    }
  ],
  "position": 0,
  "rightAlign": false,
  "title": "Account & Contact Information",
  "type": "section"
},
{
  "entityName": "WorkOrder",
  "filterCriteria": {
    "conditions": [
      {
        "field": "Status",
        "operation": "includes",
        "position": 0,
        "values": [
          "In Progress",
          "Completed"
        ]
      }
    ]
  }
},
"hideTitle": false,
"items": [
  {
    "column": 0,
    "label": "Work Order Line Item Number",
    "name": "LineItemNumber"
  },
  ...
  {
    "column": 6,
    "label": "Unit Price",
    "name": "UnitPrice"
  }
],

```

```

        "position": 5,
        "relatedEntityName": "WorkOrderLineItem",
        "relatedListName": "WorkOrderLineItems",
        "title": "Work Order Line Items",
        "type": "relatedList"
    },
    {
        "columns": 1,
        "hideFieldLabels": false,
        "hideTitle": false,
        "items": [
            {
                "entityName": "DigitalSignature",
                "label": "Signature",
                "name": "Document",
                "position": {
                    "column": 0,
                    "row": 0
                },
                "type": "entityField"
            },
            {
                "entityName": "DigitalSignature",
                "label": "Signed By",
                "name": "SignedBy",
                "position": {
                    "column": 0,
                    "row": 1
                },
                "type": "entityField"
            },
            {
                "entityName": "DigitalSignature",
                "label": "Date",
                "name": "SignedDate",
                "position": {
                    "column": 0,
                    "row": 2
                },
                "type": "entityField"
            }
        ],
        "position": 9,
        "rightAlign": false,
        "signatureType": "Default",
        "signatureTypeLabel": "Default",
        "title": "Customer Signature",
        "type": "signature"
    }
],
"type": "BODY"
}
],
"subTemplateType": "WO_SA"

```

```

    },
    {
      ...
      "subTemplateType": "WOLI_SA"
    }
  ],
  "templateId": "0SLR000000001QtOAI"
}
]
}

```

Appointment Bundling REST APIs

Use the Appointment Bundling REST APIs to create bundles, remove bundle members, start the automated process for creating bundles, unbundle one or more bundles, and update a bundle.

Limitations

Create Bundle, Remove Bundle Members, Unbundle, Unbundle Multiple, Update Bundle

- 1000 API calls within 24 hours.
- 50 API calls per engineer license within 24 hours (in addition to the 1000 API calls).
- 10 concurrent API calls.

Start Batch

- One API call per hour, per territory.

IN THIS SECTION:

[Create Bundle](#)

Creates a bundle of service appointments manually. This resource accepts service appointment IDs and a manual bundling policy ID. The manual bundling policy specifies the bundling rules and has to be marked for manual bundling. This resource returns the ID of the bundle service appointment. This API is not supported in Gov Cloud. This endpoint is available in version 54.0 and later of the Salesforce API.

[Remove Bundle Members](#)

Removes one or more service appointments from an existing bundle. This resource accepts service appointment IDs. This API is not supported in Gov Cloud. This endpoint is available in version 54.0 and later of the Salesforce API.

[Start Batch](#)

Creates service appointment bundles automatically. The resource uses appointment bundle policies that are marked for automatic bundling. Service appointments that are already bundled manually stay untouched by this API. The resource returns the success or failure message of starting the batch process of automatic bundling. This API is not supported in Gov Cloud. This endpoint is available in version 54.0 and later of the Salesforce API.

[Unbundle](#)

Unbundle a bundle of service appointments. This resource accepts a bundle ID. This API is not supported in Gov Cloud. This endpoint is available in version 54.0 and later of the Salesforce API.

[Unbundle Multiple](#)

Unbundle one or more bundles of service appointments. This resource accepts one or more bundle IDs. This API is not supported in Gov Cloud. This endpoint is available in version 54.0 and later of the Salesforce API.

Update Bundle

Adds service appointments to an existing bundle. This resource accepts a bundle ID and service appointment IDs. This API is not supported in Gov Cloud. This endpoint is available in version 54.0 and later of the Salesforce API.

Create Bundle

Creates a bundle of service appointments manually. This resource accepts service appointment IDs and a manual bundling policy ID. The manual bundling policy specifies the bundling rules and has to be marked for manual bundling. This resource returns the ID of the bundle service appointment. This API is not supported in Gov Cloud. This endpoint is available in version 54.0 and later of the Salesforce API.

Add a Remote Site

- 1. From Setup, in the Quick Find box, enter *Security*, and then select **Remote Site Settings**.
- 2. Click **New Remote Site**.
- 3. Enter a name.
- 4. For Remote Site URL, enter *https://api.salesforce.com/*
- 5. Save your changes.

URI

<host>/bundleflow/api/v1.0/bundle

Format

JSON

HTTP Method

POST

Authentication

Authorization: Bearer <token>

If the authentication token isn't valid, a 401 HTTP status is returned.

Headers

- request.setHeader('x-sfdc-tenant-id', 'core/prod/ORG-ID_18_Characters');—Field Service functional domain
- sf_api_version—Salesforce API version. Minimum version is 53.0.
- Authorization—Authorization token.
- Content-Type—File format. Valid value is JSON.
- Referer—Org domain URL.

Request Body

Properties

Field	Type	Description
user	String	Optional. User name of user sending the request.

Field	Type	Description
initiate	String	Type of bundle request. Valid value: manual
bundlePolicyId	String	ID of the related bundle policy containing the bundling rules.
saRequestPayloads	Array	Details of service appointments that are bundled.
serviceAppointmentId	String	ID of service appointments to be bundled.
action	String	Bundling action performed on this service appointment. Valid value: add



Example: JSON Example

This example shows a manual bundle request of two service appointments.

```
{
  "user": "Mishal",
  "initiate": "manual",
  "bundlePolicyId": "7sTx00000000006EAA",
  "saRequestPayloads": [
    {
      "serviceAppointmentId": "08px000000NzbmsAAB",
      "action": "add"
    },
    {
      "serviceAppointmentId": "08pT3000000006LLIAY",
      "action": "add"
    }
  ]
}
```

Response Body

Properties

Field	Type	Description
bundleId	String	ID of the new bundle. If the bundle isn't created, this field is null.
responsePayloads	Array	Details of the bundled objects.
objectName	String	Type of object. Output: ServiceAppointment
objectId	String	ID of the service appointment.
action	String	Action performed on the object.

Field	Type	Description
status	String	Specifies if this object is bundled correctly. Output: SUCCESS or FAIL
messageCode	Number	Success or error code of bundling the service appointment. 0 means success. The error codes are listed in Error Responses on page 516.
message	String	Success or error message of bundling the object.
status	String	Status of the request. Output: SUCCESS or FAIL
messageCode	Number	Success or error code of the request. 0 means success. The error codes are listed in Error Responses on page 516.
message	String	Success or error message of the request.
messageAdditionalInfo	String	More information provided by Salesforce.



Example: JSON Example: Successful Response

This example shows the output of two service appointments bundled successfully.

```
{
  "bundleId": "08px000000NzdH8AAJ",
  "responsePayloads": [
    {
      "objectName": "ServiceAppointment",
      "objectId": "08px000000NzbmsAAB",
      "action": "add",
      "status": "SUCCESS",
      "messageCode": "NONE",
      "message": "Success",
      "messageParams": [

    ]
  },
  {
    "objectName": "ServiceAppointment",
    "objectId": "08pT300000006LLIAY",
    "action": "add",
    "status": "SUCCESS",
    "messageCode": "NONE",
    "message": "Success",
    "messageParams": [

  ]
}
],
"status": "SUCCESS",
"messageCode": "NONE",
```

```

    "message": "Success"
  }

```



Example: JSON Example: Fail Response

This example shows the output of a request that's trying to bundle a service appointment that's already a bundle service appointment. A service appointment can't be part of two bundles, so the response status is **FAIL**.

```

{
  "bundleId":null,
  "responsePayloads":[
    {
      "objectName":"ServiceAppointment",
      "objectId":"08px000000NzdH8AAJ",
      "action":"add",
      "status":"FAIL",
      "messageCode":"SA_ALREADY_BUNDLED",
      "message":"Is already a bundle service appointment.",
      "messageParams":[]
    }
  ],
  "status":"FAIL",
  "messageCode":"ERROR_CREATING_BUNDLE",
  "message":"We couldn't bundle the service appointment.",
  "messageAdditionalInfo": ""
}

```

Table 1: Error Responses

Code	Name	Message
0	NONE	<Service Appointment ID#>: Success.
1	UNSUPPORTED_VERSION	Ask your admin to check the API version, then try again.
2	SA_PAYLOAD_SHOULD_CONTAIN_ADD_ACTIONS_ONLY	Create bundles using add actions only.
3	EMPTY_BUNDLE_POLICY	Enter an ID for the bundlePolicyId.
4	EMPTY_BUNDLE_INITIATE	Specify manual or auto for the initiate entry.
5	EMPTY_BUNDLE_SERVICE_APPOINTMENT	Add a bundle service appointment.
6	SA_PAYLOAD_DUPLICATED_SAS	Remove duplicate service appointment IDs.
7	SA_IS_BUNDLE	<Service Appointment ID#>: Is already a bundle service appointment.
8	SA_ALREADY_BUNDLED	<Service Appointment ID#>: Is a bundle member service appointment in another bundle.

Code	Name	Message
9	STATUS_FORBIDDEN	<Service Appointment ID#>: Can't be bundled in the current status.
10	SA_PAYLOAD_SHOULD_NOT_CONTAIN_MIX_ACTIONS	Send separate requests for add and remove actions.
11	SA_IS_NOT_BUNDLE	Select a bundle service appointment and try again.
12	EXCEEDED_BUNDLE_MEMBERS_LIMIT	You reached the maximum number of service appointments for this bundle. Remove some of them and try again.
13	SA_PAYLOAD_SHOULD_CONTAIN_ONLY_BUNDLE_MEMBERS	To remove bundle members from a bundle, include only service appointments that are bundle members.
14	ERROR_IN_BASIC_VALIDATIONS	We couldn't bundle the service appointments.
15	ERROR_IN_LIMIT_VALIDATIONS	N/A
16	ERROR_IN_CRITERIA_VALIDATIONS	N/A
17	ERROR_IN_SERVICE_TERRITORY_VALIDATIONS	Confirm or revise the service territories of the selected service appointments. Or ask your admin for help.
18	ERROR_IN_RESTRICTION_VALIDATIONS	Confirm or revise the selected service appointments. Or ask your admin to check the restriction policies.
19	EXCEEDED_BUNDLE_DURATION_LIMIT	N/A
20	EMPTY_BODY	Fill in the request.
21	SA_IS_NOT_BUNDLED	<Service Appointment ID#>: Can't remove a service appointment that isn't in the bundle.
22	ERROR_IN_AGGREGATION	Confirm or revise the selected service appointments. Or ask your admin to check the aggregation policies.
23	ERROR_IN_PROPAGATION	Confirm or revise the selected service appointments. Or ask your admin to check the propagation policies.
24	GET_POLICY_FAILURE	Ask your admin to check the bundle policies, then try again.
25	GET_CONFIG_FAILURE	Ask your admin to check the bundle config, then try again.

Code	Name	Message
26	GET_BUNDLE_INFO_FAILURE	Try again later.
27	ERROR_IN_LOGIC	We couldn't bundle the service appointments.
28	EMPTY_BUNDLE_MEMBERS	Add bundle member service appointments.
29	EMPTY_BUNDLE_PARENT_WORK_ORDER_ID	Add the work order ID for the bundle service appointment.
30	UNBUNDLE_REQUEST_ASSIGNED_RESOURCES_NOT_COMPATIBLE	To unbundle a scheduled bundle, send assigned resources for each service appointment.
31	EMPTY_OLD_VALUES	To update a bundle, send the current values.
32	EMPTY_NEW_VALUES	To update a bundle, send the new values.
33	SA_PAYLOAD_SHOULD_CONTAIN_AT_LEAST_ONE_ADD_OR_REMOVE_ACTION	To update a bundle, send at least one add or remove action.
34	FORBIDDEN_FIELDS_IN_POLICY_OBJECT	Confirm or revise the selected service appointments. Or ask your admin to check the field names selected in the related policies.
35	EMPTY_SERVICE_TERRITORY	Add a Service Territory ID.
36	EMPTY_START_TIME	Add the interval's start time.
37	EMPTY_END_TIME	Add the interval's end time.
38	ERROR_IN_BUNDLE_POLICY_RFC	Confirm or revise the selected service appointments. Or ask your admin to check the recordset filter criteria in the bundle policy.
39	ERROR_IN_TIME_ZONE_VALIDATION	Select service appointments that are in the same time zone. Or ask your admin for help.
40	ERROR_IN_LOGIC_UNBUNDLE	We couldn't unbundle the service appointment.
41	ERROR_CREATING_BUNDLE	We couldn't bundle the service appointments.
42	ERROR_ADDING_TO_BUNDLE	We couldn't add the service appointment to the bundle.
43	ERROR_REMOVING_FROM_BUNDLE	We couldn't remove the service appointment from the bundle.

Code	Name	Message
44	ERROR_UNBUNDLING	We couldn't unbundle the service appointment.
58	ERROR_IN_MSW_VALIDATIONS	We can't bundle service appointments with dependencies. Remove the dependencies and try again.
-500	http error	Hmm... Something went wrong. Try again.

Code Sample from Apex

```

public static Map<String, String> createSABundle() {
    String apiVersion = '54.0'; // Spring '22
    String bundleApi = '/bundleflow/api/v1.0/bundle';
    String host = {Namespace}.BundleLogic.getBundlerFalconEnvironment();
    String ref = URL.getOrgDomainUrl().toExternalForm();

    String bundleService = host + bundleApi;

    // Create HTTP request
    HttpRequest request = new HttpRequest();
    request.setEndpoint(bundleService);
    request.setMethod('POST');
    request.setHeader('sf_api_version', apiVersion);
    request.setHeader('Content-Type', 'application/json' );
    // NOTE: This user must have 'Field Service Integration' permissions.
    request.setHeader('Authorization', 'Bearer ' + UserInfo.getSessionId());
    request.setHeader('Referer', ref);
    request.setHeader('x-sfdc-tenant-id', 'core/prod/ORG-ID_18_Characters');
    request.setTimeout(120000);

    // Create the body
    Map<String, Object> body = new Map<String, Object>();
    body.put('initiate', 'manual');
    body.put('bundlePolicyId', '7sT9A0000004DX6UAM'); // NOTE: Use a real bundle policy ID.

    Map<String, Object> saList = new Map<String, Object>();
    saList.put('serviceAppointmentId', '08p9A0000005LEGQA2'); // NOTE: Use a real Service
Appointment ID.
    saList.put('action', 'add');

    List<Object> objectsList = new List<Object>();
    objectsList.add(saList);

    body.put('saRequestPayloads', objectsList);

    String reqBody = JSON.serialize(body);
    System.debug(body);
    request.setBody(reqBody);
}

```

```

    HttpResponse response = new Http().send(request);

    // Parse the JSON response

    // Handle a redirect message
    while (response.getStatusCode() == 302) {
        request.setEndpoint(response.getHeader('Location'));
        response = new Http().send(request);
    }

    Map<String, String> returnValue = new Map<String, String>();
    returnValue.put('statusCode', String.valueOf(response.getStatusCode()));

    // Return value when we don't get a success response
    if (response.getStatusCode() != 200) {
        returnValue.put('message', 'The status code returned was not expected: ' +
response.getStatusCode() + ' ' + response.getStatus());
        System.debug(returnValue.get('message'));
        return (returnValue);

    // Return value when we do get a success response
    } else {
        returnValue.put('message', response.getBody());
        System.debug(response.getBody());
        return (returnValue);
    }
}

Map<String, String> response = createSABundle();
System.debug(response);

```

Remove Bundle Members

Removes one or more service appointments from an existing bundle. This resource accepts service appointment IDs. This API is not supported in Gov Cloud. This endpoint is available in version 54.0 and later of the Salesforce API.

Add a Remote Site

1. From Setup, in the Quick Find box, enter *Security*, and then select **Remote Site Settings**.
2. Click **New Remote Site**.
3. Enter a name.
4. For Remote Site URL, enter *https://api.salesforce.com/*
5. Save your changes.

URI

<host>/bundleflow/api/v1.0/bundle/remove

Format

JSON

HTTP Method

PATCH

Authentication

Authorization: Bearer *<token>*

If the authentication token isn't valid, a 401 HTTP status is returned.

Headers

`request.setHeader('x-sfdc-tenant-id', 'core/prod/ORG-ID_18_Characters');`—Field Service functional domain

`sf_api_version`—Salesforce API version. Minimum version is 53.0.

`Authorization`—Authorization token.

`Content-Type`—File format. Valid value is `JSON`.

`Referer`—Org domain URL.

Request Body

Properties

Field	Type	Description
user	String	Optional. User name of user sending the request.
initiate	String	Type of bundle request. Valid value: <code>manual</code>
saRequestPayloads	Array	Details of the service appointments to be removed from a bundle.
serviceAppointmentId	String	ID of a service appointment.
action	String	Bundling action performed on this service appointment. Valid values: <code>remove</code>

 Example: JSON Example

This example shows a manual request to remove two service appointments from a bundle.

```
{
  "user": "Misha",
  "initiate": "manual",
  "saRequestPayloads": [
    {
      "serviceAppointmentId": "08px000000NzbmKAAB",
      "action": "remove"
    },
    {
      "serviceAppointmentId": "08pT300000006LLIAY",
      "action": "remove"
    }
  ]
}
```

```
    ]
  }
```

Response Body

Properties

Field	Type	Description
bundleId	String	ID of the bundle to update. If the update isn't successful, it's <code>null</code> .
responsePayloads	Array	Details of the service appointment objects that are part of the bundle request.
objectName	String	Type of object. Output: <code>ServiceAppointment</code>
objectId	String	ID of the service appointment.
action	String	Bundling action performed on this service appointment. Valid values: <code>remove</code>
status	String	Status of updating the service appointment. Output: <code>SUCCESS</code> or <code>FAIL</code>
messageCode	Number	Success or error code of updating the service appointment. 0 means success. The error codes are listed in Error Responses on page 523.
message	String	Success or error message of updating the service appointment.
status	String	Status of the request. Output: <code>SUCCESS</code> or <code>FAIL</code>
messageCode	Number	Success or error code of the request. 0 means success. The error codes are listed in Error Responses on page 523.
message	String	Success or error message of the request.
messageAdditionalInfo	String	More information provided by Salesforce.



Example: JSON Example: Successful Response

This example shows a successful response for removing a service appointment from a bundle.

```
{
  "bundleId": "08px0000000AkUXAA1",
  "responsePayloads": [
    {
      "objectName": "ServiceAppointment",
      "objectId": "08px0000000AjPRAA1",
```

```

        "action": "remove",
        "status": "SUCCESS",
        "messageCode": "NONE",
        "message": "Success",
        "messageParams": []
      }
    ],
    "status": "SUCCESS",
    "messageCode": "NONE",
    "message": "Success"
  }
}

```



Example: JSON Example: Fail Response

This example shows the output of a request where the user entered an invalid service appointment ID.

```

{
  "bundleId": null,
  "responsePayloads": [
    {
      "status": "FAIL",
      "messageCode": "GET_BUNDLE_INFO_FAILURE",
      "message": "Try again later.",
      "messageParams": []
    }
  ],
  "status": "FAIL",
  "messageCode": "ERROR_REMOVING_FROM_BUNDLE",
  "message": "We couldn't remove the service appointment from the bundle."
  "messageAdditionalInfo": ""
}

```

Table 2: Error Responses

Code	Name	Message
0	NONE	<Service Appointment ID#>: Success.
1	UNSUPPORTED_VERSION	Ask your admin to check the API version, then try again.
2	SA_PAYLOAD_SHOULD_CONTAIN_ADD_ACTIONS_ONLY	Create bundles using add actions only.
3	EMPTY_BUNDLE_POLICY	Enter an ID for the bundlePolicyId.
4	EMPTY_BUNDLE_INITIATE	Specify manual or auto for the initiate entry.
5	EMPTY_BUNDLE_SERVICE_APPOINTMENT	Add a bundle service appointment.
6	SA_PAYLOAD_DUPLICATED_SAS	Remove duplicate service appointment IDs.
7	SA_IS_BUNDLE	<Service Appointment ID#>: Is already a bundle service appointment.

Code	Name	Message
8	SA_ALREADY_BUNDLED	<Service Appointment ID#>: Is a bundle member service appointment in another bundle.
9	STATUS_FORBIDDEN	<Service Appointment ID#>: Can't be bundled in the current status.
10	SA_PAYLOAD_SHOULD_NOT_CONTAIN_MIX_ACTIONS	Send separate requests for add and remove actions.
11	SA_IS_NOT_BUNDLE	Select a bundle service appointment and try again.
12	EXCEEDED_BUNDLE_MEMBERS_LIMIT	You reached the maximum number of service appointments for this bundle. Remove some of them and try again.
13	SA_PAYLOAD_SHOULD_CONTAIN_ONLY_BUNDLE_MEMBERS	To remove bundle members from a bundle, include only service appointments that are bundle members.
14	ERROR_IN_BASIC_VALIDATIONS	We couldn't bundle the service appointments.
15	ERROR_IN_LIMIT_VALIDATIONS	N/A
16	ERROR_IN_CRITERIA_VALIDATIONS	N/A
17	ERROR_IN_SERVICE_TERRITORY_VALIDATIONS	Confirm or revise the service territories of the selected service appointments. Or ask your admin for help.
18	ERROR_IN_RESTRICTION_VALIDATIONS	Confirm or revise the selected service appointments. Or ask your admin to check the restriction policies.
19	EXCEEDED_BUNDLE_DURATION_LIMIT	N/A
20	EMPTY_BODY	Fill in the request.
21	SA_IS_NOT_BUNDLED	<Service Appointment ID#>: Can't remove a service appointment that isn't in the bundle.
22	ERROR_IN_AGGREGATION	Confirm or revise the selected service appointments. Or ask your admin to check the aggregation policies.
23	ERROR_IN_PROPAGATION	Confirm or revise the selected service appointments. Or ask your admin to check the propagation policies.
24	GET_POLICY_FAILURE	Ask your admin to check the bundle policies, then try again.

Code	Name	Message
25	GET_CONFIG_FAILURE	Ask your admin to check the bundle config, then try again.
26	GET_BUNDLE_INFO_FAILURE	Try again later.
27	ERROR_IN_LOGIC	We couldn't bundle the service appointments.
28	EMPTY_BUNDLE_MEMBERS	Add bundle member service appointments.
29	EMPTY_BUNDLE_PARENT_WORK_ORDER_ID	Add the work order ID for the bundle service appointment.
30	UNBUNDLE_REQUEST_ASSIGNED_RESOURCES_NOT_COMPATIBLE	To unbundle a scheduled bundle, send assigned resources for each service appointment.
31	EMPTY_OLD_VALUES	To update a bundle, send the current values.
32	EMPTY_NEW_VALUES	To update a bundle, send the new values.
33	SA_PAYLOAD_SHOULD_CONTAIN_AT_LEAST_ONE_ADD_OR_REMOVE_ACTION	To update a bundle, send at least one add or remove action.
34	FORBIDDEN_FIELDS_IN_POLICY_OBJECT	Confirm or revise the selected service appointments. Or ask your admin to check the field names selected in the related policies.
35	EMPTY_SERVICE_TERRITORY	Add a Service Territory ID.
36	EMPTY_START_TIME	Add the interval's start time.
37	EMPTY_END_TIME	Add the interval's end time.
38	ERROR_IN_BUNDLE_POLICY_RFC	Confirm or revise the selected service appointments. Or ask your admin to check the recordset filter criteria in the bundle policy.
39	ERROR_IN_TIME_ZONE_VALIDATION	Select service appointments that are in the same time zone. Or ask your admin for help.
40	ERROR_IN_LOGIC_UNBUNDLE	We couldn't unbundle the service appointment.
41	ERROR_CREATING_BUNDLE	We couldn't bundle the service appointments.
42	ERROR_ADDING_TO_BUNDLE	We couldn't add the service appointment to the bundle.

Code	Name	Message
43	ERROR_REMOVING_FROM_BUNDLE	We couldn't remove the service appointment from the bundle.
44	ERROR_UNBUNDLING	We couldn't unbundle the service appointment.
58	ERROR_IN_MSW_VALIDATIONS	We can't bundle service appointments with dependencies. Remove the dependencies and try again.
-500	http error	Hmm... Something went wrong. Try again.

Start Batch

Creates service appointment bundles automatically. The resource uses appointment bundle policies that are marked for automatic bundling. Service appointments that are already bundled manually stay untouched by this API. The resource returns the success or failure message of starting the batch process of automatic bundling. This API is not supported in Gov Cloud. This endpoint is available in version 54.0 and later of the Salesforce API.

Add a Remote Site

1. From Setup, in the Quick Find box, enter *Security*, and then select **Remote Site Settings**.
2. Click **New Remote Site**.
3. Enter a name.
4. For Remote Site URL, enter `https://api.salesforce.com/`
5. Save your changes.

URI

`<host>/bundleflow/api/v1.0/startbatch`

Format

JSON

HTTP Method

POST

Authentication

Authorization: Bearer `<token>`

If the authentication token isn't valid, a 401 HTTP status is returned.

Headers

`request.setHeader('x-sfdc-tenant-id', 'core/prod/ORG-ID_18_Characters');`—Field Service functional domain

`sf_api_version`—Salesforce API version. Minimum version is 53.0.

`Authorization`—Authorization token.

`Content-Type`—File format. Valid value is `JSON`.

`Referer`—Org domain URL.

Request Body

Properties

Field	Type	Description
operation	String	The bundling operation. Valid value: start-batch-processing



Example: JSON Example

This example shows an automated bundling request of service appointments.

```
{
  "operation": "start-batch-processing"
}
```

Response Body

Properties

Field	Type	Description
httpStatus	Number	Response HTTP status.
statusDescription	String	Response status description.
responsePayload	String	Message about the batch status. If it's successful, the payload is null. If the batch bundling fails, the payload shows an error message, such as, "Wrong API version" or "Failed to start batch agent".
messageCode	Number	Message code is always 0.
messageAdditionalInfo	String	More information provided by Salesforce.



Example: JSON Example: Successful Response

This example shows the output of a batch process of automatic bundling that started successfully.

```
{
  "httpStatus": 200,
  "statusDescription": null,
  "responsePayload": null,
  "messageCode": "NONE"
}
```

**Example: JSON Example: Fail Response**

This example shows the output of a batch process that failed to start because the operation parameter of the request is `null` or empty.

```
{
  "httpStatus":400,
  "statusDescription":null,
  "responsePayload":"Failed to start batch agent",
  "messageCode":"NONE"
  "messageAdditionalInfo": ""
}
```

Unbundle

Unbundle a bundle of service appointments. This resource accepts a bundle ID. This API is not supported in Gov Cloud. This endpoint is available in version 54.0 and later of the Salesforce API.

Add a Remote Site

1. From Setup, in the Quick Find box, enter *Security*, and then select **Remote Site Settings**.
2. Click **New Remote Site**.
3. Enter a name.
4. For Remote Site URL, enter `https://api.salesforce.com/`
5. Save your changes.

URI

`<host>/bundleflow/api/v1.0/bundle/<ID>?initiate=manual`

Format

JSON

HTTP Method

DELETE

Authentication

Authorization: Bearer `<token>`

If the authentication token isn't valid, a 401 HTTP status is returned.

Headers

`request.setHeader('x-sfdc-tenant-id', 'core/prod/ORG-ID_18_Characters');`—Field Service functional domain

`sf_api_version`—Salesforce API version. Minimum version is 53.0.

`Authorization`—Authorization token.

`Content-Type`—File format. Valid value is `JSON`.

`Referer`—Org domain URL.

Parameters

Parameter	Description
ID	Unique ID of the bundle.

Response Body**Properties**

Field	Type	Description
bundleId	String	ID of the bundle. If the unbundling fails, it's <code>null</code> .
responsePayloads	Array	Response details of unbundling the service appointments. If it's successful, the payload is <code>null</code> . If the unbundling fails, the payload shows error details.
status	String	Status of unbundling the service appointments. Output: <code>FAIL</code>
messageCode	Number	Error code. The error codes are listed in Error Responses on page 530.
message	String	Error message.
status	String	Status of the request. Output: <code>SUCCESS</code> or <code>FAIL</code>
messageCode	Number	Success or error code of the request. 0 means success. The error codes are listed in Error Responses on page 530.
message	String	Success or error message of the request.
messageAdditionalInfo	String	More information provided by Salesforce.

**Example: JSON Example: Successful Response**

This example shows the output of service appointments that are unbundled successfully.

```
{
  "bundleId": "08px000000NzdMXAAZ",
  "responsePayloads": null,
  "status": "SUCCESS",
  "messageCode": "NONE",
  "message": "Success"
}
```

**Example: JSON Example: Fail Response**

This example shows the output of a request where the user entered a service appointment ID instead of a bundle ID. The request only works with bundle IDs, so the response status is `FAIL`.

```
{
  "bundleId":null,
  "responsePayloads":[
    {
      "status":"FAIL",
      "messageCode":"GET_BUNDLE_INFO_FAILURE",
      "message":"Replace the service appointment with a bundle service appointment.",
      "messageParams":[

      ]
    }
  ],
  "status":"FAIL",
  "messageCode":"ERROR_UNBUNDLING",
  "message":"We couldn't unbundle the service appointment.",
  "messageAdditionalInfo":""
}
```

Table 3: Error Responses

Code	Name	Message
0	NONE	<Service Appointment ID#>: Success.
1	UNSUPPORTED_VERSION	Ask your admin to check the API version, then try again.
2	SA_PAYLOAD_SHOULD_CONTAIN_ADD_ACTIONS_ONLY	Create bundles using add actions only.
3	EMPTY_BUNDLE_POLICY	Enter an ID for the bundlePolicyId.
4	EMPTY_BUNDLE_INITIATE	Specify manual or auto for the initiate entry.
5	EMPTY_BUNDLE_SERVICE_APPOINTMENT	Add a bundle service appointment.
6	SA_PAYLOAD_DUPLICATED_SAS	Remove duplicate service appointment IDs.
7	SA_IS_BUNDLE	<Service Appointment ID#>: Is already a bundle service appointment.
8	SA_ALREADY_BUNDLED	<Service Appointment ID#>: Is a bundle member service appointment in another bundle.
9	STATUS_FORBIDDEN	<Service Appointment ID#>: Can't be bundled in the current status.
10	SA_PAYLOAD_SHOULD_NOT_CONTAIN_MIX_ACTIONS	Send separate requests for add and remove actions.

Code	Name	Message
11	SA_IS_NOT_BUNDLE	Select a bundle service appointment and try again.
12	EXCEEDED_BUNDLE_MEMBERS_LIMIT	You reached the maximum number of service appointments for this bundle. Remove some of them and try again.
13	SA_PAYLOAD_SHOULD_CONTAIN_ONLY_BUNDLE_MEMBERS	To remove bundle members from a bundle, include only service appointments that are bundle members.
14	ERROR_IN_BASIC_VALIDATIONS	We couldn't bundle the service appointments.
15	ERROR_IN_LIMIT_VALIDATIONS	N/A
16	ERROR_IN_CRITERIA_VALIDATIONS	N/A
17	ERROR_IN_SERVICE_TERRITORY_VALIDATIONS	Confirm or revise the service territories of the selected service appointments. Or ask your admin for help.
18	ERROR_IN_RESTRICTION_VALIDATIONS	Confirm or revise the selected service appointments. Or ask your admin to check the restriction policies.
19	EXCEEDED_BUNDLE_DURATION_LIMIT	N/A
20	EMPTY_BODY	Fill in the request.
21	SA_IS_NOT_BUNDLED	<Service Appointment ID#>: Can't remove a service appointment that isn't in the bundle.
22	ERROR_IN_AGGREGATION	Confirm or revise the selected service appointments. Or ask your admin to check the aggregation policies.
23	ERROR_IN_PROPAGATION	Confirm or revise the selected service appointments. Or ask your admin to check the propagation policies.
24	GET_POLICY_FAILURE	Ask your admin to check the bundle policies, then try again.
25	GET_CONFIG_FAILURE	Ask your admin to check the bundle config, then try again.
26	GET_BUNDLE_INFO_FAILURE	Try again later.
27	ERROR_IN_LOGIC	We couldn't bundle the service appointments.

Code	Name	Message
28	EMPTY_BUNDLE_MEMBERS	Add bundle member service appointments.
29	EMPTY_BUNDLE_PARENT_WORK_ORDER_ID	Add the work order ID for the bundle service appointment.
30	UNBUNDLE_REQUEST_ASSIGNED_RESOURCES_NOT_COMPATIBLE	To unbundle a scheduled bundle, send assigned resources for each service appointment.
31	EMPTY_OLD_VALUES	To update a bundle, send the current values.
32	EMPTY_NEW_VALUES	To update a bundle, send the new values.
33	SA_PAYLOAD_SHOULD_CONTAIN_AT_LEAST_ONE_ADD_OR_REMOVE_ACTION	To update a bundle, send at least one add or remove action.
34	FORBIDDEN_FIELDS_IN_POLICY_OBJECT	Confirm or revise the selected service appointments. Or ask your admin to check the field names selected in the related policies.
35	EMPTY_SERVICE_TERRITORY	Add a Service Territory ID.
36	EMPTY_START_TIME	Add the interval's start time.
37	EMPTY_END_TIME	Add the interval's end time.
38	ERROR_IN_BUNDLE_POLICY_RFC	Confirm or revise the selected service appointments. Or ask your admin to check the recordset filter criteria in the bundle policy.
39	ERROR_IN_TIME_ZONE_VALIDATION	Select service appointments that are in the same time zone. Or ask your admin for help.
40	ERROR_IN_LOGIC_UNBUNDLE	We couldn't unbundle the service appointment.
41	ERROR_CREATING_BUNDLE	We couldn't bundle the service appointments.
42	ERROR_ADDING_TO_BUNDLE	We couldn't add the service appointment to the bundle.
43	ERROR_REMOVING_FROM_BUNDLE	We couldn't remove the service appointment from the bundle.
44	ERROR_UNBUNDLING	We couldn't unbundle the service appointment.

Code	Name	Message
58	ERROR_IN_MSW_VALIDATIONS	We can't bundle service appointments with dependencies. Remove the dependencies and try again.
-500	http error	Hmm... Something went wrong. Try again.

Unbundle Multiple

Unbundle one or more bundles of service appointments. This resource accepts one or more bundle IDs. This API is not supported in Gov Cloud. This endpoint is available in version 54.0 and later of the Salesforce API.

Add a Remote Site

1. From Setup, in the Quick Find box, enter *Security*, and then select **Remote Site Settings**.
2. Click **New Remote Site**.
3. Enter a name.
4. For Remote Site URL, enter `https://api.salesforce.com/`
5. Save your changes.

URI

`<host>/bundleflow/api/v1.0/unbundleMultiple`

Format

JSON

HTTP Method

POST

Authentication

Authorization: Bearer `<token>`

If the authentication token isn't valid, a 401 HTTP status is returned.

Headers

`request.setHeader('x-sfdc-tenant-id', 'core/prod/ORG-ID_18_Characters');`—Field Service functional domain

`sf_api_version`—Salesforce API version. Minimum version is 53.0.

`Authorization`—Authorization token.

`Content-Type`—File format. Valid value is JSON.

`Referer`—Org domain URL.

Request Body

Properties

Field	Type	Description
user	String	Optional. User name of user sending the request.
initiate	String	Type of bundle request. Valid value: <code>manual</code>
saRequestPayloads	Array	IDs of bundle service appointment to unbundle.
serviceAppointmentId	String	ID of a bundle service appointment.



Example: JSON Example

This example shows a request to unbundle two bundles.

```
{
  "user": "Misha",
  "initiate": "manual",
  "saRequestPayloads": [
    {
      "serviceAppointmentId": "08px000000NzdLFAAZ"
    },
    {
      "serviceAppointmentId": "08px000000NzdH8AAJ"
    }
  ]
}
```

Response Body

Properties

The response body is an array of objects with these fields. Each object contains response details for a bundle service appointment you requested to unbundle.

Field	Type	Description
headers	Object	For future use.
body	Object	Details of unbundling the service appointment bundle.
bundle id	String	ID of the bundle that is unbundled. If the unbundling fails, it's <code>null</code> .
responsePayloads	Array	Response details of unbundling the service appointments. If it's successful, the payload is <code>null</code> . If the unbundling fails, the payload shows error details.
status	String	Failure status of unbundling the service appointments. Output: <code>FAIL</code>
messageCode	Number	Error code. The error codes are listed in Error Responses on page 536.

Field	Type	Description
message	String	Error message.
status	String	Status of unbundling the service appointments. Output: SUCCESS or FAIL
messageCode	Number	Success or error code of unbundling the bundle. 0 means success. The error codes are listed in Error Responses on page 536.
message	String	Success or error message of unbundling the bundle.
statusCode	String	Success or error code of the request.
statusCodeValue	Number	Success or error code number of the request. 0 means success. The error codes are listed in Error Responses on page 536.

Example: JSON Example: Successful Response

This example shows a successful response of unbundling two service appointment bundles.

```
{
  "httpStatus": 200,
  "statusDescription": "OK",
  "responsePayload": "[ {\n  \"headers\" : { },\n  \"body\" : {\n    \"bundleId\" : \"08px000000Aph4AAD\",\n    \"responsePayloads\" : null,\n    \"status\" : \"SUCCESS\",\n    \"messageCode\" : \"NONE\",\n    \"message\" : \"Success\"\n  },\n  \"statusCode\" : \"OK\",\n  \"statusCodeValue\" : 200\n} ]",
  "messageCode": "NONE"
}
```

Example: JSON Example: Fail Response

This example shows the output of a request where the user entered a service appointment ID instead of a bundle ID. The request only works with bundle IDs, so the response status is **FAIL**.

```
[
  {
    "httpStatus": 200,
    "statusDescription": "OK",
    "responsePayload": "[ {\n  \"headers\" : { },\n  \"body\" : {\n    \"bundleId\" : null,\n    \"responsePayloads\" : [ {\n      \"status\" : \"FAIL\",\n      \"messageCode\" : \"GET_BUNDLE_INFO_FAILURE\",\n      \"message\" : \"Try again later.\",\n      \"messageParams\" : [ ]\n    },\n    \"statusCode\" : \"FAIL\",\n    \"messageCode\" : \"ERROR_UNBUNDLING\",\n    \"message\" : \"We couldn't unbundle the service appointment.\",\n    \"messageAdditionalInfo\" : \"(bundleSaRef)\\n errorCode: NOT_FOUND\\n message: Provided external ID field does not exist or is not accessible: 123\"\n  },\n  \"statusCode\" : \"OK\",\n  \"statusCodeValue\" : 200\n} ]",
    "messageCode": "NONE",
    "messageAdditionalInfo": "(bundleSaRef)\n errorCode: NOT_FOUND\n message: Provided external ID field does not exist or is not accessible: 123"
  }
]
```

Table 4: Error Responses

Code	Name	Message
0	NONE	<Service Appointment ID#>: Success.
1	UNSUPPORTED_VERSION	Ask your admin to check the API version, then try again.
2	SA_PAYLOAD_SHOULD_CONTAIN_ADD_ACTIONS_ONLY	Create bundles using add actions only.
3	EMPTY_BUNDLE_POLICY	Enter an ID for the bundlePolicyId.
4	EMPTY_BUNDLE_INITIATE	Specify manual or auto for the initiate entry.
5	EMPTY_BUNDLE_SERVICE_APPOINTMENT	Add a bundle service appointment.
6	SA_PAYLOAD_DUPLICATED_SAS	Remove duplicate service appointment IDs.
7	SA_IS_BUNDLE	<Service Appointment ID#>: Is already a bundle service appointment.
8	SA_ALREADY_BUNDLED	<Service Appointment ID#>: Is a bundle member service appointment in another bundle.
9	STATUS_FORBIDDEN	<Service Appointment ID#>: Can't be bundled in the current status.
10	SA_PAYLOAD_SHOULD_NOT_CONTAIN_MIX_ACTIONS	Send separate requests for add and remove actions.
11	SA_IS_NOT_BUNDLE	Select a bundle service appointment and try again.
12	EXCEEDED_BUNDLE_MEMBERS_LIMIT	You reached the maximum number of service appointments for this bundle. Remove some of them and try again.
13	SA_PAYLOAD_SHOULD_CONTAIN_ONLY_BUNDLE_MEMBERS	To remove bundle members from a bundle, include only service appointments that are bundle members.
14	ERROR_IN_BASIC_VALIDATIONS	We couldn't bundle the service appointments.
15	ERROR_IN_LIMIT_VALIDATIONS	N/A
16	ERROR_IN_CRITERIA_VALIDATIONS	N/A
17	ERROR_IN_SERVICE_TERRITORY_VALIDATIONS	Confirm or revise the service territories of the selected service appointments. Or ask your admin for help.

Code	Name	Message
18	ERROR_IN_RESTRICTION_VALIDATIONS	Confirm or revise the selected service appointments. Or ask your admin to check the restriction policies.
19	EXCEEDED_BUNDLE_DURATION_LIMIT	N/A
20	EMPTY_BODY	Fill in the request.
21	SA_IS_NOT_BUNDLED	<Service Appointment ID#>: Can't remove a service appointment that isn't in the bundle.
22	ERROR_IN_AGGREGATION	Confirm or revise the selected service appointments. Or ask your admin to check the aggregation policies.
23	ERROR_IN_PROPAGATION	Confirm or revise the selected service appointments. Or ask your admin to check the propagation policies.
24	GET_POLICY_FAILURE	Ask your admin to check the bundle policies, then try again.
25	GET_CONFIG_FAILURE	Ask your admin to check the bundle config, then try again.
26	GET_BUNDLE_INFO_FAILURE	Try again later.
27	ERROR_IN_LOGIC	We couldn't bundle the service appointments.
28	EMPTY_BUNDLE_MEMBERS	Add bundle member service appointments.
29	EMPTY_BUNDLE_PARENT_WORK_ORDER_ID	Add the work order ID for the bundle service appointment.
30	UNBUNDLE_REQUEST_ASSIGNED_RESOURCES_NOT_COMPATIBLE	To unbundle a scheduled bundle, send assigned resources for each service appointment.
31	EMPTY_OLD_VALUES	To update a bundle, send the current values.
32	EMPTY_NEW_VALUES	To update a bundle, send the new values.
33	SA_PAYLOAD_SHOULD_CONTAIN_AT_LEAST_ONE_ADD_OR_REMOVE_ACTION	To update a bundle, send at least one add or remove action.
34	FORBIDDEN_FIELDS_IN_POLICY_OBJECT	Confirm or revise the selected service appointments. Or ask your admin to check the field names selected in the related policies.

Code	Name	Message
35	EMPTY_SERVICE_TERRITORY	Add a Service Territory ID.
36	EMPTY_START_TIME	Add the interval's start time.
37	EMPTY_END_TIME	Add the interval's end time.
38	ERROR_IN_BUNDLE_POLICY_RFC	Confirm or revise the selected service appointments. Or ask your admin to check the recordset filter criteria in the bundle policy.
39	ERROR_IN_TIME_ZONE_VALIDATION	Select service appointments that are in the same time zone. Or ask your admin for help.
40	ERROR_IN_LOGIC_UNBUNDLE	We couldn't unbundle the service appointment.
41	ERROR_CREATING_BUNDLE	We couldn't bundle the service appointments.
42	ERROR_ADDING_TO_BUNDLE	We couldn't add the service appointment to the bundle.
43	ERROR_REMOVING_FROM_BUNDLE	We couldn't remove the service appointment from the bundle.
44	ERROR_UNBUNDLING	We couldn't unbundle the service appointment.
58	ERROR_IN_MSW_VALIDATIONS	We can't bundle service appointments with dependencies. Remove the dependencies and try again.
-500	http error	Hmm... Something went wrong. Try again.

Update Bundle

Adds service appointments to an existing bundle. This resource accepts a bundle ID and service appointment IDs. This API is not supported in Gov Cloud. This endpoint is available in version 54.0 and later of the Salesforce API.

Add a Remote Site

1. From Setup, in the Quick Find box, enter *Security*, and then select **Remote Site Settings**.
2. Click **New Remote Site**.
3. Enter a name.
4. For Remote Site URL, enter `https://api.salesforce.com/`
5. Save your changes.

URI

`<host>/bundleflow/api/v1.0/bundle/{ID}`

Format

JSON

HTTP Method

PATCH

Authentication

Authorization: Bearer `<token>`

If the authentication token isn't valid, a 401 HTTP status is returned.

Headers

`request.setHeader('x-sfdc-tenant-id', 'core/prod/ORG-ID_18_Characters');`—Field Service functional domain

`sf_api_version`—Salesforce API version. Minimum version is 53.0.

`Authorization`—Authorization token.

`Content-Type`—File format. Valid value is `JSON`.

`Referer`—Org domain URL.

Parameters

Parameter	Description
ID	Unique ID of the bundle.

Request Body**Properties**

Field	Type	Description
user	String	Optional. User name of user sending the request.
initiate	String	Type of bundle request. Valid value: <code>manual</code>
saRequestPayloads	Array	Details of the service appointments to be added to this bundle.
serviceAppointmentId	String	ID of a service appointment.
action	String	Bundling action performed on this service appointment. Valid values: <code>add</code>

**Example: JSON Example**

This example shows a manual request to add two service appointments to a bundle.

```
{
  "user": "Misha",
```

```

    "initiate": "manual",
    "saRequestPayloads": [
      {
        "serviceAppointmentId": "08px000000NzbnkAAB",
        "action": "add"
      },
      {
        "serviceAppointmentId": "08pT300000006LLIAY",
        "action": "add"
      }
    ]
  }
}

```

Response Body

Properties

Field	Type	Description
bundleId	String	ID of the bundle to update. If the update isn't successful, it's null.
responsePayloads	Array	Details of the service appointment objects that are part of the bundle request.
objectName	String	Type of object. Output: <code>ServiceAppointment</code>
objectId	String	ID of the service appointment.
action	String	Bundling action performed on this service appointment. Valid values: <code>add</code>
status	String	Status of updating the service appointment. Output: <code>SUCCESS</code> or <code>FAIL</code>
messageCode	Number	Success or error code of updating the service appointment. 0 means success. The error codes are listed in Error Responses on page 541.
message	String	Success or error message of updating the service appointment.
status	String	Status of the request. Output: <code>SUCCESS</code> or <code>FAIL</code>
messageCode	Number	Success or error code of the request. 0 means success. The error codes are listed in Error Responses on page 541.
message	String	Success or error message of the request.
messageAdditionalInfo	String	More information provided by Salesforce.

Example: JSON Example: Successful Response

This example shows a successful response of adding a service appointment to a bundle.

```
{
  "bundleId": "08px0000000AkUXAA1",
  "responsePayloads": [
    {
      "objectName": "ServiceAppointment",
      "objectId": "08px0000000AjPRAA1",
      "action": "add",
      "status": "SUCCESS",
      "messageCode": "NONE",
      "message": "Success",
      "messageParams": []
    }
  ],
  "status": "SUCCESS",
  "messageCode": "NONE",
  "message": "Success"
}
```

Example: JSON Example: Fail Response

This example shows the output of a request where the user entered an invalid service appointment ID.

```
{
  "bundleId": *null*,
  "responsePayloads": [
    {
      "status": "FAIL",
      "messageCode": "GET_BUNDLE_INFO_FAILURE",
      "message": "Try again later.",
      "messageParams": []
    }
  ],
  "status": "FAIL",
  "messageCode": "ERROR_ADDING_TO_BUNDLE",
  "message": "We couldn't add the service appointment to the bundle.",
  "messageAdditionalInfo": ""
}
```

Table 5: Error Responses

Code	Name	Message
0	NONE	<Service Appointment ID#>: Success.
1	UNSUPPORTED_VERSION	Ask your admin to check the API version, then try again.
2	SA_PAYLOAD_SHOULD_CONTAIN_ADD_ACTIONS_ONLY	Create bundles using add actions only.
3	EMPTY_BUNDLE_POLICY	Enter an ID for the bundlePolicyId.

Code	Name	Message
4	EMPTY_BUNDLE_INITIATE	Specify manual or auto for the initiate entry.
5	EMPTY_BUNDLE_SERVICE_APPOINTMENT	Add a bundle service appointment.
6	SA_PAYLOAD_DUPLICATED_SAS	Remove duplicate service appointment IDs.
7	SA_IS_BUNDLE	<Service Appointment ID#>: Is already a bundle service appointment.
8	SA_ALREADY_BUNDLED	<Service Appointment ID#>: Is a bundle member service appointment in another bundle.
9	STATUS_FORBIDDEN	<Service Appointment ID#>: Can't be bundled in the current status.
10	SA_PAYLOAD_SHOULD_NOT_CONTAIN_MIX_ACTIONS	Send separate requests for add and remove actions.
11	SA_IS_NOT_BUNDLE	Select a bundle service appointment and try again.
12	EXCEEDED_BUNDLE_MEMBERS_LIMIT	You reached the maximum number of service appointments for this bundle. Remove some of them and try again.
13	SA_PAYLOAD_SHOULD_CONTAIN_ONLY_BUNDLE_MEMBERS	To remove bundle members from a bundle, include only service appointments that are bundle members.
14	ERROR_IN_BASIC_VALIDATIONS	We couldn't bundle the service appointments.
15	ERROR_IN_LIMIT_VALIDATIONS	N/A
16	ERROR_IN_CRITERIA_VALIDATIONS	N/A
17	ERROR_IN_SERVICE_TERRITORY_VALIDATIONS	Confirm or revise the service territories of the selected service appointments. Or ask your admin for help.
18	ERROR_IN_RESTRICTION_VALIDATIONS	Confirm or revise the selected service appointments. Or ask your admin to check the restriction policies.
19	EXCEEDED_BUNDLE_DURATION_LIMIT	N/A
20	EMPTY_BODY	Fill in the request.
21	SA_IS_NOT_BUNDLED	<Service Appointment ID#>: Can't remove a service appointment that isn't in the bundle.

Code	Name	Message
22	ERROR_IN_AGGREGATION	Confirm or revise the selected service appointments. Or ask your admin to check the aggregation policies.
23	ERROR_IN_PROPAGATION	Confirm or revise the selected service appointments. Or ask your admin to check the propagation policies.
24	GET_POLICY_FAILURE	Ask your admin to check the bundle policies, then try again.
25	GET_CONFIG_FAILURE	Ask your admin to check the bundle config, then try again.
26	GET_BUNDLE_INFO_FAILURE	Try again later.
27	ERROR_IN_LOGIC	We couldn't bundle the service appointments.
28	EMPTY_BUNDLE_MEMBERS	Add bundle member service appointments.
29	EMPTY_BUNDLE_PARENT_WORK_ORDER_ID	Add the work order ID for the bundle service appointment.
30	UNBUNDLE_REQUEST_ASSIGNED_RESOURCES_NOT_COMPATIBLE	To unbundle a scheduled bundle, send assigned resources for each service appointment.
31	EMPTY_OLD_VALUES	To update a bundle, send the current values.
32	EMPTY_NEW_VALUES	To update a bundle, send the new values.
33	SA_PAYLOAD_SHOULD_CONTAIN_AT_LEAST_ONE_ADD_OR_REMOVE_ACTION	To update a bundle, send at least one add or remove action.
34	FORBIDDEN_FIELDS_IN_POLICY_OBJECT	Confirm or revise the selected service appointments. Or ask your admin to check the field names selected in the related policies.
35	EMPTY_SERVICE_TERRITORY	Add a Service Territory ID.
36	EMPTY_START_TIME	Add the interval's start time.
37	EMPTY_END_TIME	Add the interval's end time.
38	ERROR_IN_BUNDLE_POLICY_RFC	Confirm or revise the selected service appointments. Or ask your admin to check the recordset filter criteria in the bundle policy.

Code	Name	Message
39	ERROR_IN_TIME_ZONE_VALIDATION	Select service appointments that are in the same time zone. Or ask your admin for help.
40	ERROR_IN_LOGIC_UNBUNDLE	We couldn't unbundle the service appointment.
41	ERROR_CREATING_BUNDLE	We couldn't bundle the service appointments.
42	ERROR_ADDING_TO_BUNDLE	We couldn't add the service appointment to the bundle.
43	ERROR_REMOVING_FROM_BUNDLE	We couldn't remove the service appointment from the bundle.
44	ERROR_UNBUNDLING	We couldn't unbundle the service appointment.
58	ERROR_IN_MSW_VALIDATIONS	We can't bundle service appointments with dependencies. Remove the dependencies and try again.
-500	http error	Hmm... Something went wrong. Try again.

Field Service Metadata API

The following Metadata types are used in Field Service.

IN THIS SECTION:

[FieldServiceSettings](#)

Represents an organization's Field Service settings.

[Skill](#)

Represents the settings for a skill used for field service or to route chats to agents in Chat, such as the name of the skill and which agents the skills are assigned to.

[TimeSheetTemplate](#)

Represents a template for creating time sheets in Field Service. This type extends the Metadata metadata type and inherits its `fullName` field.

FieldServiceSettings

Represents an organization's Field Service settings.

To learn more about Field Service settings, see [Enable Field Service](#) in Salesforce Help.

In the package manifest, all organization settings metadata types are accessed using the Settings name. See Settings for more details.

Version

FieldServiceSettings is available in API version 40.0 and later.

Fields

Field	Field Type	Description
apptAssistantExpiration	int	The expiration time of <code>apptAssistantInfoUrl</code> after which the customer stops seeing the mobile worker's location and estimated time of arrival. Available in API version 50.0 and later.
apptAssistantInfoUrl	string	The tracking URL that helps a customer see the mobile worker's estimated time of arrival and tracking information. Available in API version 50.0 and later.
apptAssistantRadiusUnitValue	ApptAssistantRadiusUnit (enumeration of type string)	The units for specifying the geofence radius. When the mobile worker enters this area, a Last Mile notification is automatically sent. Valid values are: • Kilometer • Meter • Mile • Yard Available in API version 50.0 and later.
apptAssistantRadiusValue	int	The geofence radius from the service appointment's address used for sending a notification to the customer when the mobile worker approaches the address. Available in API version 50.0 and later.
apptAssistantStatus	string	The status on the service appointment used to trigger En Route notification. The value has to match one of the service appointment's Status field options. When the mobile worker selects this status on a service appointment, the customer receives the En Route notification with tracking details. Available in API version 50.0 and later.
canceledDefaultStatus	string	The default status value when a service appointment is canceled. Available in API version 65.0 and later.

Field	Field Type	Description
<code>cannotCompleteDefaultStatus</code>	string	The default status value when a service appointment can't be completed. Available in API version 65.0 and later.
<code>canPopulateGoogleAddress</code>	boolean	Allows desktop and mobile to send geolocation and map data to Google and Apple. Available in API version 57.0 and later.
<code>canSendAppCenterCrashReports</code>	boolean	Allows Salesforce to send crash reports to Microsoft App Center. Available in API version 57.0 and later.
<code>canStoreMobileAnalytics</code>	boolean	Allows third parties to store mobile analytics. Available in API version 57.0 and later.
<code>completedDefaultStatus</code>	string	The default status value when a service appointment is completed. Available in API version 65.0 and later.
<code>deepLinkPublicSecurityKey</code>	string	Provides a public security key for users accessing a deep link action in the Field Service mobile app. Hide the redirection warning by configuring the deep link URL with a security key. The deep link URL then processes the security check. Available in API version 54.0 and later.
<code>dispatchedDefaultStatus</code>	string	The default status value when a service appointment is dispatched. Available in API version 65.0 and later.
<code>doesAllowEditSaForCrew</code>	boolean	Lets service crew members edit their service appointments. This setting applies only if <code>doesShareSaWithAr</code> is selected. For assigned resources of type Crew, crew members get Read-Write access to their service appointment and, if <code>doesShareSaParentWoWithAr</code> is selected, to their service appointments' parent work orders.
<code>doesShareSaParentWoWithAr</code>	boolean	Shares service appointments' parent work orders with their assigned resources. This setting applies only if <code>doesShareSaWithAr</code> is selected and sharing access for work orders is set to Private or Public Read Only. Technician assigned resources get Read-Write access to their work orders. For assigned resources of type Crew,

Field	Field Type	Description
		the crew leader gets Read-Write access and crew members get Read access. If the service appointment's parent is a work order line item, assigned resources get access to the associated work order.
doesShareSaWithAr	boolean	Shares dispatched service appointments with their assigned resources. This setting applies only if sharing access for service appointments is set to Private or Public Read Only. Technician assigned resources get Read-Write access to their service appointments. For assigned resources of type Crew, the crew leader gets Read-Write access and crew members get Read access.
enableDocumentBuilder	boolean	Enables access to Document Builder feature.
enableFloatingWorkOrder	boolean	Enables floating work orders for the org. Allows users to create work orders with a floating recurrence cadence based on the previous work order's completion.
enableLsdkMode	boolean	Enables Lightning SDK in the Field Service mobile app.
enablePopulateWorkOrderAddress	boolean	Allows the address to be populated when work orders are generated from a maintenance plan.
enableStandbyMode	boolean	Turns on standby mode in the Field Service mobile app. When enabled, it helps mobile workers conserve battery and reduce distractions by pausing notifications and disabling data syncs when not actively working.
enableWorkOrders	boolean	Enables Work Orders for the org. This setting allows users to use the Work Order object, whether or not Field Service is enabled. When Field Service is enabled, you can't turn off Work Orders.
enableWorkPlansAutoGeneration	boolean	Allows work plans and their work steps to be generated automatically when a work order or a work order line item is newly created. The specific work plans and work steps to be generated depends on matching criteria

Field	Field Type	Description
		specified in Work Plan Selection Rules. Available in API version 52.0 and later.
enableWorkStepManualStatusUpdate	boolean	Allows a work step status to be updated manually. A prompt suggests a status update that users can accept or defer.
fieldServiceNotificationsOrgPref	boolean	Turns on in-app notifications for the Salesforce mobile app and Lightning Experience users. Notifications are sent when any of the following actions occurs on a work order or work order line item that they own or follow: • A text or file post is added • A tracked field is updated • The record owner changes • The resource assignments change on a related service appointment If the option to track all related objects is selected in the feed tracking settings for work orders, users are also notified when child records of work orders—such as service appointments—are created or deleted.
fieldServiceOrgPref	boolean	Indicates whether Field Service is enabled.
inProgressDefaultStatus	string	The default status value when a service appointment is in progress. Available in API version 65.0 and later.
isGeoCodeSyncEnabled	boolean	Syncs the location of a Service Resource to an Inventory object.
isLocationHistoryEnabled	boolean	Tracks the location history of a Service Resource.
mobileFeedbackEmails	string	Stores an email address to which a feedback email is sent when users leave feedback from the Field Service mobile app. Available in API version 54.0 and later.
noneDefaultStatus	string	The default status value when a service appointment has no specific status. Available in API version 65.0 and later.
o2EngineEnabled	boolean	Enables Field Service Enhanced Scheduling and Optimization. The default value is false. Available in API version 55.0 and later.
objectMappingItem	ObjectMappingItem on page 549	Represents an organization's custom field mapping for Work Plan or Work Step

Field	Field Type	Description
		generation. Custom Fields can be mapped from WorkPlanTemplate to WorkPlan, WorkStepTemplate to WorkStep, or WorkPlanTemplateEntry to WorkStep. Available in API version 52.0 and later.
optimizationServiceAccess	boolean	Allows the optimization service to access data in your Salesforce org.
overrideDefaultLwcStyling	boolean	Allows the Field Service mobile app to override the default Lightning Web Component (LWC) styling with custom or branded styles. When enabled, it applies custom styling configurations (such as brand colors, themes, and visual elements) to LWCs within the mobile application.
scheduledDefaultStatus	string	The default status value when a service appointment is scheduled. Available in API version 65.0 and later.
serviceAppointmentsDueDateOffsetOrgValue	int	Indicates the number of days past the Created Date that the Due Date on auto-created service appointments should fall. Work types include an option to automatically add a service appointment to new work orders or work order line items using the work type.
workOrderDurationSource	WorkOrderDurationSource (enumeration of type string)	The source for the work order duration value. Possible values are: • WorkType • TotalFromWorkPlan • Custom Available in API version 55.0 and later.
workOrderLineItemSearchFields	string	The work order line item fields that the search engine should scan to suggest knowledge articles on work order line items.
workOrderSearchFields	string	The work order fields that the search engine should scan to suggest knowledge articles on work orders.

ObjectMappingItem

Represents an organization's custom field mapping for Work Plan or Work Step generation. Custom Fields can be mapped from WorkPlanTemplate to WorkPlan, WorkStepTemplate to WorkStep, or WorkPlanTemplateEntry to WorkStep. Available in API version 52.0 and later.

Field Name	Field Type	Description
mappingType	string	The type of object mapping. Valid values are: - WorkPlans_WorkPlanTemplate_WorkPlan — Maps a WorkPlanTemplate to a WorkPlan - WorkPlans_WorkStepTemplate_WorkStep — Maps a WorkStepTemplate to a WorkStep - WorkPlans_WorkPlanTemplateEntry_WorkStep — Maps a WorkPlanTemplateEntry to a WorkStep
objectMapping	ObjectMapping on page 550	The object mapping details.

ObjectMapping

Represents a map of fields in the input object to fields in the output object.

Field Name	Field Type	Description
inputObject	string	Required. The name of the object type containing the source fields for mapping. Valid values are: - WorkPlanTemplate - WorkStepTemplate - WorkPlanTemplateEntry
mappingFields	ObjectMappingField on page 550	Required. The mapping of source object fields to target object fields.
outputObject	string	Required. The name of the object type that receives data from the source fields. Valid values are: - WorkPlan - WorkStep

ObjectMappingField

A field name in the input object and the corresponding field name in the output object.

Field Name	Field Type	Description
inputField	string	Required. The name of a custom field supplying source data. This field is from the object specified in <code>inputObject</code> .

Field Name	Field Type	Description
outputField	string	Required. The name of a custom field that receives data from the source field specified in <code>inputField</code> . This field is from the object specified in <code>outputObject</code> .

Declarative Metadata Sample Definition

This sample file shows a subset of the possible field service settings that you can customize.

```
<?xml version="1.0" encoding="UTF-8"?>
<FieldServiceSettings xmlns="http://soap.sforce.com/2006/04/metadata">
  <doesAllowEditSaForCrew>false</doesAllowEditSaForCrew>
  <doesShareSaParentWoWithAr>false</doesShareSaParentWoWithAr>
  <doesShareSaWithAr>false</doesShareSaWithAr>
  <enableWorkOrders>false</enableWorkOrders>
  <enableWorkPlansAutoGeneration>true</enableWorkPlansAutoGeneration>
  <fieldServiceNotificationsOrgPref>false</fieldServiceNotificationsOrgPref>
  <fieldServiceOrgPref>true</fieldServiceOrgPref>
  <isGeoCodeSyncEnabled>false</isGeoCodeSyncEnabled>
  <isLocationHistoryEnabled>false</isLocationHistoryEnabled>
  <o2EngineEnabled>false</o2EngineEnabled>
  <objectMappingItem>
    <mappingType>WorkPlans_WorkPlanTemplate_WorkPlan</mappingType>
    <objectMapping>
      <inputObject>WorkPlanTemplate</inputObject>
      <mappingFields>
        <inputField>WorkPlanTemplate_CustomNumberField__c</inputField>
        <outputField>WorkPlan_CustomNumberField__c</outputField>
      </mappingFields>
      <mappingFields>
        <inputField>WorkPlanTemplate_CustomTextField__c</inputField>
        <outputField>WorkPlan_CustomPicklistField__c</outputField>
      </mappingFields>
      <outputObject>WorkPlan</outputObject>
    </objectMapping>
  </objectMappingItem>
  <objectMappingItem>
    <mappingType>WorkPlans_WorkStepTemplate_WorkStep</mappingType>
    <objectMapping>
      <inputObject>WorkStepTemplate</inputObject>
      <mappingFields>
        <inputField>WokStepTemplate_CustomNumberField__c</inputField>
        <outputField>WokStep_CustomNumberField__c</outputField>
      </mappingFields>
      <mappingFields>
        <inputField>WokStepTemplate_CustomTextField__c</inputField>
        <outputField>WokStep_CustomTextField__c</outputField>
      </mappingFields>
      <outputObject>WorkStep</outputObject>
    </objectMapping>
  </objectMappingItem>
  <objectMappingItem>
```

```

<mappingType>WorkPlans_WorkPlanTemplateEntry_WorkStep</mappingType>
<objectMapping>
  <inputObject>WorkPlanTemplateEntry</inputObject>
  <mappingFields>
    <inputField>WorkPlanTemplateEntry_CustomDateField__c</inputField>
    <outputField>WokStep_CustomDateField__c</outputField>
  </mappingFields>
  <outputObject>WorkStep</outputObject>
</objectMapping>
</objectMappingItem>
<optimizationServiceAccess>false</optimizationServiceAccess>
<serviceAppointmentsDueDateOffsetOrgValue>7</serviceAppointmentsDueDateOffsetOrgValue>

  <workOrderLineItemSearchFields>Subject</workOrderLineItemSearchFields>
  <workOrderSearchFields>Subject</workOrderSearchFields>
</FieldServiceSettings>

```

Wildcard Support in the Manifest File

The wildcard character * (asterisk) in the `package.xml` manifest file doesn't apply to metadata types for feature settings. The wildcard applies only when retrieving all settings, not for an individual setting. For details, see [Settings](#). For information about using the manifest file, see [Deploying and Retrieving Metadata with the Zip File](#).

Skill

Represents the settings for a skill used for field service or to route chats to agents in Chat, such as the name of the skill and which agents the skills are assigned to.

This type extends the Metadata metadata type and inherits its `fullName` field.

File Suffix and Directory Location

Skill values are stored in the `<developer_name>.skill` file in the `skills` directory.

Version

Skill is available in API version 28.0 and later.

Fields

Field Name	Field Type	Description
assignments	SkillAssignments	Specifies how skills are assigned to Chat users. Skills can be assigned to sets of users or sets of profiles.
description	string	Specifies the description of the skill. This field is available in API version 38.0 and later.
label	string	Specifies the name of the skill.

Field Name	Field Type	Description
<code>skillType</code>	string	Specifies the skill type, such as language or department, associated with the skill. This field is available in API version 58.0 and later.

SkillAssignments

Represents which users and user profiles to whom specific skills are assigned.

Fields

Field Name	Field Type	Description
<code>profiles</code>	SkillProfileAssignments	Specifies the profiles that are associated with a specific skill.
<code>users</code>	SkillUserAssignments	Specifies the users that are associated with a specific skill.

SkillProfileAssignments

Represents the profiles that are associated with a specific skill.

Fields

Field Name	Field Type	Description
<code>profile</code>	string	Specifies the custom name of the profile associated with a specific skill.

SkillUserAssignments

Represents the users that are associated with a specific skill.

Fields

Field Name	Field Type	Description
<code>user</code>	string	Specifies the username of the user associated with a specific skill.

Declarative Metadata Sample Definition

This is a sample of a `skill` file.

```
<?xml version="1.0" encoding="UTF-8"?>
<Skill xmlns="http://soap.sforce.com/2006/04/metadata">
```

```

<label>My Skill 1</label>
<assignments>
  <profiles>
    <profile>LiveAgentOperator</profile>
    <profile>LiveAgentSupervisor</profile>
  </profiles>
  <users>
    <user>jdoe@acme.com</user>
  </users>
</assignments>
</Skill>

```

Wildcard Support in the Manifest File

This metadata type supports the wildcard character * (asterisk) in the `package.xml` manifest file. For information about using the manifest file, see [Deploying and Retrieving Metadata with the Zip File](#).

TimeSheetTemplate

Represents a template for creating time sheets in Field Service. This type extends the Metadata metadata type and inherits its `fullName` field.



Important: Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

File Suffix and Directory Location

TimeSheetTemplate components have the suffix `timeSheetTemplate` and are stored in the `timeSheetTemplates` folder.

Version

TimeSheetTemplate components are available in API version 46.0 and later.

Special Access Rules

Field Service must be enabled. Users must have the Customize Application and Time Sheet Template permissions.

Fields

Field Name	Field Type	Description
<code>active</code>	boolean	Required. Indicates whether the time sheet template is active (<code>true</code>) or not (<code>false</code>).
<code>description</code>	string	The time sheet template's description.
<code>frequency</code>	TimeSheetFrequency (enumeration of type string)	Required. Defines the frequency of the time sheet creation period. One of the following values: Daily

Field Name	Field Type	Description
		- Weekly - EveryTwoWeeks - TwiceAMonth - Monthly
masterLabel	string	Required. The name of the time sheet template.
startDate	date	Required. The date when the time sheet takes effect.
timeSheetTemplateAssignments	TimeSheetTemplateAssignment	A list of profiles that the template is assigned to.
workWeekEndDay	DaysOfWeek (enumeration of type string)	Required. The end day of the template's work week. One of the following values: - Monday - Tuesday - Wednesday - Thursday - Friday - Saturday - Sunday
workWeekStartDay	DaysOfWeek (enumeration of type string)	Required. The start day of the template's work week. One of the following values: - Monday - Tuesday - Wednesday - Thursday - Friday - Saturday - Sunday

TimeSheetTemplateAssignment

Returns a quick action that's associated with an EmbeddedServiceLiveAgent setup. The quick action includes the pre-chat form fields that the embedded chat window displays and shows the order in which the fields are displayed.

Field Name	Field Type	Description
assignedTo	string	The IDs of the user profiles that a time sheet template is assigned to.

Declarative Metadata Sample Definition

The following is an example of a TimeSheetTemplate file.

```
<?xml version="1.0" encoding="UTF-8"?>
<TimeSheetTemplate xmlns="http://soap.sforce.com/2006/04/metadata">
  <active>true</active>
  <description>Time Sheet Template description</description>
  <frequency>Daily</frequency>
  <masterLabel>label</masterLabel>
  <startDate>2018-10-18</startDate>
  <timeSheetTemplateAssignments>
    <assignedTo>admin</assignedTo>
  </timeSheetTemplateAssignments>
  <timeSheetTemplateAssignments>
    <assignedTo>standard</assignedTo>
  </timeSheetTemplateAssignments>
  <workWeekEndDay>Tuesday</workWeekEndDay>
  <workWeekStartDay>Monday</workWeekStartDay>
</TimeSheetTemplate>
```

The following is an example package.xml that references the previous definition.

```
<?xml version="1.0" encoding="UTF-8"?>
<Package xmlns="http://soap.sforce.com/2006/04/metadata">
  <types>
    <members>*</members>
    <name>TimeSheetTemplate</name>
  </types>
  <version>46.0</version>
</Package>
```

Wildcard Support in the Manifest File

This metadata type supports the wildcard character * (asterisk) in the package.xml manifest file. For information about using the manifest file, see [Deploying and Retrieving Metadata with the Zip File](#).

Field Service Tooling API

The following Tooling objects are used in Field Service.

IN THIS SECTION:

[CleanRule](#)

Represents a data integration rule that controls how a data service adds and updates data for existing records in an org.

[TimeSheetTemplate](#)

Represents a template for creating time sheets in Field Service. Available in API version 46.0 and later.

CleanRule

Represents a data integration rule that controls how a data service adds and updates data for existing records in an org.

This object is available in API version 38.0 and later.

Supported SOAP Calls

`create()`, `delete()`, `describeObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Supported REST HTTP Methods

GET

Fields

Field	Details
CleanDataServiceId	Type reference Properties Create, Filter, Group, Sort Description A foreign key reference to the CleanDataService that processes this CleanRule.
DataAssessmentStatus	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable Description The status of the data assessment. The following are valid values: • Hidden (default) • Not Started • In Progress • Pending Aggregation • Aggregation Complete • Failed Aggregation • Aborted This field is read only.
Description	Type textarea Properties Create, Filter, Group, Nillable, Sort, Update Description User-friendly text that describes the data integration rule.

Field	Details
DeveloperName	Type string Properties Create, Filter, Group, , Sort, Update Description This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. This unique name prevents conflicts with rules from other packages that have the same MasterLabel.  Note: Only users with View DeveloperName OR View Setup and Configuration permission can view, group, sort, and filter this field.
IsBulkEnabled	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description If this flag is set to <code>true</code>, the system automatically applies the rule to existing records whenever the rule is updated or saved. If the flag is set to <code>false</code>, the system doesn't automatically apply the rule to existing records. You can always apply the rule manually.
IsSilentSaveEnabled	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description If <code>false</code>, updates don't modify <code>LastModifiedDate</code> and <code>LastModifiedById</code> when it applies this rule; otherwise, updates insert the current date and the current user.
Language	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The language of the data integration rule. The following values are permitted: - Chinese (Simplified): <code>zh_CN</code> - Chinese (Traditional): <code>zh_TW</code> - Danish: <code>da</code> - Dutch: <code>n1_NL</code> - English: <code>en_US</code> - Finnish: <code>fi</code>

Field	Details
	• French: <code>fr</code> • German: <code>de</code> • Italian: <code>it</code> • Japanese: <code>ja</code> • Korean: <code>ko</code> • Norwegian: <code>no</code> • Portuguese (Brazil): <code>pt_BR</code> • Russian: <code>ru</code> • Spanish: <code>es</code> • Spanish (Mexico): <code>es_MX</code> Spanish (Mexico) defaults to Spanish for customer-defined translations. • Swedish: <code>sv</code> • Thai: <code>th</code> The Salesforce user interface is fully translated to Thai, but Help is in English.
MatchConfidence	Type double Properties Create, Filter, Nillable, Sort, Update Description Indicates how closely a Lightning Data record matches a Salesforce record. The higher the minimum value is set, the more precise the matches. This field is used in packages that support match scores for third-party data. Valid values include an integer determined by the data service provider. If the value is null, the default value from the data service provider is used. Available in API version 45.0 and later.
MasterLabel	Type string Properties Create, Filter, Group, Sort, Update Description Master label for this object. This display value is the internal label that is not translated.
MatchRule	Type string Properties Create, Filter, Group, Nillable, Sort Description An internal label for the matching rule in the data service that's associated with this CleanRule.
ShouldBypassTriggers	Type boolean

Field	Details
	Properties Create, Defaulted on create, Filter, Group, Sort, Update Description If <code>true</code>, indicates that the system does not apply triggers when it applies this rule; otherwise, the system applies the triggers.
ShouldBypassWorkflow	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description If <code>true</code>, the system bypasses workflow rules when it applies the data integration rule; otherwise, the system applies workflow rules.
SubjectType	Type picklist Properties Create, Filter, Group, Restricted picklist, Sort Description The standard or custom object in your org that's acted on by this CleanRule. The possible values are: • Account • Address • Contact • CustomEntityDefinition • Lead • ResourceAbsence • ServiceAppointment • ServiceTerritory • ServiceTerritoryMember • WorkOrder • WorkOrderLineItem In addition, custom objects with data integration rules are possible.  Note: The standard objects are installed with default data integration rules, but only the default rules for account, contact, and lead can be modified.
SourceSubjectType	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort

Field	Details
	Description A data service object associated with this CleanRule. The set of picklist values includes all the object types defined in the data service. However, if you specify a non-existent object, the API call returns an error. If you enable the Salesforce data services, the following values appear. CustomEntityDefinition Information retrieved from an external source in the form of an external object to enrich an account, contact, or lead. We map the external object fields to the account, contact, and lead lookup and details. DataCloudAddress Geolocation data service. DatacloudDandBCompany Data.com data service. This service is only available in the Professional, Enterprise, Unlimited, and Performance Editions, and only with a Premium Clean license. Your administrator must enable this data service by enabling the data integration rules for account and lead. If you install Marketplace packages that contain data services, their object names also appear in the set of picklist values.
Status	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description Status of the data integration rule. Valid values are <code>Active</code> and <code>Inactive</code>.

TimeSheetTemplate

Represents a template for creating time sheets in Field Service. Available in API version 46.0 and later.

Supported SOAP Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Supported REST HTTP Methods

`GET`, `POST`, `PATCH`, `DELETE`

Special Access Rules

Field Service must be enabled to access TimeSheetTemplate. Users must have the Customize Application and Time Sheet Template permissions.

Limitations

SOQL Limitations

Fields

Field	Details
Active	Type boolean Properties Defaulted On Create, Filter, Group, Sort Description Indicates whether the time sheet template is active (<code>true</code>) or not (<code>false</code>).
Description	Type textarea Properties Nillable Description The time sheet template's description.
DeveloperName	Type string Properties Filter, Group, Sort Description The API name of the time sheet template. Can contain alphanumeric characters and underscores, and must begin with a letter. Only users with View DeveloperName OR View Setup and Configuration permission can view, group, sort, and filter this field.
Frequency	Type picklist Properties Filter, Group, Restricted picklist, Sort Description Defines the frequency of the time sheet creation period. One of the following values: • Daily • Weekly • EveryTwoWeeks • TwiceAMonth • Monthly

Field	Details
Fullname	Type string Properties Create, Group, Nillable Description The full name of the associated metadata object in Metadata API. Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.
Language	Type picklist Properties Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort Description The language associated with the time sheet template.
ManageableState	Type ManageableState enumerated list Properties Filter, Group, Nillable, Restricted picklist, Sort Description Indicates the manageable state of the specified component that is contained in a package: • beta • deleted • deprecated • deprecatedEditable • installed • installedEditable • released • unmanaged Available in API version 48.0 and later.
MasterLabel	Type string Properties Filter, Group, Sort Description The name of the time sheet template.

Field	Details
Metadata	Type mns:TimeSheetTemplate on page 554 Properties Create, Nillable, Update Description The TimeSheetTemplate metadata.
NamespacePrefix	Type string Properties Filter, Group, Nillable, Sort Description The namespace that identifies the package that manages the time sheet template. Available in API version 48.0 and later.
StartDate	Type date Properties Filter, Group, Sort Description The date when the time sheet takes effect.
TimeSheetTemplateAssignments	Type QueryResult Properties Nillable Description A list of profiles that the template is assigned to. This field is only visible when the template has been assigned to at least one user profile. Available in API version 48.0 and later.
WorkWeekEndDay	Type picklist Properties Defaulted On Create, Filter, Group, Restricted picklist, Sort Description The end day of the template's work week. One of the following values: - Monday - Tuesday - Wednesday - Thursday - Friday

Field	Details
	• Saturday • Sunday (default)
WorkWeekStartDay	Type picklist Properties Defaulted On Create, Filter, Group, Restricted picklist, Sort Description The start day of the template's work week. One of the following values: • Monday • Tuesday • Wednesday • Thursday • Friday • Saturday • Sunday (default)

Field Service Apex Namespace

The Field Service managed package provides the FSL Apex namespace, whose custom methods and functions you can use to extend managed package features programmatically.

IN THIS SECTION:

[FSL Apex Namespace](#)

The Field Service (FSL) namespace contains all classes, methods, Visualforce pages, and custom objects within the Field Service managed package. To allow access to namespace elements, assign the FSL custom permission set that's appropriate for the user's persona. For example, to allow a user to book appointments, assign the FSL Agent custom permission set.

SEE ALSO:

[Install the Field Service Managed Package](#)

FSL Apex Namespace

The Field Service (FSL) namespace contains all classes, methods, Visualforce pages, and custom objects within the Field Service managed package. To allow access to namespace elements, assign the FSL custom permission set that's appropriate for the user's persona. For example, to allow a user to book appointments, assign the FSL Agent custom permission set.

IN THIS SECTION:

[AdvancedGapMatrix Class](#)

Represents a matrix of service resource IDs and graded time slots. An instance of this class is used only within the context of the results returned by the `GradeSlotsService` class.

[AppointmentBookingService Class](#)

Represents the appointment booking scheduling process in field service. Appointment booking returns the available slots for a service appointment, while considering scheduling policies, work rules, and service objectives. For example, we can schedule an appointment tomorrow between 9 and 11 AM, or next Monday between 4 and 6 PM.

[AppointmentBookingSlot Class](#)

Represents a booking slot available for a given service appointment. An instance of this class is part of the results returned by the `AppointmentBookingService` and `GradeSlotsService` classes.

[GradeSlotsService Class](#)

Represents the results shown in the Candidates quick action. Use the `GradeSlotsService` class to evaluate all possible slots where a given service appointment can be scheduled.

[AppointmentInsightsResult Class](#)

Represents the response returned by the `getAppointmentInsights` on page 618 method. `AppointmentInsightsResult` includes details about why a service appointment can't be added to the Gantt, including resource availability, blocked slots, and applicable blocking rules.

[BlockingRule Class](#)

Represents a global class that holds an array of blocking rules. `BlockingRule` is a required parameter for `AppointmentInsightsResult` on page 582, which is the response class for the `getAppointmentInsights` on page 618 method of `ScheduleService` on page 615.

[GroupData Class](#)

Represents a global class that combines a group policy ID and a list of service territory IDs. `GroupData` is a required input parameter for `FSL.ScheduleJobsApi.getJob.setTerritory` on page 611.

[OAAS Class](#)

Contains all the methods used in the available optimization services, such as global or in-day optimizations, reshuffle operations, and resource schedule optimizations.

[OAASRequest Class](#)

Represents the request sent in the `OAAS.Optimize()` method. This class contains all the details of a global optimization call.

[PolygonUtils Class](#)

Represents the result of using the `getTerritoryIdByPolygons` method to return a service territory record ID using latitude and longitude geolocation values.

[RecurringAppointmentsManager Class](#)

Represents the result of using the `getRecurringAppointmentSlots` on page 604 method to return a pattern of appointments that recur weekly.

[RecurringPattern Class](#)

Represents a global sharing class that is a required parameter for `RecurringAppointmentsManager` on page 603 to return a pattern of appointments that recur weekly.

[RecurringSequence Class](#)

Represents the result of using the `getRecurringAppointmentSlots` on page 604 method to return a pattern of appointments that recur daily, weekly, or monthly.

[ResourceScheduleData Class](#)

Contains all the details of a service resource's available slots for a given service appointment. An instance of this class is used only within the context of the results returned by the `AdvancedGapMatrix` class.

[ScheduleJobsApi Class](#)

Represents configurations to the Territory tab that is available for all scheduled jobs that are categorized under Enhanced Optimization.

[ScheduleResult Class](#)

Represents the result of scheduling a service appointment that is returned by the `ScheduleService.schedule` method.

[ScheduleService Class](#)

Represents the scheduling of a given service appointment in the best available slot based on the applied scheduling policy.

[SchedulingOption Class](#)

Provides the details of an individual slot available for scheduling. Represents a run-time object used within the context of the results returned in the `AdvancedGapMatrix` class.

[TimeInterval Class](#)

Holds an interval's start and end times. An instance of this class is used when capturing the start and end times of a scheduling horizon, during which a scheduling process runs. It considers scheduled jobs and is also used to represent the start and end times of an appointment slot or window.

AdvancedGapMatrix Class

Represents a matrix of service resource IDs and graded time slots. An instance of this class is used only within the context of the results returned by the `GradeSlotsService` class.

Namespace

[FSL](#)

Example

For an example of how to parse the class results, see [GradeSlotsService Class](#).

IN THIS SECTION:

[AdvancedGapMatrix Properties](#)

AdvancedGapMatrix Properties

`AdvancedGapMatrix` contains the following properties.

IN THIS SECTION:

[FSLOperationId](#)

This field only applies to scheduling service appointments that don't use Enhanced Scheduling and Optimization (ESO). Populated when scheduling a chain of two service appointments with scheduling dependencies, also known as complex work. An asynchronous call is made to schedule both service appointments in the chain. This property contains the `FSL__FSL_Operation__c` ID.

[partialResults](#)

Relevant only when Limit Apex Operations is enabled in the Field Service Admin app. If the processing time of the `ScheduleService` is about to exceed the maximum CPU limits, it returns the results that were already calculated and halts the process. This list contains the details of the partial, processed results. All other parameters are the same.

[resourceIDToScheduleData](#)

A map that returns service resource IDs and their corresponding available slots stored in the `ResourceScheduleData` class.

[Service](#)

The service appointment for which the `AdvancedGapMatrix` was run.

FSLOperationId

This field only applies to scheduling service appointments that don't use Enhanced Scheduling and Optimization (ESO). Populated when scheduling a chain of two service appointments with scheduling dependencies, also known as complex work. An asynchronous call is made to schedule both service appointments in the chain. This property contains the `FSL__FSL_Operation__c` ID.

Signature

```
public Id FSLOperationId {get; set;}
```

Property Value

Type: [Id](#)

Usage

To learn more about the complex work feature, see [Create Scheduling Dependencies Between Service Appointments](#).

partialResults

Relevant only when Limit Apex Operations is enabled in the Field Service Admin app. If the processing time of the `ScheduleService` is about to exceed the maximum CPU limits, it returns the results that were already calculated and halts the process. This list contains the details of the partial, processed results. All other parameters are the same.

Signature

```
public List<FSL.PartialResultsInfo> partialResults {get; set;}
```

Property Value

Type: [List<FSL.PartialResultsInfo>](#)

resourceIDToScheduleData

A map that returns service resource IDs and their corresponding available slots stored in the `ResourceScheduleData` class.

Signature

```
public Map<Id, FSL.ResourceScheduleData> resourceIDToScheduleData {get; set;}
```

Property Value

Type: [Map<Id, FSL.ResourceScheduleData>](#)

Service

The service appointment for which the `AdvancedGapMatrix` was run.

Signature

```
public Service {get; set;}
```

Property Value

Type: `ServiceAppointment`

AppointmentBookingService Class

Represents the appointment booking scheduling process in field service. Appointment booking returns the available slots for a service appointment, while considering scheduling policies, work rules, and service objectives. For example, we can schedule an appointment tomorrow between 9 and 11 AM, or next Monday between 4 and 6 PM.

Namespace

[FSL](#)

Usage

Before calling the `AppointmentBookingService` class, make sure that the parent work order and service appointment already exist.



Note: When scheduling a service appointment, the user scheduling the appointment must have one of four managed package permission sets: Field Service Admin, Field Service Dispatcher, Field Service Agent, Self-Service. If you're using platform events to schedule appointments, you must explicitly configure the user so that the user has the correct permissions. Without the proper configuration, the platform event runs as the Automated Process system user and doesn't have the correct permissions to schedule an appointment. To learn more, see [Configure the User and Batch Size for Your Platform Event Trigger](#).

The time zone is typically based on the location of the customer requesting the service. As a best practice, use the time zone specified on the service territory's operating hours.

If you want to return time slots ordered by grade or by date, use the `getSlots()` method that accepts a sort parameter; either `sortResultsBy` or `sortBy`.

When the `AppointmentBookingService` class is called, the `ExactAppointment` value on the associated work type is ignored. If exact appointments are needed, set this value to `true` in the parameters. If you want to respect the `ExactAppointment` of the work type, query it directly from the work type.

After receiving the slots, the developer can decide how to display, or manage, the slots. Typically, the slots are displayed to the customer, who selects a slot. After a slot is selected, the service appointment's `ArrivalWindowStartTime` and `ArrivalWindowEndTime` fields are updated with the slot's start and end times; for example:

```
FSL.AppointmentBookingSlot slot = slots[0];
sa.ArrivalWindowStartTime = slot.Interval.Start.addSeconds(tz.getOffset(slot.Interval.Start)
/ -1000);
```

```
sa.ArrivalWindowEndTime = slot.Interval.Finish.addSeconds(tz.getOffset(slot.Interval.Finish)
/ -1000);

update sa;
```

The `FSL.ScheduleService` method is then called so that the service appointment is scheduled correctly within the selected slot.

IN THIS SECTION:

[AppointmentBookingService Methods](#)

AppointmentBookingService Methods

`AppointmentBookingService` includes the following static methods.

IN THIS SECTION:

[getSlots\(serviceID, policyId, operatingHoursId, tz, exactAppointment\)](#)

Returns a list of `FSL.AppointmentBookingSlot` records.

[getSlots\(serviceID, policyId, calendar, tz, exactAppointment\)](#)

Returns a list of `FSL.AppointmentBookingSlot` records.

[getSlots\(serviceID, policyId, calendar, tz, sortResults, exactAppointment\)](#)

Returns a sorted list of `FSL.AppointmentBookingSlot` records.

[getSlots\(serviceID, policyId, calendar, tz, sortBy, exactAppointment\)](#)

Returns a sorted list of `FSL.AppointmentBookingSlot` records.

[getABSLOTS\(serviceID, policyId, calendar, tz, sortBy, exactAppointment\)](#)

Returns a list of `FSL.AppointmentBookingSlot` records that respect complex work dependencies for a chain of service appointments.

getSlots(serviceID, policyId, operatingHoursId, tz, exactAppointment)

Returns a list of `FSL.AppointmentBookingSlot` records.

Signature

```
public static List<FSL.AppointmentBookingSlot> getSlots(Id serviceID, Id policyId, Id
operatingHoursId, System.TimeZone tz, Boolean exactAppointment)
```

Parameters

serviceID

Type: `Id`

Required. The ID of the service appointment being scheduled.

policyId

Type: `Id`

Required. The ID of the scheduling policy being used.

operatingHoursId

Type: [Id](#)

Required. The ID of the operating hours record used to determine time slot intervals.



Note: The *TimeZone* on the operating hours record is ignored.

tz

Type: [System.TimeZone](#)

Required. The time zone in which the slots are returned. This time zone must be the time zone of the service territory in which the service appointment is performed. If any other timezone is used, the appointment booking slots are converted to the service territory time zone at run time.

exactAppointment

Type: [Boolean](#)

Required. Specifies whether the result uses exact appointments or an arrival window. When the `getSlots()` method is called, the *exactAppointment* value on the work type is ignored.

Return Value

Type: [List<FSL.AppointmentBookingSlot>](#)

Usage

This method returns a list of appointment booking time slots available for a given service appointment. Results are returned in the time zone sent in the original request.

Appointment windows depend on the *operatingHoursId* value provided in the call. The *operatingHoursId* represents various slots offered to the customer during appointment booking (for example, Mondays 9 to 11 AM, 11 AM to 1 PM, 1 to 3 PM; Tuesdays 9 to 11 AM, 12 to 4 PM; and so forth.)

This method only returns available slots and doesn't schedule the service appointment. After the end user selects the desired slot, update the *ArrivalWindowStartTime* and *ArrivalWindowEndTime* properties on the service appointment and call the [FSL.ScheduleService](#) method.

To convert dateTime values to UTC before they're updated in the database, perform a time zone conversion.

Example

This example calls the `AppointmentBookingService` class using the Customer First scheduling policy and the Gold Appointments Calendar operating hours as the desired appointment slots.

```
// FSL.AppointmentBookingService
// The getSlots method returns a list of AppointmentBookingSlot objects.

ServiceAppointment sa = [SELECT Id, EarliestStartTime, DueDate FROM ServiceAppointment
                          WHERE Id='07r3F0000009dTSGEC'];
Id schedulingPolicyId = [SELECT Id FROM FSL__Scheduling_Policy__c
                          WHERE Name='Customer first' LIMIT 1].Id;
Id operatingHoursId    = [SELECT id FROM OperatingHours
                          WHERE name='Gold Appointments Calendar' LIMIT 1].Id;
Timezone tz = UserInfo.getTimeZone();

List<FSL.AppointmentBookingSlot> slots = FSL.AppointmentBookingService.GetSlots(sa.Id,
```

```
schedulingPolicyId, operatingHoursId, tz, false);

System.debug('Returned ' + slots.size() + ' appointment slots');
for(integer i=0; i<slots.size(); i++){
    system.debug('Slot:'+i+' Start: '+slots[i].Interval.Start+' Finish:'+
slots[i].Interval.Finish+' Grade: '+slots[i].Grade);
}
```

Here's an implementation suggestion for returning time slots in a different time zone. In this example, we're returning the time slots in the Europe/Jersey time zone. The code doesn't take daylight saving time into account, which can cause issues when daylight saving time begins or ends.

```
ServiceAppointment sa = [SELECT Id, EarliestStartTime,
DueDate,serviceterritory.operatinghours.timezone FROM ServiceAppointment WHERE
Id='08pTC0000000rCPYAY'];

Id schedulingPolicyId = [SELECT Id FROM FSLQA__Scheduling_Policy__c WHERE
Id='a0cTC000000JDOB' LIMIT 1].Id;

Id operatingHoursId = [SELECT id FROM OperatingHours WHERE Id='0OHTC0000000c0U' LIMIT
1].Id;

Timezone tz = TimeZone.getTimeZone(sa.serviceterritory.operatinghours.timezone);//timezone
of the territory of the Service Appointment

List<FSLQA.AppointmentBookingSlot> slots = FSLQA.AppointmentBookingService.GetSlots(sa.Id,
schedulingPolicyId, operatingHoursId, tz, false);

Timezone JerseyTZ = TimeZone.getTimeZone('Europe/Jersey');//desired output timezone

for(integer i=0; i<slots.size(); i++){
//Remove the time offset of the current timezone.
    slots[i].Interval.Start =
slots[i].Interval.Start.addSeconds(tz.getOffset(slots[i].Interval.Start) / -1000);
    slots[i].Interval.Finish =
slots[i].Interval.Finish.addSeconds(tz.getOffset(slots[i].Interval.Finish) / -1000);
//Add the time offset of the new timezone.
    slots[i].Interval.Start =
slots[i].Interval.Start.addSeconds(JerseyTZ.getOffset(slots[i].Interval.Start) / 1000);
    slots[i].Interval.Finish =
slots[i].Interval.Finish.addSeconds(JerseyTZ.getOffset(slots[i].Interval.Finish) / 1000);
}
```

getSlots(serviceId, policyId, calendar, tz, exactAppointment)

Returns a list of FSL.AppointmentBookingSlot records.

Signature

```
public static List<FSL.AppointmentBookingSlot> getSlots(Id serviceId, Id policyId,
OperatingHours calendar, System.TimeZone tz, Boolean exactAppointment)
```


Parameters

*serviceID*Type: [Id](#)

Required. The ID of the service appointment being scheduled.

*policyId*Type: [Id](#)

Required. The ID of the scheduling policy being used.

*calendar*Type: [OperatingHours](#) objectRequired. The *calendar* is used to determine time slot intervals.**Note:** The *TimeZone* on the operating hours record is ignored.*tz*Type: [System.TimeZone](#)

Required. The time zone in which the slots are returned. This time zone must be the time zone of the service territory in which the service appointment is performed. If any other timezone is used, the appointment booking slots are converted to the service territory time zone at run time..

*exactAppointment*Type: [Boolean](#)Required. Specifies whether the result uses exact appointments or an arrival window. When the `getSlots()` method is called, the *exactAppointment* value on the work type is ignored.

Return Value

Type: [List<FSL.AppointmentBookingSlot>](#)

Usage

This method returns a list of appointment booking time slots for a given service appointment. Results are returned in the time zone sent in the original request. The appointment windows (9–11, 11–1, 1–3, and so on) depend on slots returned in a subquery of the [OperatingHours](#) object.

This method only returns available slots and does not schedule the service appointment. Update the *ArrivalWindowStartTime* and *ArrivalWindowEndTime* properties on the service appointment and call the [FSL.ScheduleService](#) method.

Perform a time zone conversion to convert *dateTime* values to UTC before they are updated in the database.

Example

This example shows the use of a subquery to retrieve time slots from *calendar*, an [OperatingHours](#) object.

```
// FSL.AppointmentBookingService
// The getSlots method returns a list of AppointmentBookingSlot objects.

ServiceAppointment sa = [SELECT Id, EarliestStartTime, DueDate FROM ServiceAppointment
                          WHERE Id='08p4F0000008sRMQAY'];
Id schedulingPolicyId = [SELECT Id FROM FSL__Scheduling_Policy__c
                          WHERE Name='Customer First' LIMIT 1].Id;
```

```

operatinghours oh = new operatinghours();
oh = [SELECT id, (SELECT EndTime, StartTime, Type, DayOfWeek FROM TimeSlots) FROM
OperatingHours
      WHERE name='AB Slot: 2 Hour Slots' LIMIT 1];
Timezone tz = UserInfo.getTimeZone();

List<FSL.AppointmentBookingSlot> slots = FSL.AppointmentBookingService.GetSlots(sa.Id,
schedulingPolicyId, oh, tz, false);

System.debug('Returned ' + slots.size() + ' appointment slots');
for(integer i=0; i<slots.size(); i++){
    system.debug('Slot:'+i+' Start: '+slots[i].Interval.Start+' Finish:'+
slots[i].Interval.Finish+' Grade: '+slots[i].Grade);
}

```

getSlots(serviceID, policyId, calendar, tz, sortResults, exactAppointment)

Returns a sorted list of `FSL.AppointmentBookingSlot` records.

Signature

```

public static List<FSL.AppointmentBookingSlot> getSlots(Id serviceID, Id policyId,
OperatingHours calendar, System.TimeZone tz, FSL.AppointmentBookingService.SortResultsBy
sortResults, Boolean exactAppointment)

```

Parameters

serviceID

Type: `Id`

Required. The ID of the service appointment being scheduled.

policyId

Type: `Id`

Required. The ID of the scheduling policy being used.

calendar

Type: `OperatingHours` object

Required. The *calendar* is used to determine time slot intervals.



Note: The *TimeZone* on the operating hours record is ignored.

tz

Type: `System.TimeZone`

Required. The time zone in which the slots are returned. This time zone must be the time zone of the service territory in which the service appointment is performed. If any other timezone is used, the appointment booking slots are converted to the service territory time zone at run time.

sortResults

Type: `FSL.AppointmentBookingService.SortResultsBy` enumeration

Required. Used to sort `AppointmentBookingSlot` results that are returned. Possible sort values are `Grade`, `NoSort`, and `SlotDateTime`.

exactAppointment

Type: [Boolean](#)

Required. Specifies whether the result uses exact appointments or an arrival window. When the `getSlots()` method is called, the *exactAppointment* value on the work type is ignored.

Return Value

Type: [List<FSL.AppointmentBookingSlot>](#)

Usage

This method returns a sorted list of appointment booking time slots for a given service appointment. Results are returned in the time zone sent in the original request. The appointment windows (9–11, 11–1, 1–3, and so on) depend on slots returned in a subquery of *calendar*, an *OperatingHours* object.

This method only returns available slots and doesn't schedule the service appointment. Update the *ArrivalWindowStartTime* and *ArrivalWindowEndTime* properties on the service appointment and call the [FSL.ScheduleService](#) method.

Perform a time zone conversion to convert *dateTime* values to UTC before they are updated in the database.

Example

This example sorts the returned slots by grade. When using a signature that contains a *calendar* parameter, use a subquery for time slots in the *OperatingHours* object.

```
// FSL.AppointmentBookingService
// The getSlots method returns a list of AppointmentBookingSlot objects.

ServiceAppointment sa = [SELECT Id, EarliestStartTime, DueDate FROM ServiceAppointment
                          WHERE Id='08p4F0000008sRMQAY'];
Id schedulingPolicyId = [SELECT Id FROM FSL__Scheduling_Policy__c
                          WHERE Name='Customer First' LIMIT 1].Id;
operatinghours oh = new operatinghours();
oh = [SELECT id, (SELECT EndTime, StartTime, Type, DayOfWeek FROM TimeSlots) FROM
      OperatingHours
      WHERE name='AB Slot: 2 Hour Slots' LIMIT 1];
Timezone tz = UserInfo.getTimeZone();

FSL.AppointmentBookingService.SortResultsBy sortResults =
FSL.AppointmentBookingService.SortResultsBy.Grade;
List<FSL.AppointmentBookingSlot> slots = FSL.AppointmentBookingService.GetSlots(sa.Id,
schedulingPolicyId, oh, tz, sortResults, false);

System.debug('Returned ' + slots.size() + ' appointment slots');
for(integer i=0; i<slots.size(); i++){
    system.debug('Slot:' + i + ' Start: ' + slots[i].Interval.Start + ' Finish: ' +
slots[i].Interval.Finish + ' Grade: ' + slots[i].Grade);
}
```

getSlots(serviceId, policyId, calendar, tz, sortBy, exactAppointment)

Returns a sorted list of `FSL.AppointmentBookingSlot` records.

Signature

```
public static List<FSL.AppointmentBookingSlot> getSlots(Id serviceID, Id policyId,
OperatingHours calendar, System.TimeZone tz, String sortBy, Boolean exactAppointment)
```

Parameters

serviceID

Type: [Id](#)

Required. The ID of the service appointment being scheduled.

policyId

Type: [Id](#)

Required. The ID of the scheduling policy being used.

calendar

Type: [OperatingHours](#) object

Required. The *calendar* is used to determine time slot intervals.



Note: The *TimeZone* on the operating hours record is ignored.

tz

Type: [System.TimeZone](#)

Required. The time zone in which the slots are returned. This time zone must be the time zone of the service territory in which the service appointment is performed. If any other timezone is used, the appointment booking slots are converted to the service territory time zone at run time.

sortBy

Type: [String](#)

Required. Used to sort AppointmentBookingSlot results. Acceptable values are SORT_BY_GRADE, SORT_BY_DATE, and SORT_BY_NO_SORT.

exactAppointment

Type: [Boolean](#)

Required. Specifies whether the result uses exact appointments or an arrival window. When the `getSlots()` method is called, the *exactAppointment* value on the work type is ignored.

Return Value

Type: [List<FSL.AppointmentBookingSlot>](#)

Usage

This method returns a sorted list of appointment booking time slots for a given service appointment. Results are returned in the time zone sent in the original request. The appointment windows (9–11, 11–1, 1–3, and so on) depend on slots returned in a subquery of *calendar*, an [OperatingHours](#) object.

This method only returns available slots and doesn't schedule the service appointment. Update the *ArrivalWindowStartTime* and *ArrivalWindowEndTime* properties on the service appointment and call the [FSL.ScheduleService](#) method.

Perform a time zone conversion to convert dateTime values to UTC before they are updated in the database.

getABSlots(serviceID, policyId, calendar, tz, sortBy, exactAppointment)

Returns a list of `FSL.AppointmentBookingSlot` records that respect complex work dependencies for a chain of service appointments.

Signature

```
public static FSL.ABSlotsResponse getABSlots(Id serviceID, Id policyId, OperatingHours calendar, System.TimeZone tz, String sortBy, Boolean exactAppointment)
```

Parameters

serviceID

Type: `Id`

Required. The ID of the service appointment being scheduled.

policyId

Type: `Id`

Required. The ID of the scheduling policy being used.

calendar

Type: `OperatingHours` object and related `TimeSlot` object

Required. The *calendar* is used to determine time slot intervals.



Note: The *TimeZone* on the operating hours record is ignored.

tz

Type: `System.TimeZone`

Required. The time zone in which the slots are returned. This time zone must be the time zone of the service territory in which the service appointment is performed. If any other timezone is used, the appointment booking slots are converted to the service territory time zone at run time.

sortBy

Type: `String`

Required. Used to sort the results. Acceptable values are `SORT_BY_GRADE`, `SORT_BY_DATE`, and `SORT_BY_NO_SORT`.

exactAppointment

Type: `Boolean`

Required. Specifies whether the result uses exact appointments or an arrival window. When this method is called, the *exactAppointment* value on the work type is ignored.

Return Value

Type: `FSL.ABSlotsResponse`

Usage

This method is used to retrieve appointment booking slots for a complex work chain of two appointments. This method respects the complex work setting **Use all-or-none scheduling for related appointments**.

The appointment booking slots returned by this method are for the service appointment that was sent to the API.

If Enhanced Scheduling and Optimization (ESO) is enabled:

- The method runs synchronously.
- The method always uses the ESO behavior, which is all-or-none for related appointments scheduling of complex work.
- The method is valid for a chain of up to five to appointments.

If ESO isn't enabled, this method is asynchronous. To examine results that the asynchronous method returns, use the streaming API and subscribe to `MstCompletedChannel`, the channel for the Field Service managed package.

This method only returns available slots and doesn't schedule the service appointment. Update the `ArrivalWindowStartTime` and `ArrivalWindowEndTime` properties on the service appointment and call the `scheduleExtended` method of the `FSLScheduleService` class.

Perform a time zone conversion to convert `dateTime` values to UTC before they're updated in the database.

Example

This code example retrieves data related to a specific service appointment, including details about the appointment, scheduling policy, and operating hours. It then uses this data to fetch available appointment slots based on the user's timezone and logs the details of these slots for debugging purposes.

```
// Retrieve a specific ServiceAppointment record by its ID
ServiceAppointment sa = [
    SELECT Id, EarliestStartTime, DueDate
    FROM ServiceAppointment
    WHERE Id='08p4F0000008sRMQAY'
];

// Retrieve the ID of a Scheduling Policy by its name
Id schedulingPolicyId = [
    SELECT Id
    FROM FSL__Scheduling_Policy__c
    WHERE Name='Customer First'
    LIMIT 1
].Id;

// Retrieve Operating Hours and its related Time Slots by name
OperatingHours oh = [
    SELECT Id, (SELECT EndTime, StartTime, Type, DayOfWeek,
FSL__Designated_Work_Boolean_Fields__c
    FROM TimeSlots)
    FROM OperatingHours
    WHERE Name='AB Slot: 2 Hour Slots'
    LIMIT 1
];

// Retrieve the current user's timezone
TimeZone tz = UserInfo.getTimeZone();

// Call a method to get available appointment slots
FSL.ABSlotsResponse ABSlotsResponse = FSL.AppointmentBookingService.getABSlots(sa.Id,
schedulingPolicyId, oh, tz, 'SORT_BY_DATE', false);

// Debug statement to log the number of returned slots
List<FSL.AppointmentBookingSlot> slots = ABSlotsResponse.Slots;
System.debug('Returned ' + slots.size() + ' appointment slots');
```

```
// Loop through each slot and log its details
for (integer i = 0; i < slots.size(); i++) {
    System.debug('Slot:' + i + ' Start: ' + slots[i].Interval.Start + ' Finish: ' +
slots[i].Interval.Finish + ' Grade: ' + slots[i].Grade);
}
```

SEE ALSO:

[Example: Subscribe to and Replay Events Using a Visualforce Page](#)

AppointmentBookingSlot Class

Represents a booking slot available for a given service appointment. An instance of this class is part of the results returned by the `AppointmentBookingService` and `GradeSlotsService` classes.

Namespace

[FSL](#)

Example

The debug log shows the relevant parts of the result: `Grade` and `Interval`. The `BestSlotGrades` parameters are private and inaccessible via code.

```
AppointmentBookingSlot:
[
  Grade=85.1851851851851851851851851852,
  Interval=TimeInterval:
  [
    2018-01-26 09:00:00,2018-01-26 11:00:00
  ]
]
```

IN THIS SECTION:

[AppointmentBookingSlot Properties](#)

AppointmentBookingSlot Properties

`AppointmentBookingSlot` contains the following properties.

IN THIS SECTION:

[grade](#)

The grade of the available appointment booking slot.

[interval](#)

The time frame of the returned appointment booking slot.

grade

The grade of the available appointment booking slot.

Signature

```
public Decimal grade {get; set;}
```

Property Value

Type: [Decimal](#)

interval

The time frame of the returned appointment booking slot.

Signature

```
public FSL.TimeInterval interval {get; set;}
```

Property Value

Type: [FSL.TimeInterval](#)

GradeSlotsService Class

Represents the results shown in the Candidates quick action. Use the `GradeSlotsService` class to evaluate all possible slots where a given service appointment can be scheduled.

Namespace

[FSL](#)

Usage



Note: When scheduling a service appointment, the user scheduling the appointment must have one of four managed package permission sets: Field Service Admin, Field Service Dispatcher, Field Service Agent, Self-Service. If you're using platform events to schedule appointments, you must explicitly configure the user so that the user has the correct permissions. Without the proper configuration, the platform event runs as the Automated Process system user and doesn't have the correct permissions to schedule an appointment. To learn more, see [Configure the User and Batch Size for Your Platform Event Trigger](#).

IN THIS SECTION:

[GradeSlotsService Constructors](#)

[GradeSlotsService Methods](#)

GradeSlotsService Constructors

The following are constructors for `GradeSlotsService`.

IN THIS SECTION:

[GradeSlotsService\(schedulingPolicyId, serviceAppointmentId\)](#)

Creates a new instance of the `GradeSlotsService` class using the specified scheduling policy ID and service appointment ID.

GradeSlotsService(schedulingPolicyId, serviceAppointmentId)

Creates a new instance of the `GradeSlotsService` class using the specified scheduling policy ID and service appointment ID.

Signature

```
public GradeSlotsService(Id schedulingPolicyId, Id serviceAppointmentId)
```

Parameters

schedulingPolicyId

Type: `Id`

The record ID of the scheduling policy being used to schedule the service appointment.

serviceAppointmentId

Type: `Id`

The record ID of the service appointment being scheduled.

GradeSlotsService Methods

`GradeSlotsService` includes the following static method.

IN THIS SECTION:

[`getGradedMatrix\(i\_ResultsInUserTimeZone\)`](#)

Returns a matrix of resource IDs and graded time slots. The information is similar to that shown in the Candidates quick action.

getGradedMatrix(i_ResultsInUserTimeZone)

Returns a matrix of resource IDs and graded time slots. The information is similar to that shown in the Candidates quick action.

Signature

```
public FSL.AdvancedGapMatrix getGradedMatrix(Boolean i_ResultsInUserTimeZone)
```

Parameters

i_ResultsInUserTimeZone

Type: `Boolean`

If `true`, all DateTimes are returned in the user's time zone. If `false`, all DateTimes are returned in UTC.

Return Value

Type: `FSL.AdvancedGapMatrix`

Usage

If Enhanced Scheduling and Optimization (ESO) is enabled, this method runs synchronously. If ESO isn't enabled, this method can be called with only one service appointment at a time, and runs asynchronously. To examine results that the asynchronous method returns, use the streaming API and subscribe to `MstCompletedChannel`, the channel for the Field Service managed package.

Example

This example illustrates how to parse the results in the `FSL.AdvancedGapMatrix` to extract the service resource ID, start and end times, and grade of each slot.

```
// FSL.GradeSlotsService class
// The getGradedMatrix method returns a matrix of resource id's AND graded time slots

Id serviceAppointmentId = '08p1N000000qN4sQAE';
Id schedulingPolicyId=[SELECT Id FROM FSL__Scheduling_Policy__c WHERE Name='Customer First'
LIMIT 1].Id;

// GENERATE the graded time slots for the service appointment
FSL.GradeSlotsService mySlotService = new
FSL.GradeSlotsService(schedulingPolicyId,serviceAppointmentId);

// STORE the matrix of service resource id's and graded time slots
FSL.AdvancedGapMatrix myResultMatrix = mySlotService.getGradedMatrix(true);

Map<Id,FSL.ResourceScheduleData> mySRGradedTimeSlotMap =
myResultMatrix.ResourceIDToScheduleData;
for (Id thisresourceid : mySRGradedTimeSlotMap.keySet()){
    for (FSL.SchedulingOption thisso :
mySRGradedTimeSlotMap.get(thisresourceid).SchedulingOptions ) {
        system.debug('***** Resource Id' + thisresourceid);
        system.debug('***** Start - ' + thisso.Interval.Start);
        system.debug('***** Finish - ' + thisso.Interval.Finish);
        system.debug('***** Grade - ' + thisso.Grade);
    }
}
```

AppointmentInsightsResult Class

Represents the response returned by the `getAppointmentInsights` on page 618 method. `AppointmentInsightsResult` includes details about why a service appointment can't be added to the Gantt, including resource availability, blocked slots, and applicable blocking rules.

Namespace

[FSL](#)

Example

This code block represents the output of the `getAppointmentInsights` method from the Appointment Insights API. It provides a detailed result of an appointment insight operation. The result indicates there are no slots blocked.

```
AppointmentInsightsResult:[
blockedSlots=0,
blockingRules=(BlockingRule:[
    ruleName=Due Date,
    slotsBlockedByMultipleRules=0,
    slotsBlockedByRule=0
```

```

    },
    horizonEndDate=2024-06-06 14:00:00,
    horizonStartDate=2024-05-27 14:00:00,
    operationTimeStamp=2024-06-16 14:00:53,
    policyId=a0c8B00000638CMQAY,
    resourcesEvaluated=0,
    serviceAppointmentId=08p8B000000jCjBQAU,
    serviceTerritoryId=0Hh8B000000HrctSAC
  ]

```

IN THIS SECTION:

[Appointment Insights Result Properties](#)

Appointment Insights Result Properties

`AppointmentInsightsResult` contains the following properties.

IN THIS SECTION:

[blockedSlots](#)

The total number of potential blocked slots returned by the `getAppointmentInsights` method.

[blockingRules](#)

Array of blocking rules with the rule name and number of slots

[horizonStartDate](#)

The horizon start date as calculated and used by the operation.

[horizonEndDate](#)

The horizon end date as calculated and used by the operation.

[operationTimeStamp](#)

The time the operation was executed in Coordinated Universal Time (UTC).

[policyId](#)

The default policy ID or the policy ID provided in the request.

[resourcesEvaluated](#)

The total number of service resources loaded by fetch data and evaluated by the `getAppointmentInsights` method.

[serviceAppointmentId](#)

The ID of the service appointment used for the request.

[serviceTerritoryId](#)

The service territory associated with the service appointment. If there is no territory for the service appointment, keep this field empty. This property is optional.

blockedSlots

The total number of potential blocked slots returned by the `getAppointmentInsights` method.

Signature

```
global Integer blockedSlots {get; set;}
```

Property Value

Type: [Integer](#)

blockingRules

Array of blocking rules with the rule name and number of slots

Signature

```
global List<BlockingRule> blockingRules {get; set;}
```

Property Value

Type: [List<BlockingRule>](#)

horizonStartDate

The horizon start date as calculated and used by the operation.

Signature

```
global Datetime horizonStartDate {get; set;}
```

Property Value

Type: [Datetime](#)

horizonEndDate

The horizon end date as calculated and used by the operation.

Signature

```
global Datetime horizonEndDate {get; set;}
```

Property Value

Type: [Datetime](#)

operationTimeStamp

The time the operation was executed in Coordinated Universal Time (UTC).

Signature

```
global Datetime operationTimeStamp {get; set;}
```

Property Value

Type: [Datetime](#)

policyId

The default policy ID or the policy ID provided in the request.

Signature

```
global Id policyId {get; set;}
```

Property Value

Type: [Id](#)

resourcesEvaluated

The total number of service resources loaded by fetch data and evaluated by the `getAppointmentInsights` method.

Signature

```
global Integer resourcesEvaluated {get; set;}
```

Property Value

Type: [Integer](#)

serviceAppointmentId

The ID of the service appointment used for the request.

Signature

```
global Id serviceAppointmentId {get; set;}
```

Property Value

Type: [Id](#)

serviceTerritoryId

The service territory associated with the service appointment. If there is no territory for the service appointment, keep this field empty. This property is optional.

Signature

```
global Id serviceTerritoryId {get; set;}
```

Property Value

Type: [Id](#)

BlockingRule Class

Represents a global class that holds an array of blocking rules. . `BlockingRule` is a required parameter for [AppointmentInsightsResult](#) on page 582, which is the response class for the [getAppointmentInsights](#) on page 618 method of [ScheduleService](#) on page 615.

Namespace

[FSL](#)

Usage

`BlockingRule` is the Apex class type for `blockingRules`, a required parameter for [FSL.AppointmentInsightsResult](#).

IN THIS SECTION:

[BlockingRule Properties](#)

BlockingRule Properties

`BlockingRule` contains the following properties.

IN THIS SECTION:

[ruleName](#)

The name of the rule that is blocking the slot. When there is no rule name, use the rule type.

[slotsBlockedByMultipleRules](#)

The number of slots that are blocked by a combination of this rule and others.

[slotsBlockedByRule](#)

The number of slots that are blocked only by this rule.

ruleName

The name of the rule that is blocking the slot. When there is no rule name, use the rule type.

Signature

```
global String ruleName {get; set;}
```

Property Value

Type: [String](#)

slotsBlockedByMultipleRules

The number of slots that are blocked by a combination of this rule and others.

Signature

```
global Integer slotsBlockedByMultipleRules {get; set;}
```

Property Value

Type: [Integer](#)

slotsBlockedByRule

The number of slots that are blocked only by this rule.

Signature

```
global Integer slotsBlockedByRule {get; set;}
```

Property Value

Type: [Integer](#)

GroupData Class

Represents a global class that combines a group policy ID and a list of service territory IDs. `GroupData` is a required input parameter for [FSL.ScheduleJobsApi.getJob.setTerritory](#) on page 611.

Namespace

[FSL](#)

Usage

`GroupData` is a required input parameter for [FSL.ScheduleJobsApi.getJob.setTerritory](#). `GroupData` has two input parameters: `groupPolicyId` and a list of `territoryIds`.

Example

This sample code shows you how to populate `GroupData`.

```
global class GroupData {
    public String groupPolicyId { get; set; }
    public List<String> territoryIds { get; set; }

    global GroupData(String groupPolicyId, List<String> territoryIds) {
        this.groupPolicyId = groupPolicyId;
        this.territoryIds = territoryIds;
    }
}
```

See [FSL.ScheduleJobsApi.getJob.setTerritory](#) to see how to use `GroupData` with `setTerritory`.

IN THIS SECTION:

[GroupData Properties](#)

GroupData Properties

GroupData contains the following properties.

IN THIS SECTION:

[groupPolicyId](#)

The group policy of a job required by [FSL.ScheduleJobsApi.getJob.setTerritory](#).

[territoryIds](#)

A list of service territory IDs required by [FSL.ScheduleJobsApi.getJob.setTerritory](#).

groupPolicyId

The group policy of a job required by [FSL.ScheduleJobsApi.getJob.setTerritory](#).

Signature

```
public String groupPolicyId { get; set; }
```

Property Value

Type: [String](#)

territoryIds

A list of service territory IDs required by [FSL.ScheduleJobsApi.getJob.setTerritory](#).

Signature

```
public List<String> territoryIds { get; set; }
```

Property Value

Type: [String](#)

OAAS Class

Contains all the methods used in the available optimization services, such as global or in-day optimizations, reshuffle operations, and resource schedule optimizations.

Namespace

[FSL](#)

Usage

If you call OAAS APIs, running asynchronously, from the following methods, you receive a `Database.executeBatch` exception error message.

- `start batch` Apex method
- `execute batch` Apex method

- `future` methods

The limitation doesn't apply if you're using Enhanced Scheduling and Optimization. See [Using Batch Apex & Future Methods](#) in the *Apex Developer Guide*.

IN THIS SECTION:

[OAAS Methods](#)

OAAS Methods

OAAS includes the following methods.

IN THIS SECTION:

[optimize\(request\)](#)

Returns the optimization request ID of the global optimization process that was triggered. If the scheduling policy includes an in-day optimization Boolean that's set to true, the request triggers an in-day optimization.

[reshuffle\(serviceld, policyId\)](#)

Returns the optimization request ID of a reshuffle operation that has begun on a given service appointment.

[resourceDayOptimization\(resourceId, policyId, horizon, includeAllTasks, includeOnlyResourceFutureSA, radius, candidateSasFields, unschedulableServicesField, maxOptRuntime\)](#)

Returns the optimization request ID with the details of the optimization call. This method runs resource schedule optimization—the optimization of a single service resource's schedule—using the parameters provided.

[resourceDayOptimization\(resourceId, policyId, horizon, includeAllTasks, includeOnlyResourceFutureSA, radius, candidateSasFields, unschedulableServicesField, maxOptRuntime, nowTimeOnSchedule\)](#)

Returns the optimization request ID with the details of the optimization call. This method runs resource schedule optimization—the optimization of a single service resource's schedule—using the parameters provided.

[resourceDayOptimization\(resourceId, policyId, horizon, includeAllTasks, includeOnlyResourceFutureSA, radius, candidateSas, unschedulableServices, maxOptRuntime\)](#)

Returns the optimization request ID with the details of the optimization call. This method runs resource schedule optimization—the optimization of a single service resource's schedule—using the parameters provided.

[resourceDayOptimization\(resourceId, policyId, horizon, includeAllTasks, includeOnlyResourceFutureSA, radius, candidateSas, unschedulableServices, maxOptRuntime, nowTimeOnSchedule\)](#)

Returns the optimization request ID with the details of the optimization call. This method runs resource schedule optimization—the optimization of a single service resource's schedule—using the parameters provided.

optimize(request)

Returns the optimization request ID of the global optimization process that was triggered. If the scheduling policy includes an in-day optimization Boolean that's set to true, the request triggers an in-day optimization.

Signature

```
public static Id optimize (FSL.OAASRequest request)
```

Parameters

*request*Type: [FSL.OAASRequest](#)

The optimization request.

Return Value

Type: [Id](#)

Record ID of the optimization request.

Example

This example creates an instance of the `OAASRequest` class that holds all the details of the optimization call being initiated. Next, the example calls the `optimize` method and passes in the request.

```
DateTime start=DateTime.now();
DateTime finish=DateTime.now().addDays(3);

LIST<Id> lstServiceTerritories = new List<Id>();
lstServiceTerritories.add('0Hh0b000000cIwsCAE');

FSL.OAASRequest oaasRequest = new FSL.OAASRequest();
oaasRequest.allTasksMode = true;
oaasRequest.filterFieldAPIName = null;
oaasRequest.start = start;
oaasRequest.finish = finish;
oaasRequest.includeServicesWithEmptyLocation = false;
oaasRequest.locations = lstServiceTerritories;
oaasRequest.schedulingPolicyID = 'a0N4E0000031HKkUAM';

FSL.OAAS oaas = new FSL.OAAS();
id optRequest = oaas.optimize(oaasRequest);
```

reshuffle(serviceId, policyId)

Returns the optimization request ID of a reshuffle operation that has begun on a given service appointment.

Signature

```
public static Id reshuffle(Id serviceId, Id policyId)
```

Parameters

*serviceId*Type: [Id](#)

The record ID of the service appointment that must be scheduled.

*policyId*Type: [Id](#)

The record ID of the scheduling policy being used to schedule the service appointment.

Return Value

Type: [Id](#)

Record ID of the optimization request.

Usage

The Reshuffle action is used when a high-priority service appointment must be scheduled within a full schedule. It runs a “mini-optimization” that attempts to reshuffle the schedule to accommodate the appointment.

To learn more about the Reshuffle action, see [Reschedule Service Appointments](#).

Example

```
FSL.OAAS oaas = new FSL.OAAS();
id optRequest = oaas.reshuffle('08p4E000000M21CQAS', 'a0N4E0000031HKkUAM');
```

resourceDayOptimization(resourceId, policyId, horizon, includeAllTasks, includeOnlyResourceFutureSA, radius, candidateSasFields, unschedulableServicesField, maxOptRuntime)

Returns the optimization request ID with the details of the optimization call. This method runs resource schedule optimization—the optimization of a single service resource’s schedule—using the parameters provided.

Signature

```
public static Id resourceDayOptimization(Id resourceId, Id policyId, FSL.TimeInterval
horizon, Boolean includeAllTasks, Boolean includeOnlyResourceFutureSA, Decimal radius,
String candidateSasFields, String unschedulableServicesField, Decimal maxOptRuntime)
```

Parameters

*resourceId*Type: [Id](#)

The record ID of the service resource whose schedule is being optimized.

*policyId*Type: [Id](#)

The record ID of the scheduling policy being used to schedule the service appointment.

*horizon*Type: [FSL.TimeInterval](#)

The time frame used to run resource schedule optimization.

*includeAllTasks*Type: [Boolean](#)

If **true**, all relevant service appointments within the time frame are considered during optimization. If **false**, only unscheduled service appointments are considered, and all scheduled service appointments are pinned (unmovable).

*includeOnlyResourceFutureSA*Type: [Boolean](#)

If **true**, only service appointments that are already assigned to the service resource are considered during optimization. If **false**, service appointments assigned to other service resources are also considered.

radius

Type: [Decimal](#)

The suggested distance between required service appointments and adjacent appointments. Required appointments are defined by the `unschedulableServicesField` parameter. Resource schedule optimization tries to group service appointments so that their distance from a required appointment is less than this radius. Appointments that are further from a required appointment can still be scheduled, but are deprioritized. If **null**, the data is not filtered based on the radius.

candidateSasFields

Type: [String](#)

Boolean service appointment field that indicates which appointments are candidates to be scheduled.

unschedulableServicesField

Type: [String](#)

Boolean service appointment field that indicates whether a service appointment is required (pinned), meaning it must remain on the schedule during resource schedule optimization.

maxOptRuntime

Type: [Decimal](#)

Total time in seconds during which the optimization results must be returned. This parameter only enforces the optimization time, and does not include optimization queue or Apex job queue times. If **null**, the default value of 30 seconds is used.

Usage

To learn more about resource schedule optimization, see [Optimize a Single Resource's Schedule](#).

Return Value

Type: [Id](#)

Record ID of the optimization request.

resourceDayOptimization(resourceId, policyId, horizon, includeAllTasks, includeOnlyResourceFutureSA, radius, candidateSasFields, unschedulableServicesField, maxOptRuntime, nowTimeOnSchedule)

Returns the optimization request ID with the details of the optimization call. This method runs resource schedule optimization—the optimization of a single service resource's schedule—using the parameters provided.

Signature

```
public static Id resourceDayOptimization(Id resourceId, Id policyId, FSL.TimeInterval
horizon, Boolean includeAllTasks, Boolean includeOnlyResourceFutureSA, Decimal radius,
String candidateSasFields, String unschedulableServicesField, Decimal maxOptRuntime,
Datetime nowTimeOnSchedule)
```

Parameters

resourceId

Type: [Id](#)

The record ID of the service resource whose schedule is being optimized.

policyId

Type: [Id](#)

The record ID of the scheduling policy being used to schedule the service appointment.

horizon

Type: [FSL.TimeInterval](#)

The time frame used to run resource schedule optimization.

includeAllTasks

Type: [Boolean](#)

If [true](#), all relevant service appointments within the time frame are considered during optimization. If [false](#), only unscheduled service appointments are considered, and all scheduled service appointments are pinned (unmovable).

includeOnlyResourceFutureSA

Type: [Boolean](#)

If [true](#), only service appointments that are already assigned to the service resource are considered during optimization. If [false](#), service appointments assigned to other service resources are also considered.

radius

Type: [Decimal](#)

The suggested distance between required service appointments and adjacent appointments. Required appointments are defined by the `unschedulableServicesField` and `nowTimeOnSchedule` parameters. Resource schedule optimization tries to group service appointments so that their distance from a required appointment is less than this radius. Appointments that are further from a required appointment can still be scheduled, but are deprioritized. If [null](#), the data is not filtered based on the radius.

candidateSasFields

Type: [String](#)

Boolean service appointment field that indicates which appointments are candidates to be scheduled.

unschedulableServicesField

Type: [String](#)

Boolean service appointment field that indicates whether a service appointment is required (pinned), meaning it must remain on the schedule during resource schedule optimization.

maxOptRuntime

Type: [Decimal](#)

Total time in seconds during which the optimization results must be returned. This parameter only enforces the optimization time, and does not include optimization queue or Apex job queue times. If [null](#), the default value of 30 seconds is used.

nowTimeOnSchedule

Type: [Datetime](#)

An appointment whose scheduled start time is earlier than this time is considered required and isn't updated during resource schedule optimization. Because resource schedule optimization is asynchronous, this parameter indicates when optimization was initiated. For example, if `nowTimeOnSchedule` is set to April 17, 2018, 10:30, appointments with an earlier scheduled start time are considered required and excluded from resource schedule optimization.

Return Value

Type: [Id](#)

Record ID of the optimization request.

Usage

To learn more about resource schedule optimization, see [Optimize a Single Resource's Schedule](#).

Example

To use this code sample, replace the ID placeholders—for example, *Service Appointment ID*—with record IDs from your org. Surround the IDs with single quotes: `'08p4E00000017Gq'`.

```
FSL.OAAS a = new FSL.OAAS();

//SET the horizon interval
DateTime start = DateTime.newInstanceGmt(DateTime.Now().dateGmt(),
Time.newInstance(0,0,0,0));
DateTime finish = start.addDays(3);
FSL.TimeInterval horizon = new FSL.TimeInterval(start,finish);

//SELECT the candidatesServices
List<ServiceAppointment> services = [SELECT Id FROM ServiceAppointment WHERE Id IN
    (Service Appointment ID,Service Appointment ID) ];
SET<Id> candidatesIds = new SET<Id>();
FOR(ServiceAppointment service : services) {
    candidatesIds.add(service.Id);
}

//SET the RSO required appointment services
Set<Id> requiredSaIds = new Set<Id>();
List<ServiceAppointment> services2 = [SELECT Id FROM ServiceAppointment WHERE Id=Service
Appointment ID];
FOR(ServiceAppointment service : services2) {
    requiredSaIds.add(service.Id);
}

//START the RSO process
Id requestId = a.resourceDayOptimization(Service Resource ID,Scheduling Policy ID,
horizon,false,true,50,candidatesIds,requiredSaIds,60,DateTime.newInstance(2018,1,0,0,0,0));
```

resourceDayOptimization(resourceId, policyId, horizon, includeAllTasks, includeOnlyResourceFutureSA, radius, candidateSas, unschedulableServices, maxOptRuntime)

Returns the optimization request ID with the details of the optimization call. This method runs resource schedule optimization—the optimization of a single service resource's schedule—using the parameters provided.

Signature

```
public static Id resourceDayOptimization(Id resourceId, Id policyId, FSL.TimeInterval
horizon, Boolean includeAllTasks, Boolean includeOnlyResourceFutureSA, Decimal radius,
Set<String> candidateSas, Set<String> unschedulableServices, Decimal maxOptRuntime)
```

Parameters

*resourceId*Type: [Id](#)

The record ID of the service resource whose schedule is being optimized.

*policyId*Type: [Id](#)

The record ID of the scheduling policy being used to schedule the service appointment.

*horizon*Type: [FSL.TimeInterval](#)

The time frame used to run resource schedule optimization.

*includeAllTasks*Type: [Boolean](#)

If [true](#), all relevant service appointments within the time frame are considered during optimization. If [false](#), only unscheduled service appointments are considered, and all scheduled service appointments are pinned (unmovable).

*includeOnlyResourceFutureSA*Type: [Boolean](#)

If [true](#), only service appointments that are already assigned to the service resource are considered during optimization. If [false](#), service appointments assigned to other service resources are also considered.

*radius*Type: [Decimal](#)

The suggested distance between required service appointments and adjacent appointments. Required appointments are defined by the *unschedulableServices* parameter. Resource schedule optimization tries to group service appointments so that their distance from a required appointment is less than this radius. Appointments that are further from a required appointment can still be scheduled, but are deprioritized. If [null](#), the data is not filtered based on the radius.

*candidateSas*Type: [Set<Id>](#)

Set of IDs of service appointments that are candidates for scheduling.

*unschedulableServices*Type: [Set<Id>](#)

Set of IDs of service appointments that are required (pinned), meaning they must remain on the schedule during resource schedule optimization

*maxOptRuntime*Type: [Decimal](#)

Total time in seconds during which the optimization results must be returned. This parameter only enforces the optimization time, and does not include optimization queue or Apex job queue times. If [null](#), the default value of 30 seconds is used.

Usage

To learn more about resource schedule optimization, see [Optimize a Single Resource's Schedule](#).

Return Value

Type: [Id](#)

Record ID of the optimization request.

resourceDayOptimization(resourceId, policyId, horizon, includeAllTasks, includeOnlyResourceFutureSA, radius, candidateSas, unschedulableServices, maxOptRuntime, nowTimeOnSchedule)

Returns the optimization request ID with the details of the optimization call. This method runs resource schedule optimization—the optimization of a single service resource’s schedule—using the parameters provided.

Signature

```
public static Id resourceDayOptimization(Id resourceId, Id policyId, FSL.TimeInterval
horizon, Boolean includeAllTasks, Boolean includeOnlyResourceFutureSA, Decimal radius,
Set<String> candidateSas, Set<String> unschedulableServices, Decimal maxOptRuntime,
Datetime nowTimeOnSchedule)
```

Parameters

resourceId

Type: [Id](#)

The record ID of the service resource whose schedule is being optimized.

policyId

Type: [Id](#)

The record ID of the scheduling policy being used to schedule the service appointment.

horizon

Type: [FSL.TimeInterval](#)

The time frame used to run resource schedule optimization.

includeAllTasks

Type: [Boolean](#)

If **true**, all relevant service appointments within the time frame are considered during optimization. If **false**, only unscheduled service appointments are considered, and all scheduled service appointments are pinned (unmovable).

includeOnlyResourceFutureSA

Type: [Boolean](#)

If **true**, only service appointments that are already assigned to the service resource are considered during optimization. If **false**, service appointments assigned to other service resources are also considered.

radius

Type: [Decimal](#)

The suggested distance between required service appointments and adjacent appointments. Required appointments are defined by the *unschedulableServices* and *nowTimeOnSchedule* parameters. Resource schedule optimization tries to group service appointments so that their distance from a required appointment is less than this radius. Appointments that are further from a required appointment can still be scheduled, but are deprioritized. If **null**, the data is not filtered based on the radius.

candidateSas

Type: [Set<Id>](#)

Set of IDs of service appointments that are candidates for scheduling.

unschedulableServices

Type: [Set<Id>](#)

Set of IDs of service appointments that are required, meaning they must remain on the schedule during resource schedule optimization. These appointments may be moved to a different time slot, but they will continue to comply with their Earliest Start Permitted and Due Date values.

maxOptRuntime

Type: [Decimal](#)

Total time in seconds during which the optimization results must be returned. This parameter only enforces the optimization time, and does not include optimization queue or Apex job queue times. If **null**, the default value of 30 seconds is used.

nowTimeOnSchedule

Type: [Datetime](#)

Optional. An appointment whose scheduled start time is earlier than this time is considered required and isn't updated during resource schedule optimization. Because resource schedule optimization is asynchronous, this parameter indicates when optimization was initiated. For example, if *nowTimeOnSchedule* is set to April 17, 2018, 10:30, appointments with an earlier scheduled start time are considered required and excluded from resource schedule optimization.

Return Value

Type: [Id](#)

Record ID of the optimization request.

Usage

To learn more about resource schedule optimization, see [Optimize a Single Resource's Schedule](#).

Example

```
FSL.OAAS a = new FSL.OAAS();

//SET the horizon interval
DateTime start = DateTime.newInstanceGmt(DateTime.Now().dateGmt(),
Time.newInstance(0,0,0,0));
DateTime finish = start.addDays(3);
FSL.TimeInterval horizon = new FSL.TimeInterval(start,finish);

//SELECT the candidatesServices
List<ServiceAppointment> services = [SELECT Id FROM ServiceAppointment WHERE Id IN
('08p4E00000017Gp','08p4E00000017Go') ];
SET<Id> candidatesIds = new SET<Id>();
FOR(ServiceAppointment service : services) {
    candidatesIds.add(service.Id);
}

//SET the RSO required appointment services
Set<Id> requiredSaIds = new Set<Id>();
List<ServiceAppointment> services2 = [SELECT Id FROM ServiceAppointment WHERE
Id='08p4E00000017Gq'];
FOR(ServiceAppointment service : services2) {
    requiredSaIds.add(service.Id);
}

//START the RSO process
```

```
Id requestId =
a.resourceDayOptimization('0Hn4E0000004JRS','a1w4E000000Ac6S',horizon,false,true,50,
    candidatesIds,requiredSaIds,60,DateTime.newInstance(2018,1,0,0,0,0));
```

OAASRequest Class

Represents the request sent in the `OAAS.optimize()` method. This class contains all the details of a global optimization call.

Namespace

FSL

IN THIS SECTION:

[OAASRequest Properties](#)

OAASRequest Properties

`OAASRequest` contains the following properties.

IN THIS SECTION:

[allTasksMode](#)

If `true`, all service appointments within the time frame are considered during optimization. If `false`, only unscheduled appointments are considered, and scheduled appointments are pinned (unmovable).

[filterFieldAPIName](#)

The API name of a Boolean field on the ServiceAppointment object. Only service appointments on which the Boolean field is set to `true` are optimized.

[start](#)

The start of the time frame in which the optimization process considers data for scheduling.

[finish](#)

The end of the time frame in which the optimization process considers data for scheduling.

[includeServicesWithEmptyLocation](#)

If `true`, service appointments both with and without an associated service territory are considered during optimization. If `false`, only appointments with an associated service territory are considered.

[keepApptScheduled \(Beta\)](#)

The API name of a Boolean field on the ServiceAppointment object. Only service appointments on which the Boolean field is set to `true` are never dropped from the schedule after global or in-day optimization. `keepApptScheduled` is only available for Enhanced Scheduling and Optimization.

[locations](#)

A list of IDs of service territories for which optimization will run.

[numberOfServicesToSchedule](#)

Not currently in use. Leave as `null`.

[schedulingPolicyId](#)

The record ID of the scheduling policy that is used to schedule the service appointment.

allTasksMode

If **true**, all service appointments within the time frame are considered during optimization. If **false**, only unscheduled appointments are considered, and scheduled appointments are pinned (unmovable).

Signature

```
public Boolean allTasksMode {get; set;}
```

Property Value

Type: [Boolean](#)

filterFieldAPIName

The API name of a Boolean field on the ServiceAppointment object. Only service appointments on which the Boolean field is set to **true** are optimized.

Signature

```
public String filterFieldAPIName {get; set;}
```

Property Value

Type: [String](#)

start

The start of the time frame in which the optimization process considers data for scheduling.

Signature

```
public Datetime start {get; set;}
```

Property Value

Type: [Datetime](#)

finish

The end of the time frame in which the optimization process considers data for scheduling.

Signature

```
public Datetime finish {get; set;}
```

Property Value

Type: [Datetime](#)

includeServicesWithEmptyLocation

If `true`, service appointments both with and without an associated service territory are considered during optimization. If `false`, only appointments with an associated service territory are considered.

Signature

```
public Boolean includeServicesWithEmptyLocation {get; set;}
```

Property Value

Type: [Boolean](#)

keepApptScheduled (Beta)

The API name of a Boolean field on the ServiceAppointment object. Only service appointments on which the Boolean field is set to `true` are never dropped from the schedule after global or in-day optimization. `keepApptScheduled` is only available for Enhanced Scheduling and Optimization.

Signature

```
public String keepApptScheduled {get; set;}
```

Property Value

Type: [String](#)

Usage

Use this property to ensure that appointments are optimized but not removed from the schedule. They can be moved to another resource or time slot.

There are four possible property value options:

- Valid field name: the field name is a valid Boolean field on the ServiceAppointment object. Service appointments that are marked as "Keep Scheduled" won't be dropped from the schedule after global or in-day optimization.
- Invalid field name: the field doesn't exist on the Service Appointment object or isn't a Boolean type. An error message appears.
- Null: Service appointments are marked "Keep Scheduled" based on the org-level defaults defined in the global or in-day optimization settings.+
- `NO_KEEP_APPT_SCHEDULED`: Service appointments aren't marked as "Keep Scheduled".

locations

A list of IDs of service territories for which optimization will run.

Signature

```
public List<Id> locations {get; set;}
```

Property Value

Type: [List<Id>](#)

numberOfServicesToSchedule

Not currently in use. Leave as `null`.

Signature

```
public Integer numberOfServicesToSchedule {get; set;}
```

Property Value

Type: [Integer](#)

schedulingPolicyId

The record ID of the scheduling policy that is used to schedule the service appointment.

Signature

```
public Id schedulingPolicyId {get; set;}
```

Property Value

Type: [Id](#)

PolygonUtils Class

Represents the result of using the `getTerritoryIdByPolygons` method to return a service territory record ID using latitude and longitude geolocation values.

Namespace

[FSL](#)

Usage

Before calling the `PolygonUtils` class, make sure that you have existing map polygon records that are associated with a service territory.

If **Base service appointment territories on polygons** is enabled, global actions consume this class to populate the Service Territory field on service appointments. When standard FSL Global Actions are not being used, you can use the `getTerritoryIdByPolygons` method to populate the Service Territory lookup field on objects.

If `getTerritoryIdByPolygons` doesn't find a territory it returns `null`.

For more information, see [Create and Manage Map Polygons](#).

IN THIS SECTION:

[PolygonUtils Methods](#)

PolygonUtils Methods

`PolygonUtils` includes the following static method.

IN THIS SECTION:

[getTerritoryIdByPolygons\(longitude, latitude\)](#)

Returns a service territory record ID from longitude and latitude values.

[getAllPolygonsByLatLong\(longitude, latitude\)](#)

Returns a list of IDs of all matching polygons that include the point with the longitude and latitude values.

getTerritoryIdByPolygons(longitude, latitude)

Returns a service territory record ID from longitude and latitude values.

Signature

```
static Id getTerritoryIdByPolygons(Double longitude, Double latitude)
```

Parameters

longitude

Type: [Double](#)

Required. The longitude value of the geolocation to be used to retrieve a service territory record ID via Polygon Classification.

latitude

Type: [Double](#)

Required. The latitude value of the geolocation to be used to retrieve a service territory record ID via Polygon Classification.

Return Value

Type: [Id](#)

Usage

This method returns a service territory record ID from longitude and latitude values. If a service appointment's geolocation matches more than one polygon, the appointment is assigned to either the highest or lowest-level territory in the hierarchy depending on your settings.

Example

This example retrieves a service territory record ID using latitude and longitude geolocation fields.

```
//fetch Service Appointment record(s)
List<ServiceAppointment> sas = [select id, latitude, longitude from serviceappointment
where appointmentnumber = 'SA-3600' limit 1];
If(!sas.isEmpty()){
    //retrieve the service territory record id from the getTerritoryIdByPolygons method
    using geolocation fields from the first appointment in the list
    Id ServiceTerritoryId =
    FSL.PolygonUtils.getTerritoryIdByPolygons(double.valueOf(sas[0].longitude),double.valueOf(sas[0].latitude));

    system.debug(ServiceTerritoryId);
}
```

getAllPolygonsByLatLong(longitude, latitude)

Returns a list of IDs of all matching polygons that include the point with the longitude and latitude values.

Signature

```
static List getAllPolygonsByLatLong(Double longitude, Double latitude)
```

Parameters

longitude

Type: [Double](#)

Required. The longitude value of the geolocation to be used to retrieve a service territory record ID using polygon classification.

latitude

Type: [Double](#)

Required. The latitude value of the geolocation to be used to retrieve a service territory record ID using polygon classification.

Return Value

Type: [List](#) <Id>

Usage

This method returns a list of polygon record IDs from longitude and latitude values. The method doesn't include sharing.

Example

This example retrieves a list of relevant polygons that include the point with the longitude and latitude values.

```
//fetch Service Appointment record(s)
Double longitude = 32.077213;
Double latitude = 34.792759;
List<FSL__Polygon__c> relevantPolygons =
FSL.PolygonUtils.getAllPolygonsByLatLong
(longitude,latitude);
system.debug(relevantPolygons);
```

RecurringAppointmentsManager Class

Represents the result of using the [getRecurringAppointmentSlots](#) on page 604 method to return a pattern of appointments that recur weekly.

Namespace

[FSL](#)

Usage

`RecurringAppointmentsManager` is an Apex class that takes the [RecurringPattern](#) on page 606 class as a required parameter, and returns a list of [RecurringSequence](#) on page 608 appointments that recur weekly.

IN THIS SECTION:

[RecurringAppointmentsManager Methods](#)

RecurringAppointmentsManager Methods

`RecurringAppointmentsManager` includes the following static method.

IN THIS SECTION:

[getRecurringAppointmentsSlots](#)

Returns a list of [RecurringSequence](#) on page 608 appointments.

getRecurringAppointmentsSlots

Returns a list of [RecurringSequence](#) on page 608 appointments.

Signature

```
static List getRecurringAppointmentsSlots
```

Parameters

ServiceID

Type: Id

Required. The ID of a service appointment that represents a recurring visit. This record defines the scheduling requirements and constraints for each appointment in the recurring visits sequence. All visits use the location, required resources, skill requirements, and other constraints that are associated with this service appointment.

PolicyID

Type: Id

Required. The policy to be used to get the relevant work rules and objectives for the operation.

CalendarOperatingHoursId

Type: Id

Required. The ID of the operating hours used to determine the structure of the slots returned by the API.

RecurringPattern

Type: RecurringPattern Class

Required. This class instance is sent as a parameter that contains all the required details from the recurring pattern.

SchedulingOptionsCount

Type: Integer

Required. Indicates how many sets of scheduling options the API returns. You can have between one and three sets of options for the recurring appointment. If you select three, but fewer than three sets of recurrence scheduling options are available, you receive however many options are found given your constraints.



Important: As of API version 60.0, *SchedulingOptionsCount* is hard coded to 3.

Return Value

Type: [List](#) [RecurringSequence](#)

Usage

This method returns a list of recurring appointments to repeat weekly.

This method only returns a list of potential scheduling options in the [RecurringSequence](#) on page 608 class and doesn't schedule the recurring appointments. Similar to the `getSlots` method for [AppointmentBookingService](#) on page 569, you use the response from `getRecurringAppointmentsSlots` to create a list of records to insert, and then schedule the appointments.

The time constraint fields on `ServiceAppointment` that you pass into the API are used for all appointments within the recurrence and determine the availability of scheduling options. Ensure that there's a significant difference between `EarliestStartTime` and `DueDate` fields. For example, if you must schedule six appointments, and your customer only has visiting hours on Monday, you need six weeks to schedule all visits. Therefore, make sure you set `EarliestStartTime` and `DueDate` to at least six weeks apart. Don't set other time constraint fields on the appointment that are more restrictive, such as **`ArrivalWindowStartTime`** and **`ArrivalWindowEndTime`**, which can further limit scheduling options.

If you need service to occur within a given time frame, for example 8 AM to 10 AM on a Monday, set the `DaysOfWeek` to Monday. Then, set the time frame using the visiting hours record that has operating hours time slots during your required time frame.

See the [ServiceAppointment](#) on page 299 object reference for more information on time constraint fields.

Example 1: Using an Execution Script

The following code sample creates an instance of the [RecurringPattern](#) on page 606 class. Then, it sets the properties for that recurring pattern. In this case, the code creates a recurring pattern that excludes weekends, and considers that the customer isn't available on Tuesdays, needs two visits per week, and requires six total visits. It requests that the API return three scheduling options.

```
//Fill the pattern object
FSL.RecurringPattern pattern = new FSL.RecurringPattern();
pattern.DaysOfWeek = new
Set<FSL.RecurringPattern.DaysOfWeek>{FSL.RecurringPattern.DaysOfWeek.Monday,
FSL.RecurringPattern.DaysOfWeek.Wednesday, FSL.RecurringPattern.DaysOfWeek.Thursday,
FSL.RecurringPattern.DaysOfWeek.Friday};
pattern.FrequencyType = FSL.RecurringPattern.FrequencyType.WEEKLY;
pattern.Frequency = 2;
pattern.NumberOfVisits = 6;

Integer schedulingOptionsCount = 3;
Id policyID = 'a1Lx00000004CUXEA2';
Id serviceId = '08px000000NzmOeAAJ';
Id operatingHoursId = '0OHx0000000D3ETGA0';

//Call the API
FSL.RecurringAppointmentSlots result =
FSL.RecurringAppointmentsManager.getRecurringAppointmentsSlots(serviceId , policyID,
operatingHoursId, schedulingOptionsCount, pattern);

//Handle the response of the API (example only)
for (Integer i=1 ; i<= result.recurringSequences.size() ; i++){
    System.debug('Sequence number: ' + i);
    System.debug('participatingResources details: /n' + result);
    System.debug('visitSchedulingOptions details: /n' + result);
}
```

```

System.debug('averageObjectivesGrades details: /n' + result);
System.debug('sequenceScore details: /n' + result);
System.debug('firstPatternOccurrence details: /n' + result);
}

```

Example 2: Using a Method

This code sample uses a method to create the same recurring pattern.

```

//Using a method (example):
public callRecurringVisitsAPI(Id serviceID, Id policyID, Id calendarOperatingHoursId,
Integer SchedulingOptionsCount) {

//Fill the pattern object
RecurringPattern pattern = new FSL.RecurringPattern();
pattern.DaysOfWeek = new Set<RecurringPattern.DaysOfWeek>{RecurringPattern.DaysOfWeek.Monday,
RecurringPattern.DaysOfWeek.Wednesday, RecurringPattern.DaysOfWeek.Thursday,
RecurringPattern.DaysOfWeek.Friday};

pattern.FrequencyType = RecurringPattern.FrequencyType.WEEKLY;
pattern.Frequency = 2;
pattern.NumberOfVisits = 6;

//Call the API
FSL.RecurringAppointmentSlots result =
RecurringAppointmentsManager.getRecurringAppointmentSlots(serviceID, policyID,
calendarOperatingHoursId, SchedulingOptionsCount, pattern);
}

}

```

RecurringPattern Class

Represents a global sharing class that is a required parameter for [RecurringAppointmentsManager](#) on page 603 to return a pattern of appointments that recur weekly.

Namespace

[FSL](#)

Usage

[RecurringPattern](#) is a global sharing class that is a required parameter for [RecurringAppointmentsManager](#) on page 603. All properties must be complete to call the [RecurringAppointmentsManager](#) API.

Example

This example shows code for [RecurringPattern](#) and its properties.

```

global class RecurringPattern {
    global enum DaysOfWeek {Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday}
}

```

```

global enum FrequencyType {DAILY, WEEKLY, MONTHLY}
global Set<DaysOfWeek> DaysOfWeek { get; set; }
global FrequencyType FrequencyType { get; set; }
global Integer Frequency { get; set; }
global Integer NumberOfVisits { get; set; }
}

```

IN THIS SECTION:

[Recurring Pattern Properties](#)

Recurring Pattern Properties

`RecurringPattern` contains the following properties.

IN THIS SECTION:

[DaysOfWeek](#)

The list of the days that can be used to book appointments, which can be derived from the visiting hours record associated with the service appointment..

[Frequency](#)

The number of appointments to book inside the `FrequencyType` time frame provided.

[FrequencyType](#)

The frequency of recurring appointments. Currently, only weekly frequency is supported.

[NumberOfVisits](#)

Total number of appointments to schedule.

DaysOfWeek

The list of the days that can be used to book appointments, which can be derived from the visiting hours record associated with the service appointment..

Signature

```

global Set<DaysOfWeek> DaysOfWeek { get; set; }
global enum<FSL.RecurringPattern> DaysOfWeek {Sunday, Monday, Tuesday, Wednesday,
Thursday, Friday, Saturday}

```

Property Value

Type: [Set<enum>](#)

Usage

You can set `DaysofWeek` manually. Alternately, you can derive the days the customer is available from the visiting hours record associated with the work order that is associated with the service appointment passed into the API.

Only one visit should be scheduled per day. For example, if you are scheduling a total of six visits and set `DaysOfWeek` to Monday and Wednesday, the API doesn't provide a response with more than one visit scheduled for Monday and one visit scheduled for Wednesday.

Frequency

The number of appointments to book inside the `FrequencyType` time frame provided.

Signature

```
global Integer Frequency { get; set; }
```

Property Value

Type: [Integer](#)

FrequencyType

The frequency of recurring appointments. Currently, only weekly frequency is supported.

Signature

```
global FrequencyType FrequencyType { get; set; }
```

Property Value

Type: [Enum](#)

NumberOfVisits

Total number of appointments to schedule.

Signature

```
global Integer NumberOfVisits { get; set; }
```

Property Value

Type: [Integer](#)

RecurringSequence Class

Represents the result of using the [getRecurringAppointmentsSlots](#) on page 604 method to return a pattern of appointments that recur daily, weekly, or monthly.

Namespace

[FSL](#)

Example

This example shows the contents of a returned `RecurringSequence` object.

```
global class RecurringAppointmentSlots {
    global ServiceAppointment rootServiceAppointment;
    global List<RecurringSequence> recurringSequences;
}

global class RecurringSequence {
    global List<EncapsulatedResource> participatingResources;
    global List<SchedulingOption> visitSchedulingOptions;
    global List<GradeSlotResult> averageObjectivesGrades;
    global Double sequenceScore;
    global List<TimeInterval> firstPatternOccurrence;
}
```

ResourceScheduleData Class

Contains all the details of a service resource's available slots for a given service appointment. An instance of this class is used only within the context of the results returned by the `AdvancedGapMatrix` class.

Namespace

[FSL](#)

Usage

For more information, see [AdvancedGapMatrix](#).

IN THIS SECTION:

[ResourceScheduleData Properties](#)

ResourceScheduleData Properties

`ResourceScheduleData` contains the following properties.

IN THIS SECTION:

[currentSlotsIndexInAB](#)

Index of the slot.

[resource](#)

Not currently in use.

[schedulingOptions](#)

A list of all possible scheduling options for the service resource.

currentSlotsIndexInAB

Index of the slot.

Signature

```
public Integer currentSlotsIndexInAB {get; set;}
```

Property Value

Type: [Integer](#)

resource

Not currently in use.

Signature

```
public FSL.EncapsulatedResource resource {get; set;}
```

Property Value

Type: [FSL.EncapsulatedResource](#)

schedulingOptions

A list of all possible scheduling options for the service resource.

Signature

```
public List<FSL.SchedulingOption> schedulingOptions {get; set;}
```

Property Value

Type: [List<FSL.SchedulingOption>](#)

ScheduleJobsApi Class

Represents configurations to the Territory tab that is available for all scheduled jobs that are categorized under Enhanced Optimization.

Namespace

[FSL](#)

Usage

This API uses two methods, `getJob` and `setTerritory`. `getJob` passes a job to `setTerritory`, and `setTerritory` applies a `territoryID` and `groupPolicyID` to that job via [GroupData](#) on page 587.

IN THIS SECTION:

[ScheduleJobsApi Methods](#)

ScheduleJobsApi Methods

`ScheduleJobsApi` includes the following static methods.

IN THIS SECTION:

[getJob](#)

Passes a scheduled job to the `setTerritory` method, which modifies the territories.

[setTerritory](#)

Sets territories for a job returned by `getJob` and performs various validation checks.

getJob

Passes a scheduled job to the `setTerritory` method, which modifies the territories.

Signature

```
global static ScheduleJobsApi getJob(String jobName)
```

Parameters

jobName

Type: [String](#)

Required. The name of the job to update for service territory. The job name is case-sensitive.

Return Value

Type: `ScheduleJobsApi(validateJobId(jobId))`

Usage

This method passes a job to [setTerritory](#) on page 611, which updates the Territory tab. See [setTerritory](#) for sample code.

setTerritory

Sets territories for a job returned by `getJob` and performs various validation checks.

Signature

```
public void setTerritory(List<GroupData> groupList)
```

Parameters

groupList

Type: `List<GroupData>`

Required. A list of `GroupData` objects representing the territories and group policies to be associated with the job.

Return Value

Type: `Void`

Usage

This method configures the Territory tab based on optimized territory and scheduling policy for a scheduled job. Here are two examples of code that configures the territory tab for a scheduled job.

Example 1:

```
String jobName = 'Optimization';

List<FSL.ScheduleJobsApi.GroupData> groupList = new List<FSL.ScheduleJobsApi.GroupData>();

FSL.ScheduleJobsApi.GroupData firstGroup = new FSL.ScheduleJobsApi.GroupData
    ('a0cSM0000000fEU', new List<String>{'0HhSM0000000S5x', '0HhSM0000000TbV0AU'});

FSL.ScheduleJobsApi.GroupData secondGroup = new FSL.ScheduleJobsApi.GroupData
    ('0', new List<String>{'0HhSM0000000RbJ'});

groupList.add(firstGroup);
groupList.add(secondGroup);

FSL.ScheduleJobsApi.GetJob(jobName).setTerritory(groupList);
```

Example 2:

```
String jobName = "Optimization";

// Get all service territory groups.
List<YourGroupType> All_ST_GROUPS = getAllGroupsFromAnotherSource();

List<FSL.ScheduleJobsApi.GroupData> groupList = new List<FSL.ScheduleJobsApi.GroupData>();

for (Integer i = 0; i < All_ST_GROUPS.size(); i++) {
    FSL.ScheduleJobsApi.GroupData groupData = new FSL.ScheduleJobsApi.GroupData(
        All_ST_GROUPS[i].policyId,
        new List<String> {
            All_ST_GROUPS[i].territoryIds
        }
    );
    // Add the created groupData to the list.
    groupList.add(groupData);
}

FSL.ScheduleJobsApi.GetJob(jobName).setTerritory(groupList);
```

ScheduleResult Class

Represents the result of scheduling a service appointment that is returned by the `ScheduleService.schedule` method.

Namespace

[FSL](#)

Example

This example shows the contents of a returned `ScheduledResult` object.

```
[
  BreakObject=null,
  Grade=100.000000000000000000000000000000,
  LongOperationId=null,
  PartialResults=(),
  Resource=ServiceResource:
  {Id=0Hn4E0000004bucSAA, Name=Crew A, IsActive=true, IsCapacityBased=false, ResourceType=C,
  ServiceCrewId=1cr4E0000004CM7QAM, Contractor_Priority__c=2},
  Service=ServiceAppointment:
  {Id=08p4E000000LeMiQAK, Status=None, FSL__Same_Day__c=false, FSL__Same_Resource__c=false,
  AppointmentNumber=SA-6214, DueDate=2018-09-20 17:07:00, EarliestStartTime=2018-09-17
  17:07:00, Duration=1.5, DurationType=Hours, Latitude=37.793872000000000,
  Longitude=-122.394865000000000, FSL__InternalSLRGeolocation__Latitude__s=37.793834,
  FSL__InternalSLRGeolocation__Longitude__s=-122.395123, ServiceTerritoryId=0Hh4E00000000tPSAU,
  FSL__Schedule_over_lower_priority_appointment__c=false, FSL__Use_Async_Logic__c=false,
  FSL__IsMultiDay__c=false, ParentRecordId=0WO4E000000V16rWAC, FSL__Emergency__c=false,
  SchedStartTime=2018-09-17 17:13:00, SchedEndTime=2018-09-17 18:43:00,
  FSL__Schedule_Mode__c=Automatic, FSL__Scheduling_Policy_Used__c=a0N4E0000031HPVUA2}
]
```

IN THIS SECTION:

ScheduleResult Properties

ScheduleResult Properties

`ScheduleResult` contains the following properties.

IN THIS SECTION:

breakObject

Not currently in use.

grade

The grade of the slot in which the service appointment was scheduled.

longOperationId

This field only applies to scheduling service appointments that don't use Enhanced Scheduling and Optimization (ESO). Populated when scheduling a group of two dependent service appointments, also known as complex work. An asynchronous call is made to schedule both service appointments in the dependency group. This property contains the `FSL_FSL_Operation_c` ID.

partialResults

Relevant only when Limit Apex Operations is enabled in the Field Service Admin app. If the processing time of the `ScheduleService` is about to exceed the maximum CPU limits, `partialResults` returns the results that were already calculated and halts the process. This list contains the details of the partial, processed results.

serviceResource

The service resource assigned to the service appointment.

serviceAppointment

The scheduled service appointment.

breakObject

Not currently in use.

Signature

```
public ResourceAbsence breakObject {get; set;}
```

Property Value

Type: ResourceAbsence

grade

The grade of the slot in which the service appointment was scheduled.

Signature

```
public Decimal grade {get; set;}
```

Property Value

Type: [Decimal](#)

longOperationId

This field only applies to scheduling service appointments that don't use Enhanced Scheduling and Optimization (ESO). Populated when scheduling a group of two dependent service appointments, also known as complex work. An asynchronous call is made to schedule both service appointments in the dependency group. This property contains the `FSL__FSL_Operation__c` ID.

Signature

```
public Id longOperationId {get; set;}
```

Property Value

Type: [Id](#)

Usage

To learn more about scheduling dependencies and complex work, see [Create Scheduling Dependencies Between Service Appointments](#).

To learn about Enhanced Scheduling and Optimization, see [Get Ready for Scheduling and Optimization](#).

partialResults

Relevant only when Limit Apex Operations is enabled in the Field Service Admin app. If the processing time of the `ScheduleService` is about to exceed the maximum CPU limits, `partialResults` returns the results that were already calculated and halts the process. This list contains the details of the partial, processed results.

Signature

```
public List<FSL.PartialResultsInfo> partialResults {get; set;}
```

Property Value

Type: [List<FSL.PartialResultsInfo>](#)

serviceResource

The service resource assigned to the service appointment.

Signature

```
public ServiceResource serviceResource {get; set;}
```

Property Value

Type: [ServiceResource](#)

serviceAppointment

The scheduled service appointment.

Signature

```
public ServiceAppointment serviceAppointment {get; set;}
```

Property Value

Type: [ServiceAppointment](#)

ScheduleService Class

Represents the scheduling of a given service appointment in the best available slot based on the applied scheduling policy.

Namespace

[FSL](#)

Usage

This class calls the scheduling engine and schedules the given service appointment in the highest-scoring available slot.

When using Enhanced Scheduling and Optimization, calling this API triggers a callout to the Enhanced Scheduling and Optimization service to retrieve results. When not using Enhanced Scheduling and Optimization, you can make a callout to obtain required travel times when:

- SLR or point-to-point predictive travel is the selected routing, and
- Results aren't stored in the local cache

We recommend that you avoid performing any DML in the same Apex transaction before calling this API.

 **Note:** When scheduling a service appointment, the user scheduling the appointment must have one of four managed package permission sets: Field Service Admin, Field Service Dispatcher, Field Service Agent, Self-Service. If you're using platform events to schedule appointments, you must explicitly configure the user so that the user has the correct permissions. Without the proper configuration, the platform event runs as the Automated Process system user and doesn't have the correct permissions to schedule an appointment. To learn more, see [Configure the User and Batch Size for Your Platform Event Trigger](#).

IN THIS SECTION:

[ScheduleService Methods](#)

ScheduleService Methods

`ScheduleService` includes the following static methods.

IN THIS SECTION:

[schedule\(policy, serviceId\)](#)

Returns `FSL.ScheduleResult` with the result of the scheduling process.

[scheduleExtended\(policy, serviceId\)](#)

Returns `List<FSL.ScheduleResult>` with the result of the scheduling process for appointments in a complex work chain.

[getAppointmentInsights](#)

Returns an [AppointmentInsightsResult](#) on page 582 class with details about why a service appointment can't be scheduled on the Gantt, including blocking rules, blocked slots, and resource availability.

schedule(policy, serviceId)

Returns `FSL.ScheduleResult` with the result of the scheduling process.

Signature

```
public static FSL.ScheduleResult schedule(Id policy, Id serviceId)
```

Parameters

policy

Type: `Id`

The record ID of the scheduling policy being used to schedule the service appointment.

serviceId

Type: `Id`

The record ID of the service appointment being scheduled.

Return Value

Type: `FSL.ScheduleResult`

Usage

This method schedules the service appointment in the best available slot. If there are no available slots, the appointment isn't scheduled. This method can be called with only one service appointment at a time. To schedule multiple service appointments, use an Apex batch class. Call this method in batches of one.

If you are using the `schedule` method with the appointment booking method, perform a time zone conversion. The results of appointment booking are returned in the time zone specified in the method signature. Convert these values back to UTC.

Example

```
// FSL.ScheduleService class
// The Schedule method returns a ScheduleResult result
FSL.ScheduleResult myResult = new FSL.ScheduleResult();

// static FSL.ScheduleResult Schedule(Scheduling Policy ID, Service Appointment ID)
myResult = FSL.ScheduleService.schedule(Scheduling Policy ID, Service Appointment ID);

System.debug(myResult);
```

scheduleExtended(policy, serviceId)

Returns `List<FSL.ScheduleResult>` with the result of the scheduling process for appointments in a complex work chain.

Signature

```
public static List<FSL.ScheduleResult> scheduleExtended(Id policy, Id serviceId)
```

Parameters

policy

Type: `Id`

The record ID of the scheduling policy that's used to schedule the service appointment.

serviceId

Type: `Id`

The record ID of a service appointment in a complex work chain of service appointments being scheduled.

Return Value

Type: `List<FSL.ScheduleResult>`

Usage

Use this method to schedule two service appointments in a complex work chain. This method respects the complex work setting **Use all-or-none scheduling for related appointments**. If there are no available slots, appointments in the complex work chain aren't scheduled.

The `scheduleExtended` method is valid for a chain of two appointments; if the appointment in *serviceId* has dependencies with more than one other appointment, scheduling results can be different than expected.

This method can be called with only one service appointment at a time, and runs asynchronously. To examine results that the asynchronous method returns, use the streaming API and subscribe to `MstCompletedChannel`, the channel for the Field Service managed package.

If Enhanced Scheduling and Optimization (ESO) is enabled:

- The method runs synchronously.
- The method always uses the ESO behavior, which is **all-or-none** for related appointments scheduling of complex work.
- The method is valid for a chain of up to five to appointments.

If you're using the `scheduleExtended` method with the appointment booking method, perform a time zone conversion. The results of appointment booking are returned in the time zone specified in the method signature. Convert these values back to UTC.

SEE ALSO:

[Example: Subscribe to and Replay Events Using a Visualforce Page](#)

getAppointmentInsights

Returns an [AppointmentInsightsResult](#) on page 582 class with details about why a service appointment can't be scheduled on the Gantt, including blocking rules, blocked slots, and resource availability.

Signature

```
public static FSL.AppointmentInsightsResult<FSL.AppointmentInsightsResult>
getAppointmentInsights(Id policyId, Id serviceAppointmentId)
```

Parameters

serviceAppointmentId

Type: [Id](#)

The ID of the appointment for which insights are being requested.

policyId

Type: [Id](#)

The ID of the policy under which the appointment is evaluated.

Return Value

Type: [FSL.AppointmentInsightsResult](#)

Example

The following code sample uses the `getAppointmentInsights` method to return an `AppointmentInsightsResult` class that provides details about a specific service appointment that can't be scheduled on the Gantt.

```
// FSL.ScheduleService class
// The getAppointmentInsights method returns a AppointmentInsightsResult result
FSL.AppointmentInsightsResult myresult = new FSL.AppointmentInsightsResult();

// static FSL.AppointmentInsightsResult
myresult = FSL.ScheduleService.getAppointmentInsights(Service Appointment ID,Scheduling
Policy ID);

System.debug(myresult);
```

SchedulingOption Class

Provides the details of an individual slot available for scheduling. Represents a run-time object used within the context of the results returned in the `AdvancedGapMatrix` class.

Namespace

[FSL](#)

IN THIS SECTION:

[SchedulingOption Properties](#)

SchedulingOption Properties

`SchedulingOption` contains the following properties.

IN THIS SECTION:

[grade](#)

The grade of the available slot.

[interval](#)

The time frame of the returned slot.

grade

The grade of the available slot.

Signature

```
public Decimal grade {get; set;}
```

Property Value

Type: [Decimal](#)

interval

The time frame of the returned slot.

Signature

```
public FSL.TimeInterval interval {get; set;}
```

Property Value

Type: [FSL.TimeInterval](#)

TimeInterval Class

Holds an interval's start and end times. An instance of this class is used when capturing the start and end times of a scheduling horizon, during which a scheduling process runs. It considers scheduled jobs and is also used to represent the start and end times of an appointment slot or window.

Namespace

[FSL](#)

IN THIS SECTION:

[TimeInterval Properties](#)

[TimeInterval Methods](#)

TimeInterval Properties

`TimeInterval` contains the following properties.

IN THIS SECTION:

[start](#)

The start time of the `TimeInterval` class.

[finish](#)

The end time of the `TimeInterval` class.

start

The start time of the `TimeInterval` class.

Signature

```
public Datetime start {get; set;}
```

Property Value

Type: [Datetime](#)

finish

The end time of the `TimeInterval` class.

Signature

```
public Datetime finish {get; set;}
```

Property Value

Type: [Datetime](#)

TimeInterval Methods

`TimeInterval` includes the following method.

IN THIS SECTION:

`toString()`

Returns the start and end datetimes as a string.

`toString()`

Returns the start and end datetimes as a string.

Signature

```
public String toString()
```

Return Value

Type: `String`

Field Service Custom Triggers

Field Service custom triggers ensure that the objects and respective fields are processed before or after scheduling.

The Field Service package provides triggers on Salesforce objects. Each trigger checks for various conditions and then performs tasks based on what it finds. Some of the triggers run on custom objects and some on standard objects.

Table 6: Field Service Quick Reference for Triggers

Trigger Name	Host Object	Object Type
FSL__TR021_AssignedResource.trigger	Assigned Resource	Standard Object
FSL__TR004_Event.trigger	Event	Standard Object
FSL__TR030_GanttPalette.trigger	Gantt Palette	Custom Object
FSL__TR029_GanttFilter.trigger	Gantt Filter	Custom Object
FSL__TR034_OperatingHours.trigger	Operating Hours	Standard Object
FSL__TR013_OptimizationRequest.trigger	Optimization Request	Custom Object
FSL__TR028_Polygon.trigger	Map Polygon	Custom Object
FSL__TR007_ResourceAbsence.trigger	Resource Absence	Standard Object
FSL__TR010_SchedulingPolicy.trigger	Scheduling Policy	Custom Object
FSL__SchedulingPolicyWorkRule.trigger	Scheduling Policy Work Rule	Custom Object
FSL__TR001_Service.trigger	Service Appointment	Standard Object
FSL__TR008_ServiceResource.trigger	Service Resource	Standard Object

Trigger Name	Host Object	Object Type
FSL__TR012_Capacity.trigger	Service Resource Capacity	Standard Object
FSL__TR066_ServiceObjective.trigger	Service Objective	Custom Object
FSL__TR025_ServiceResourceSkill.trigger	Service Resource Skill	Standard Object
FSL__TR0023_ServiceTerritory.trigger	Service Territory	Standard Object
FSL__TR020_ResourceTerritories.trigger	Service Territory Member	Standard Object
FSL__TR0024_ServiceTerritory.trigger	Skill Requirement	Standard Object
FSL__TR051_TimeDependency.trigger	Time Dependency	Custom Object
FSL__TR027_TimeSlot.trigger	Time Slot	Standard Object
FSL__TR005_UserTerritory.trigger	User Territory	Custom Object
FSL__TR022_WorkOrder.trigger	Work Order	Standard Object
FSL__TR026_WorkRule.trigger	Work Rule	Custom Object
FSL__TR022_WorkOrderLineItem.trigger	Work Order Line Item	Standard Object

Assigned Resource

Before Insert:

- Stops the trigger execution if one of the created resources is crew members. This situation is handled in a different flow.
- Fails the trigger if the inserted assigned resources don't have the scheduled start and schedule end time fields populated.
- Fails the trigger if the created resources don't have a valid service territory member assigned to them.
- Populates the created resource's ServiceCrewId field if the resources are from type Crew.

After Insert:

- If the Enable User Territories sharing setting is enabled, this trigger shares the created service appointment with the service territory public group. This functionality provides sharing to the dispatchers that are part of the service appointment territory. If the service resource is relocated, the appointment is shared with the public group of the new territory, ensuring dispatchers have access to relevant service appointments.
- If the Enable User Territories sharing setting is enabled and the respective service appointment parent sharing setting is enabled, this trigger shares the created service appointment's parent with the service territory public group. This action provides sharing to the dispatchers that are part of the service appointment territory. To learn more, see [Limit Access to Field Service Records](#).
- If the status categories change to Dispatched, this trigger shares the created resource's service appointments with the service resource's users. If the service territories change, this trigger shares with the service territory public groups.
- Mentions the assigned resource user if the relevant setting is enabled.
- Performs this operation in an asynchronous method if possible.
- Stops the trigger execution if one of the created resources is crew members. This behavior is handled in a different flow.
- Updates or creates a calendar event based on the created resource.
- If the assigned resources' service statuses are either None or Canceled, this trigger changes their statuses to Assigned.

- If the created resources' service is a multi-day service appointment, this trigger checks if it's necessary to recalculate the service duration. If so, the trigger recalculates the length.
- If the travel trigger is enabled in the org, this trigger recalculates travel for all services scheduled on the same day as the services' start and end dates.
- Creates assigned resources for crew members. If the resource in the trigger is for the resource type crew, we create a single resource per crew member as well.
- Updates the ResourceCapacity object according to the resources' service appointments. If a service appointment is scheduled to a contractor, this trigger updates their capacity to reflect the actual scheduled working time and working items allocated.

Before Update:

- Fails the trigger if the created Assigned Resources record doesn't have a valid Service Territory Member assigned to them.
- Populates the ServiceCrewId field if the service resources are from type crew.

After Update:

- If the Enable User Territories sharing setting is enabled, this trigger shares the created service appointment with the service territory public group. This functionality provides sharing to the dispatchers that are part of the service appointment territory. If the service resource is relocated, the appointment is shared with the public group of the new territory, ensuring dispatchers have access to relevant service appointments.
- If the service appointment is a multi-day appointment, this trigger checks if there's a need to recalculate the service duration. If so, this trigger recalculates the length.
- Updates or creates a calendar event based on the assigned resources updated.
- Creates and deletes assigned resources based on the resources' updates. If an assigned resource in the trigger is for the service resource of type crew, we create a single assigned resource per crew member as well.
- Updates the Resource Capacity object according to the assigned resources' service appointments. If an assigned resource was scheduled to a contractor, we update their capacity to reflect the actual scheduled working time and working items allocated.
- If the Service Resource field was changed on the assigned resource update, this trigger removes old sharing and creates sharing according to the service resource's and the dispatcher's territories.
- If the Make assigned resources followers of service appointments that are Dispatched or In Progress setting is enabled in the settings page, and the respective service appointment Status Category field was updated, this trigger makes the updated assigned resources followers of the service appointments and parents.
- If the travel trigger is enabled in the org, this trigger recalculates travel for all services scheduled in the same day of the services' start and end dates.
- Updates the schedule mode according to the actual scheduling operation (that is, Optimization, Automatic, Manual).

Before Delete:

- Removes sharing from the delete assigned resources' service appointments.
- If the deleted assigned resource is a resource of the type crew, this trigger deletes the respective crew members appointment resources.

After Delete:

- If the travel trigger is enabled in the org, this trigger recalculates travel for all services scheduled in the same day of the deleted assigned resources' services' start and end dates.
- Updates the Resource Capacity object according to the deleted assigned resources' service appointments. If a service appointment was unscheduled from a contractor, update their capacity to reflect the actual scheduled working time and working items allocated.
- If the Make assigned resources followers of service appointments that are Dispatched or In Progress setting is enabled in the settings page, this trigger unfollows the deleted assigned resources' service appointments and service appointments' parents.
- Nullifies the deleted assigned resources' Service Appointments' scheduled start and end time, and changes the status to None.

- If one of two Follow Immediately chains were unscheduled, that is, the assigned resource got deleted, this trigger unschedules the respective service appointment in the chain. To learn more, see [Schedule an Appointment That Immediately Follows Another](#).

Event

All triggers for this object are part of the Calendar Sync feature, which is described in [Calendar Sync's Knowledge Article](#).

Before Insert:

- Creates Resource Absences, Work Orders, Service Appointments, and Assigned Resources according to the settings described in [Calendar Sync's Knowledge Article](#).
- Calendar Sync's settings are located in the Field Service settings page. From the settings page, select **Sharing > Calendar Sync**.

After Update:

This file is called FSL__TR004_Event_BeforeUpdate but fires only after update.

- If one or more of the following fields has changed, this trigger updates the Salesforce records accordingly: OwnerId, Subject, Location, StartDateTime, EndDateTime.
- If the FSL__Event_Type__c field has changed and the new value is valid, this trigger creates the Salesforce records accordingly.

After Delete:

This file is called FSL__TR004_Event_BeforeDelete but fires only after delete.

- If the FSL__Event_Type__c field matches the 'Calendar Event type' setting of Service Appointment, and the service hasn't been updated, then this trigger deletes the Assigned Resource related to that service. This trigger unschedules the Service Appointment.
- If the FSL__Event_Type__c field matches the 'Calendar Event type' setting of Resource Absence, then this trigger deletes the related absence.

Gantt Filter

After Insert:

- If the Make this filter available for all users checkbox is checked, this trigger creates a Share record for the FSL__Gantt_Filter__c and AllInternalUsers group.

After Update:

- If the Make this filter available for all users checkbox is unchecked in the update, then this trigger deletes its share record.
- If the checkbox is checked in the update, then this trigger adds a sharing record as described in After Insert.

Gantt Palette

After Insert:

- Add sharing to the AllInternalUsers group.

After Update:

- Add sharing to the AllInternalUsers group if no such sharing exists already.

Map Polygon

Before Insert:

- Sets the minimum and maximum Latitude and Longitude of the polygon to the designated fields.

Before Update:

- Sets the minimum and maximum Latitude and Longitude of the polygon to the designated fields.

Operating Hours

Before Delete:

- Validates that the record deleted isn't the default operating hours in appointment booking settings. From the Field Service settings page, select **Global Actions > Appointment Booking**.

After Update:

- If the Appointment Bundling feature is enabled, and the Use Bundle Apex Mode custom setting equals 2, the serviceTerritoryRefresh API http request is triggered.

Optimization Request

Before Insert:

- Sets the FSL__Text_Status__c text field with the value of the FSL__Status__c picklist field. The FSL__Text_Status__c is a text field marked as an External ID and is used to improve query performance.

Before Update:

- Prevents the FSL__Status__c picklist field from changing from In Progress to Queued. This trigger returns the status to In Progress.
- Validates that a record with the FSL__Status__c value of Aborted doesn't change to Completed. If it does, this trigger prevents the change and throws an error.
- If the status is changed to Aborted, this trigger checks the following conditions:
 - The change wasn't caused by the optimization user.
 - The Optimization Request wasn't aborted by the auto-kill feature. That is, the feature that kills requests that stay in a specific status more than what's defined in the custom settings.
 - The user has Abort_Optimization_Request custom permission on their profile or one of their permission sets.

If all the previously mentioned criteria are met, the request is aborted.

Resource Absence

Before Insert:

- Changes the scale of the geolocation values (latitude and longitude) to 6 digits after the decimal point.
- Removes seconds and milliseconds from the resource absence DateTime fields.
- Fills the InternalSLRGeolocation fields (latitude and longitude) based on the geolocation values. The InternalSLRGeolocation fields are used in street-level routing and predictive route calculations.

After Insert:

- If the inserted resource absence is the type nonavailability then:
 - If a multi-day service appointment was affected by the resource absence and there's just one service appointment, this trigger updates its duration.
 - If Fix Overlaps is enabled in the org and Travel Trigger is disabled, this trigger calls Fix Overlaps.
 - If Travel Trigger is enabled in the org and no multi-day appointment was updated in step 1, this trigger recalculates travel for all services scheduled in the same day of the absences' start and end dates.

- If the calendar sync feature is turned on then the trigger creates a calendar event for the resource absence.
- Otherwise, this trigger does nothing.

Before Update: (content is the same as Before Insert)

- Changes the scale of the geolocation values (latitude and longitude) to 6 digits after the decimal point.
- Removes seconds and milliseconds from the resource absence's DateTime fields.
- Fills the InternalSLRGeolocation fields (latitude and longitude) based on the geolocation values. The InternalSLRGeolocation fields are used in street-level routing and predictive routes' calculations. We don't check for a change in the geolocation values—we update them like in the Before Insert trigger.

After Update:

- If the updated resource absence is the type non availability then:
 - If a multi-day service appointment was affected by the resource absence change and there's just one service appointment, this trigger updates its duration.
 - If Fix Overlaps is enabled in the org and Travel Trigger is disabled, this trigger calls Fix Overlaps.
 - If Travel Trigger is enabled in the org and no multi-day appointment was updated in step 1, this trigger recalculates travel for all services scheduled in the same day of the absences' start and end dates.
 - If the calendar sync feature is turned on then the trigger creates a calendar event for the resource absence.
- Otherwise, this trigger does nothing.

Before Delete:

- Deletes the matching calendar event if it exists. See the [Calendar Sync's Knowledge Article](#) for more information.

After Delete:

- If the deleted resource absence is the type nonavailability then:
 - If a multi-day service appointment was affected by the resource absence deletion and there's just one service appointment, this trigger updates its duration.
 - If Fix Overlaps is enabled in the org and Travel Trigger is disabled, this trigger calls Fix Overlaps.
 - If Travel Trigger is enabled in the org and no multi-day appointment was updated in step 1, this trigger recalculates travel for all services scheduled in the same day of the absences' start and end dates.

Scheduling Policy

Before Insert:

- Validates that the Travel Mode checkbox is checked only when this feature is enabled in the org.
- Validates that In-Day Optimization is checked only when Enhanced Optimization is enabled in the org.

After Insert:

- Creates Earliest Start Time and Due Date basic time rules if they don't exist.
- Create the Scheduling Policy Work Rule junction object to assign these work rules to the inserted policy.

Before Update:

- Validates that the Travel Mode checkbox is checked only when this feature is enabled in the org.
- Validates that In-Day Optimization is checked only when Enhanced Optimization is enabled in the org.

Before Delete:

- Validate that the deleted record isn't the default scheduling policy for the appointment booking action.
- Validates that the Travel Mode checkbox is checked only when this feature is enabled in the org.
- Validates that In-Day Optimization is checked only when Enhanced Optimization is enabled in the org.
- Deletes the Earliest Start Time and Due Date Scheduling Policy Work Rules junction objects.

Scheduling Policy Work Rule

Before Insert:

- Validates that there are no duplications. That is, validates that there's just one Scheduling Policy Work Rule record referencing a specific work rule in the same policy.
- Validates that there aren't more than two enhanced match rules in the same policy. If there are two, make sure they're referencing different Linking Objects.
- Validates that there aren't more than four Count Rules with Count Type of CustomValue.
- Validates that all Count Rule fields are valid.

Before Update:

Same as Before Insert.

Before Delete:

- Prevents the deletion of a Scheduling Policy Work Rule record related to Due Date and Earliest Start Time basic rules.
- If you delete the Scheduling Policy, it passes over this validation and deletes all related Scheduling Policy Work Rules.

Service Appointment

Before Insert:

- Remove seconds from the Service Appointment's DateTime fields.
- If there's a geolocation (latitude and longitude) on the created ServiceAppointment record, limit the decimal places after the decimal point to a maximum of 6 digits.
- If there's a geolocation (latitude and longitude) on the created ServiceAppointment record, populate the FSL__InternalSLRGeolocation__Latitude__s, FSL__InternalSLRGeolocation__Longitude__s fields. These fields are used when calculating street-level routing routes.
- If the Derive the Service Appointment due date from its Work Type setting is enabled, this trigger derives the Duration field from the work order parent record and populates the Due Date field with the value of Earliest Start Permitted plus Due Date offset, which comes from the parent's work type field.
- If the Set your default service appointment duration to one hour setting is enabled, this trigger sets the service appointment duration to 1 hour if one of the following occurs:
 - The duration field is null.
 - The duration is shorter than 1 minute.

After Insert:

- If the Enable User Territories sharing setting is enabled, share the created service appointment with the service territory public group. This behavior provides sharing to the dispatchers who are part of the service appointment's territory.
- If the Enable User Territories sharing setting is enabled and the respective service appointment parent sharing setting is enabled, that is, Share parent Work Order when Service Appointment is shared, this trigger shares the created service appointment's parent

with the service territory public group. This behavior provides sharing to the dispatchers who are part of the service appointment's territory.

- If the Derive the Service Appointment due date from its Work Type setting is enabled, this trigger derives the Duration field from the work order parent record and populates the Due Date field with the value of Earliest Start Permitted plus Due Date offset, which comes from the parent's work type field. This behavior is intentionally in both the Before Insert and the After Insert triggers.
- Schedules created service appointments in batches if they have the Auto Schedule field value set to true. The default policy, which is set in the Field Service settings page, is used when auto-scheduling service appointments upon creation.
- After the batch was invoked, the Auto Schedule flag is set to false automatically.
- If the Use polygons to assign service territories setting is enabled, this trigger auto populates the created service appointment's service territory field based on the service appointment's address and its match of a Map Polygon Service Territory field.
- If there's no match between the service appointment's address and the organization map polygons, the service territory field remains unchanged.
- If the Appointment Bundling feature is enabled; the Use Bundle Apex Mode custom setting equals 2; and the inserted service appointment is a bundle or a bundle member; then the afterServiceAppointmentUpdate API http request is triggered.
- If the Appointment Bundling feature is enabled; automatic bundling is enabled; the inserted service appointment isn't a bundle or a bundle member; the modified fields of the service appointment exist within the restrictions of the automatic mode; the service territory of the service appointment is in the defined ApptBundlePolicySvcTerr; the Recordset Filter Criteria matches; the status of the service appointment exists in the bundle statuses; the service appointment isn't pinned; and the due date isn't in the past while considering the timezone of the service appointment's service territory; then the automatic API request is triggered.

Before Update:

- Remove seconds from the service appointment's DateTime fields.
- If there's a geolocation (latitude and longitude) on the created ServiceAppointment record, this trigger limits the decimal places after the decimal point to a maximum of 6 digits.
- If Service Appointment Status Transitions are enabled in the Field Service settings page and there was a change in the service appointment status field, this trigger validates the status transition legality.
- If the updated service appointment's Status Category was changed to either Canceled or New, this trigger removes the Scheduled Start Time and Scheduled End Time field values.
- Prevents updating the Scheduled Start Time, Scheduled End Time, Latitude, and Longitude field values if the Is Pinned field value is set to true.
- If the service appointment is a multi-day service appointment, this trigger checks if there's a need to recalculate the service duration. If so, it recalculates the length.
- If the Drip-Feed feature is enabled in the settings page, this trigger dispatches the next service appointment in the day. This action is done in case the current service appointment status categories changed to Completed, Canceled or, Couldn't Complete. For more information, see [Drip Feed Service Appointments](#) in Salesforce Help.
- If the Mention assigned user when the Service Appointment is dispatched setting is enabled, mention the user upon service cancellation.
- Validates that the updated service appointment duration is longer than 1 minute. if the duration is null or 1 minute, change the duration to 1 hour.
- If there's a geolocation (latitude and longitude) on the updated service record and it has changed within the update, this trigger populates the FSL__InternalSLRGeolocation__Latitude__s, FSL__InternalSLRGeolocation__Longitude__s fields. Those fields are used when calculating street-level routing routes.
- If the FSL__Prevent_Geocoding_For_Chatter_Actions__c field is checked, this trigger disables Data.com's geolocation cleanup, which happens on address change, and returns the value of FSL__Prevent_Geocoding_For_Chatter_Actions__c field to false.

- If the Auto Schedule field on the updated service appointment record is set to true upon the service appointment's update process, this trigger prepares a set of service appointments that are called on the After Update trigger operation and sets the field to false.
- If the Use polygons to assign service territories setting is enabled and their geolocations fall within relevant Map Polygons records, this trigger classifies the updated service appointment territories based on a geographic area. For more information, see [Create and Manage Map Polygons](#) and [Enable Map Polygons](#) in Salesforce Help.

After Update:

- Shares the updated service appointments with the assigned resources if the status categories were changed to Dispatched. Shares with the service territory public groups if the service territories changed.
- Mentions the assigned resource user if the relevant setting is enabled.
- Performs this operation in an asynchronous method if not already in an async context.
- Creates and deletes Salesforce calendar events based on the service appointment status changes.
 - If the SA Status category changed from New/Scheduled to Dispatched, this trigger creates a calendar event.
 - If the service appointment status category changed from Dispatched to Scheduled/New, this trigger deletes the calendar event.
 - If a service appointment assigned resource was changed and the service appointment remained dispatched, this trigger updates the relevant Salesforce calendar event accordingly.
- Updates the Resource Capacity object according to the service appointment change. If a service appointment was scheduled to a contractor, this trigger updates their capacity to reflect the actual scheduled working time and working items allocated.
- Schedules updated service appointments in batches if they have the Auto Schedule field value set to true. The default policy, which is set in the Field Service settings page, is used when auto-scheduling service appointments upon update.
- After the batch is invoked, the Auto Schedule flag is set to false automatically.
- If the Make assigned resources followers of service appointments that are Dispatched or In Progress setting is enabled in the settings page, and the Status Category field was updated, this trigger makes the assigned resources followers of the updated service appointments.
- If an updated service appointment is assigned to a Service Crew, this trigger creates/updates/deletes the service's assigned resources according to the newly updated service appointment to reflect the actual Service Crew scheduling. To learn more, see [Considerations for Scheduling Service Crews](#) in Salesforce Help.
- If the travel trigger is enabled in the org, this trigger recalculates travel for all services scheduled in the same day of the services' start and end dates.
- If the Scheduled Start and End Times of the updated service appointments are null but the service appointments have an assigned resource related to them, this trigger deletes the assigned resources.
- Updates the schedule mode according to the actual scheduling operation, that is, Optimization, Automatic, or Manual.
- Evaluates the scheduling recipes according to the actual service appointment update, that is, Canceled, Shortened, Late-end, or Emergency.
- To learn more, see [Create Scheduling Recipes for Common Events](#) in Salesforce Help.
- If the Appointment Bundling feature is enabled; the Use Bundle Apex Mode custom setting equals 2; and the updated service appointment is a bundle or a bundle member; then run the bundle service appointment logic.
- If the Appointment Bundling feature is enabled; the Use Bundle Apex Mode custom setting equals 2; and the updated service appointment is a bundle or a bundle member; then the afterServiceAppointmentUpdate API http request is triggered.
- If the Appointment Bundling feature is enabled; the Use Bundle Apex Mode custom setting equals 2; and the updated service appointment is a bundle member; then the trigger doesn't update the SchedStartTime and the SchedEndTime fields when the RelatedBundled field hasn't changed. Only if the RelatedBundled is updated, the SchedStartTime and the SchedEndTime fields change.

- If the Appointment Bundling feature is enabled; automatic bundling is enabled; the updated service appointment isn't a bundle or a bundle member; the modified fields of the service appointment exist within the restrictions of the automatic mode; the service territory of the service appointment is in the defined ApptBundlePolicySvcTerr; the Recordset Filter Criteria matches; the status of the service appointment exists in the bundle statuses; the service appointment isn't pinned; and the due date isn't in the past while considering the timezone of the service appointment's service territory; then the automatic API request is triggered.

Before Delete:

- Deletes the matching calendar event if it exists. See the [Calendar Sync's Knowledge Article](#) for more information.
- Deletes assigned resource records from the deleted service appointments.
- Remove sharing on the parent record, that is, the Work Order, from the territories' public groups.

Service Objective

Before Insert:

- Validates that the service objective has a record type.
- If the service objective is a custom logic objective, validates and decodes the Custom Logic Data (must be filled).

Before Update:

Same as Before Insert.

Service Resource

Before Insert:

- If the Efficiency field isn't empty, this trigger checks if its value is in the valid range (between 0.1–10). If not, this trigger prevents the insertion and shows an error next to the Efficiency field.
- If the ServiceCrewId field is empty, this trigger checks if there's another Service Resource record pointing to the same service crew. If so, the trigger throws an error and prevents the insertion.

After Insert:

- If location-based sharing is turned on in your org, this trigger creates a ServiceResourceShare object for the user specified in the RelatedRecordId field. If location-based sharing is turned off it, this trigger does nothing. If the RelatedRecordId field is empty for a specific record, for example, a Service Resource of type Crew, this trigger doesn't create a share object for the specific record.

Before Update:

- If the IsCapacityBased checkbox changed from false to true, this trigger validates there's only one service territory member, excluding secondary service territory members, linked to that resource. Otherwise, this trigger throws an error and prevents the update.
- If the Efficiency field isn't empty, this trigger checks if its value is in the valid range (between 0.1–10). If not, this trigger prevents the update and shows an error next to the Efficiency field.
- If the IsCapacityBased checkbox is checked, this trigger validates that this service resource doesn't have any Service Crew Member records. Otherwise, this trigger throws an error and prevents the update.
- If the ServiceCrewId field changed and the new value isn't empty, this trigger validates that there's no existing Service Resource record pointing to that Service Crew. Otherwise, this trigger throws an error and prevents the update.

After Update

- If the RelatedRecordId changed in the update and location-based sharing is turned on in your org, this trigger deletes all manual sharing records for the Service Resource, including those records created by the user, and creates a ServiceResourceShare object for the new related user. If it fails to delete one of the manual sharing objects this trigger doesn't delete any of them.

Service Resource Capacity

The trigger is named TR012_CapacityAfterUpdate but it fires only before insert and before update.

Before Insert:

- Validates that monthly capacities are defined in the first day of the month.
- Validates that capacities of the same duration type don't overlap each other, that is, they don't share a resource and date.
- Validates that the Time Period and End Date fields are valid.
- Updates the MinutesUsed__c field according to the duration of services scheduled within the capacity.
- If the CapacityInWorkItems field isn't empty, then this trigger updates the Work_Items_Allocated__c field with the number of services scheduled within the capacity.
- HoursInUse__c field is a formula field based on MinutesUsed__c. This field updates accordingly.
- Updates the Last Updated Epoch field with the number of milliseconds since January 1, 1970, 00:00:00 GMT.

Before Update:

- Validates that monthly capacities are defined in the first day of the month.
- Validates that capacities of the same duration type don't overlap, that is, they don't share a resource and date.
- Validates that the the Time Period and End Date fields are valid.
- Validates if one or more of the following fields have changed: TimePeriod, StartDate, CapacityInHours, CapacityInWorkItems, ServiceResourceId.
- If any of these fields have changed, this trigger updates the MinutesUsed__c field according to the duration of services scheduled within the capacity.
- If the CapacityInWorkItems field isn't empty, then this trigger updates the Work_Items_Allocated__c field with the number of services scheduled within the capacity.
- HoursInUse__c field is a formula field based on MinutesUsed__c. This field updates accordingly.
- Updates the Last Updated Epoch field with the number of milliseconds since January 1, 1970, 00:00:00 GMT.

Service Resource Skill

Before Insert:

- Removes seconds and milliseconds from the Effective Start Date and Effective End Date fields.

Before Update:

Same as Before Insert.

Service Territory

Before Insert:

- If the Enable Territory Name Duplicates custom setting is off (the default value is off), then this trigger makes sure there's no duplication in the names of the inserted territories.
- If the territory has a geolocation, then this trigger fills up the internal street-level routing geolocation values.

After Insert:

- If the Enable User Territories sharing setting is enabled, this trigger creates a new public group with the name of the territory, if no such group exists, and creates a sharing record for that group.
- Adds the public group of the territory as a group member to the public group of the parent territory, if it has one.

- If the Enable Service Auto Classification custom setting is on (the default is on), this trigger sets the FSL__TerritoryLevel__c field according to the parent territory and top-level territory values of all territories. This field represents the level of the territory in the hierarchy and is used in the polygon's algorithm.
- If the Appointment Bundling feature is enabled; and the Use Bundle Apex Mode custom setting equals 2; then the serviceTerritoryRefresh API http request is triggered.

Before Update:

- If the Enable Territory Name Duplicates custom setting is off (the default value is off) and the Name field has changed, then this trigger makes sure there's no duplication in the names of the updated territories.
- If the territory has a geolocation, then this trigger fills up the internal street-level routing geolocation values.

After Update:

- If the name or owner fields changed, this trigger updates the public group of the territory accordingly.
- If the Parent Territory field changed, this trigger updates the Parent Territory's public group and recalculates the FSL__TerritoryLevel__c field value to all records according to the new hierarchy.
- If the Appointment Bundling feature is enabled; the Use Bundle Apex Mode custom setting equals 2; and the Operating Hours property was updated; then the serviceTerritoryRefresh API http request is triggered.

Before Delete:

- Deletes the Service Territory's public group.
- Updates children territory's FSL__TerritoryLevel__c field according to the new hierarchy.
- If the Appointment Bundling feature is enabled; and the Use Bundle Apex Mode custom setting equals 2; then the serviceTerritoryRefresh API http request is triggered.

Service Territory Member

Before Insert:

- Removes seconds from the service territory member DateTime fields.
- Checks for date collisions. If a primary or relocation service territory member exists in the same date, the trigger fails.
- If a secondary service territory member exists in the same date and the created service territory member is in the same service territory, the trigger fails.
- If there's a geolocation (latitude and longitude) on the created service territory member record, this trigger populates the FSL__InternalSLRGeolocation__Latitude__s, FSL__InternalSLRGeolocation__Longitude__s fields. Those fields are used when calculating street-level routing routes.
- Prevents creating service territory members of type Relocation to contractors.
- Prevents creating more than one service territory member, excluding Secondary service territory members, to contractors.

Before Update:

- Checks for date collisions. If a primary or relocation service territory member exists in the same date, the trigger fails.
- If a secondary service territory member exists in the same date and the created service territory member is in the same service territory, the trigger fails.
- If there's a geolocation (latitude and longitude) on the updated service territory member record, this trigger populates the FSL__InternalSLRGeolocation__Latitude__s, FSL__InternalSLRGeolocation__Longitude__s fields. Those fields are used when calculating street-level routing routes.
- Prevents creating multiple service territory members of type Relocation or Primary to contractors.
- Prevents creating more than one service territory member (excluding Secondary service territory members) to contractors.

Skill Requirement

All triggers are empty.

Time Dependency

After Insert:

- Prevents creating duplicated dependencies between service appointments.
- Prevents adding an Immediately Follow dependency type to an already created time dependency chain.
- Adds service appointments to an existing chain according to the created Time Dependency. If needed, this trigger merges two chains according to the created Time Dependency.

Before Update:

- Prevents updating a Time Dependency if an existing Time Dependency has the same two service appointments.
- Prevents adding an Immediately Follow dependency type to an already created time dependency chain.

After Delete:

- Splits two dependencies according to the deleted Time Dependency. For example, if chain A contains SerA SerB SerC SerD SerE and the deleted Time Dependency is SerC SerD, this trigger splits the chain into two: SerA SerB SerC and SerD SerE.

Time Slot

Before Insert:

- Removes seconds from DateTime fields.

Before Update:

- Removes seconds from DateTime fields.

User Territory

Before Insert:

- If the Enable User Territories sharing setting is enabled, this trigger checks for uniqueness within all user territories. In other words, this trigger ensures that the same user isn't related to the same territory.

After Insert:

- If the Enable User Territories sharing setting is enabled, this trigger adds users to the respective public groups according to the created User Territory. Make the User Territory.Service Territory name the same as the public group name.

BeforeUpdate:

- If the Enable User Territories sharing setting is enabled, this trigger checks for uniqueness within all user territories. In other words, this trigger ensures that the same user isn't related to the same territory.

After Update:

- If the Enable User Territories sharing setting is enabled, this trigger adds and removes users to or from the respective public groups according to the updated user territory. For example, if a user territory is updated from User: David, Service Territory: LA to User: David, Service Territory: Washington, this change results in removing David from the LA public group and adding him to the Washington public group.

After Delete:

- If the Enable User Territories sharing setting is enabled, this trigger removes users from the respective public groups according to the deleted user territory.

Work Order

Before Insert:

- If the AccountId field isn't empty and the VisitingHoursId field is empty, this trigger populates VisitingHoursId with the related account's operating hours ID.

Before Update:

- If the AccountId or VisitingHoursId fields changed, the AccountId field isn't empty, and the VisitingHoursId field is empty, this trigger populates VisitingHoursId with the related account's operating hours ID.
- If the FSL__Prevent_Geocoding_For_Chatter_Actions__c field is checked, this trigger disables Data.com's geolocation cleanup, which happens on an address change, and returns the value of FSL__Prevent_Geocoding_For_Chatter_Actions__c field to false.

Work Order Line Item

All triggers are empty.

Work Rule

Before Update:

- If the work rule is Availability Rule, this trigger prevents setting a minimum gap if the fixed gap checkbox is disabled.
- If the work rule is Availability Rule, this trigger prevents setting a break time if the Break And Travel trigger is disabled.
- If the work rule is Match Boolean Rule, this trigger prevents updating the work rule if the resource property field isn't populated.
- If the work rule is a Time Rule, this trigger prevents updating a time rule with Scheduled Start Time Equal To/Before Arrival Window Start/End.
- If the work rule is an Enhanced Match Rule or a Count Rule, this trigger prevents updating the rules with invalid field values.
- If basic time work rules don't exist in the org or are changed, this trigger creates valid basic time rules (Early Start Permitted & Due Date work rules).

Before Delete:

- Prevent deletion of the basic time work rules (Early Start Permitted & Due Date).

Field Service Code Examples

Use these code examples to get started working programmatically with Field Service features.

IN THIS SECTION:

[Create a Service Report with Apex](#)

The Create Service Report action on work orders, work order line items, and service appointments can also be called using Apex code. The code example shown creates a service report with two signatures by making an Apex callout to the `createServiceReport` action REST API resource.

[Generate Work Orders on Maintenance Plans with Apex](#)

The **Generate Work Orders** action on maintenance plans can also be called using Apex code. The following code sample creates work order records by making an Apex callout to the `generateWorkOrder` action REST API resource.

[Code Examples: Dispatcher Console Custom Actions](#)

Learn how to configure Apex classes or Visualforce pages that you want to link to a custom action in the dispatcher console.

[Create Service Appointment Lists in the Dispatcher Console](#)

You can create user-specific temporary appointment lists by adding the `Create Temporary Service Appointment List` custom permission in Field Service managed package. If the custom permission is enabled, the Field Service app creates a connection to the `CreateFilterEvent__e` platform event channel and subscribes to related messages.

Create a Service Report with Apex

The Create Service Report action on work orders, work order line items, and service appointments can also be called using Apex code. The code example shown creates a service report with two signatures by making an Apex callout to the `createServiceReport` action REST API resource.

Each service report contains data from its parent record and signatures from customers, technicians, or others who must sign off on the work. In your API call, you can define one or more signatures to be used in the report. The number and type of signatures must match the service report template's signature settings.

Note:

- This code example applies to API version 41.0 and later. Service reports generated with API version 40.0 can contain only one signature.
- This code example assumes that you have remote site settings enabled to make REST callouts. For more information, see [Adding Remote Site Settings](#).
- The mock record ID in the code example is `0WOxx000000001E`. The mock service report template ID is `0SLR00000004DBFOA2`.

API Endpoint (v41.0): `/services/data/v41.0/actions/standard/createServiceReport`

```
String salesforceHost = System.Url.getSalesforceBaseUrl().toExternalForm();
String url = salesforceHost + '/services/data/v41.0/actions/standard/createServiceReport';
// Create HTTP request
HttpRequest request = new HttpRequest();
request.setEndpoint(url);
request.setMethod('POST');
request.setHeader('Content-Type', 'application/json');
request.setHeader('Authorization', 'OAuth ' + UserInfo.getSessionId());
// Set the body as a JSON object
request.setBody('{"inputs" : [ {"entityId" : "0WOxx000000001E","signatures" :
[{"data": "Base64 code for the captured signature
image","contentType": "image/png", "name": "Customer
Signature","signatureType": "Default", "place": "San Francisco", "signedBy": "John
Doe", "signedDate": "2019-06-05 12:00:00"}], "templateId" : "0SLR00000004DBFOA2"} ]}');
Http http = new Http();
HttpResponse response = http.send(request);
// Parse the JSON response
if (response.getStatusCode() != 201) {
    System.debug('The status code returned was not expected: ' +
        response.getStatusCode() + ' ' + response.getStatus());
} else {
```

```
System.debug(response.getBody());
}
```

This code example creates a payload with two signatures.

```
{
  "inputs": [
    {
      "entityId": "0WOxx000000001E",
      "signatures": [
        {
          "data": "Base64 code for the captured signature image",
          "contentType": "image/png",
          "name": "Customer Signature",
          "signatureType": "Customer",
          "place": "San Francisco",
          "signedBy": "John Doe",
          "signedDate": "Thu Jul 13 22:34:43 GMT 2017"
        },
        {
          "data": "Base64 code for the captured signature image",
          "contentType": "image/png",
          "name": "Technician Signature",
          "signatureType": "Technician"
        }
      ]
    },
    {
      "templateId": "0SLR00000004DBFOA2"
    }
  ]
}
```

Field Descriptions

- **entityId:** The ID of the work order, work order line item, or service report that you want to create the service report for.
- **signatures:** A list of JSON definitions for a digital signature.
 - **data:** (Required) The base64 code for an image.
 - **contentType:** (Required) The file type of the signature.
 - **signatureType:** (Required) The role of the person signing such as Customer. Signature Type picklist values are defined by the Salesforce admin ahead of time. Each signature block must use a different signature type, and the signature types that you define in your call must match the service report template's signature types.
 - **name:** The signature block title. This value appears on the generated service report.
 - **place:** The place of signing. This value appears on the generated service report.
 - **signedBy:** The name of the person signing. This value appears on the generated service report.
 - **signedDate:** The date of signing. This value appears on the generated service report.
- **templateId:** The ID of the service report template used for the report. To find the ID, run a SOQL query on the `ServiceReportLayout` object.

Limitations

If you're a guest user and are using *Future* annotation, the ContentDocument object can't be created and you can't create the Service Document.

SEE ALSO:

[Salesforce Actions Developer Guide: Create Service Report Actions](#)

Generate Work Orders on Maintenance Plans with Apex

The **Generate Work Orders** action on maintenance plans can also be called using Apex code. The following code sample creates work order records by making an Apex callout to the *generateWorkOrder* action REST API resource.

You can use this code sample in several different ways:

- Add it to Apex controller code for an Aura component, and tie it to a custom UI or app functionality
- Use it in an Apex trigger to semi-automate the creation of the work orders (for example, whenever a maintenance plan is created or updated)
- Use it in an Apex REST service to create work orders when called from an external integration service



Note:

- This code sample assumes that you have remote site settings enabled to make REST callouts. For more information, see [Adding Remote Site Settings](#).
- The mock maintenance plan ID in the code sample is *1MPR000000000Bu*.
- This code sample references API version 45.0, but can be used with any version from 40.0 onward.

API Endpoint (v45.0): `/services/data/v45.0/actions/standard/generateWorkOrders`

```
String salesforceHost = System.Url.getSalesforceBaseUrl().toExternalForm();
String url = salesforceHost + '/services/data/v45.0/actions/standard/generateWorkOrders';
// Create HTTP request
HttpRequest request = new HttpRequest();
request.setEndpoint(url);
request.setMethod('POST');
request.setHeader('Content-Type', 'application/json;charset=UTF-8');
request.setHeader('Authorization', 'OAuth ' + UserInfo.getSessionId());
// Set the body as a JSON object
request.setBody('{"inputs" : [{"recordId" : "1MPR000000000Bu"}]}');
Http http = new Http();
HttpResponse response = http.send(request);
// Parse the JSON response
if (response.getStatusCode() != 201) {
    System.debug('The status code returned was not expected: ' +
        response.getStatusCode() + ' ' + response.getStatus());
} else {
    System.debug(response.getBody());
}
```

SEE ALSO:

[Salesforce Actions Developer Guide: Generate Work Orders Actions](#)

Code Examples: Dispatcher Console Custom Actions

Learn how to configure Apex classes or Visualforce pages that you want to link to a custom action in the dispatcher console.

Custom actions can either call an Apex class or open a Visualforce page, and can be run on records in several areas of the dispatcher console. To learn how to create custom actions, see [Create Custom Actions for the Dispatcher Console](#).



Note: All Apex classes implementing the quick action interfaces must be declared as Global to be accessible from the dispatcher console.

Creating Apex Classes

When you create an Apex class to link to a dispatcher console custom action, implement one of the following three interfaces in your class.

Interface	Description
<code>CustomGanttServiceResourceAction</code>	For actions on service resources. The parameters are the service resource record ID, the service territory member record ID reflected on the Gantt, and the start and end dates of the current Gantt view. No additional parameters are included. Use the following format: <code>String action(Id resourceId, Id stmId, Datetime strGanttStartDate, Datetime strGanttEndDate, Map<String, Object> additionalParameters)</code>
<code>CustomGanttServiceAppointmentAction</code>	For actions on service appointments. The parameters are the service appointment record IDs—used for bulk actions—and the start and end dates of the current Gantt view. No additional parameters are included. The <code>Map<String, Object> additionalParameters</code> input parameter is reserved for future use but must be included to run the code. Use the following format: <code>String action(List<Id> serviceAppointmentsIds, Datetime strGanttStartDate, Datetime strGanttEndDate, Map<String, Object> additionalParameters)</code> When this action is implemented, multiple service appointments can be returned. In your method, we recommend creating an <code>if</code> statement to check how many IDs are returned. First, validate that at least one ID was returned: <code>serviceAppointmentsIds.size() > 1</code>. Then, you can take different actions depending on whether 0, 1, or more appointment IDs were returned.
<code>CustomGanttResourceAbsenceAction</code>	For actions on resource absences. The parameters are the resource absence record ID, the absence type ('na' or 'break'), and the start and end dates of the current Gantt view. No additional parameters are included. The <code>Map<String, Object></code>

Interface	Description
	<code>additionalParameters</code> input parameter is reserved for future use but must be included to run the code. Use the following format: <code>String action(Id absenceId, String absenceType, Datetime strGanttStartDate, Datetime strGanttEndDate, Map<String, Object> additionalParameters)</code>

These functions must be global and require that a string be returned. If the string isn't empty, it is used in the Gantt notification shown when a user clicks the related action.

 **Note:** Use the [Datetime class](#) for all date and time calculations.

Code Example: Service Resource Custom Action

This action creates a resource absence of type Non Availability for the selected service resource that spans the days visible on the Gantt.

```
global class BlockResourceVisibleTime implements FSL.CustomGanttServiceResourceAction {

    global String action(Id resourceId, Id stmId, Datetime ganttStartDate, Datetime
ganttEndDate, Map<String, Object> additionalParameters) {

        ResourceAbsence na = new ResourceAbsence();

        // get Resource Absence record type - NA
        RecordType recordTypeNA = [
            SELECT
                Id, SubjectType, Name
            FROM
                RecordType
            WHERE
                DeveloperName =: 'Non_Availability'
                AND
                SubjectType =: ResourceAbsence.getSubjectType().getDescribe().getName()
        ];

        na.RecordTypeId = recordTypeNA.Id;
        na.ResourceId = resourceId;
        na.FSL__Approved__c = true;
        na.Start = ganttStartDate;
        na.End = ganttEndDate;

        insert na;

        ServiceResource resource = [SELECT Name FROM ServiceResource WHERE Id =: resourceId];

        return 'Blocked availability to ' + resource.Name + ' from ' +
ganttStartDate.format() + ' to ' + ganttEndDate.format();

    }
}
```

```
}

```

Code Example: Service Appointment Custom Action

This action toggles the In Jeopardy field between True and False.

```
global class toggleServiceAppointmentJeopardy implements
FSL.CustomGanttServiceAppointmentAction {

    global String action(List<Id> serviceAppointmentsIds, Datetime ganttStartDate, Datetime
ganttEndDate, Map<String, Object> additionalParameters) {

        List<ServiceAppointment> saList = [SELECT FSL__InJeopardy__c, AppointmentNumber
FROM ServiceAppointment WHERE Id in : serviceAppointmentsIds];
        String reply = '';
        List<String> saNames = new List<String>();

        for (ServiceAppointment s : saList) {
            s.FSL__InJeopardy__c = !s.FSL__InJeopardy__c;
            saNames.add(s.AppointmentNumber);
        }

        upsert saList;

        reply = String.join(saNames, ', ');
        return 'Service Appointments successfully processed: ' + reply;
    }
}
```

Code Example: Resource Absence Custom Action

For resource absences of type NA, this action creates a duplicate absence on the following day.

```
global class copyAbsenceToNextDay implements FSL.CustomGanttResourceAbsenceAction {

    global String action(Id absenceId, String absenceType, Datetime ganttStartDate, Datetime
ganttEndDate, Map<String, Object> additionalParameters) {

        ResourceAbsence resourceAbsence = [SELECT Id, AbsenceNumber, Start, End, ResourceId,
RecordTypeId, FSL__Approved__c FROM ResourceAbsence WHERE Id =: absenceId LIMIT 1];

        ResourceAbsence raClone = resourceAbsence.clone(false, true, false, false);
        raClone.Start = resourceAbsence.Start.addDays(1);
        raClone.End = resourceAbsence.End.addDays(1);
        raClone.ResourceId = resourceAbsence.ResourceId;
        raClone.RecordTypeId = resourceAbsence.RecordTypeId;
        raClone.FSL__Approved__c = true;
        insert raClone;

        return 'Resource Absence successfully copied.';
    }
}
```

Creating Visualforce Pages

When you create a Visualforce page, use the following GET parameters.

For actions on...	Description
Service appointments	Use the following format: <code>services [if multiple], id (comma delimited if multiple), start (current Gantt start date, string), end (current Gantt end date, string)</code> Example for a Visualforce page used to update a single service appointment: <code>?id=08p4E000000Kj5hQAC&start=5-7-2018&end=5-8-2018</code> Example for a Visualforce page used to update multiple service appointments: <code>?services=08p4E000000Kj5hQAC,08p4E000430Kj5hAPP&start=5-7-2018&end=5-8-2018</code>
Service resources	Use the following format: <code>id, stm (ID of service resource's current service territory member record), start (current Gantt start date, string), end (current Gantt end date, string)</code> Example: <code>?id=0Hn4E0000001OMQSA2&stm=0Hu4E0000005cpPSAQ&start=5-7-2018&end=5-8-2018</code>
Resource absences	Use the following format: <code>id, type ('break' or 'na'), start (current Gantt start date, string), end (current Gantt end date, string)</code> Example: <code>?id=0Hw4E00000091HSSAY&type=break&start=5-7-2018&end=5-8-2018</code>

To close the Visualforce lightbox from your code, use: `parent.postMessage('closeLightbox','*');`



Note: Custom dispatcher console actions can't open Visualforce pages that are part of a managed package.

SEE ALSO:

[Create Custom Actions for the Dispatcher Console](#)

Create Service Appointment Lists in the Dispatcher Console

You can create user-specific temporary appointment lists by adding the `Create Temporary Service Appointment List` custom permission in Field Service managed package. If the custom permission is enabled, the Field Service app creates a connection to the `CreateFilterEvent__e` platform event channel and subscribes to related messages.

For information on allocation limits consumed by subscribing to the event channel, see [Change Data Capture Allocations](#).

The custom filters are created and accessible only by users who created the `CreateFilterEvent__e` platform event, and not for other users. Based on a custom filter, you can create temporary lists with identical names. Each list is created as a new instance and doesn't override an existing one. Each list can include a maximum of 300 service appointments.

After a temporary appointment list is loaded to the dispatch console, the list remains static. To auto-refresh the list, enable **Enhanced Live Updates** in the Field Service Admin app and make sure you have read access to all fields in the service appointment's field sets. When the dispatch console is refreshed, the temporary list is no longer available. To delete a temporary list without refreshing, click **Delete Filter**.

To create a custom filter for appointments, define and publish a new `CreateFilterEvent__e` platform event by adding these custom fields.

- `FSL__ServiceApptIds__c`: JSON array of service appointment IDs. For example, `["08pO10000049k7lIAA", "08pO10000049k7mIAA"]`
- `FSL__FilterName__c`: Name of the temporary list based on the specified service appointment IDs. For example, *High Priority Appointments*
- `FSL__Description__c` (optional): Description. For example, *This filter displays high priority appointments.*
- `FSL__FilterCategory__c`: Value must be set to `GENERAL`
- `FSL__LoadFilterImmediately__c`: When set to `true`, the dispatcher console switches to the new temporary list.

This code sample creates a `FSL__CreateFilterEvent__e` platform event and loads a temporary list created from the queried list of service appointment IDs to the dispatcher console.

```
List<id> ids = new List<id>();
for (ServiceAppointment s :[select id from ServiceAppointment limit 10])
    { ids.add(s.id);
    }

FSL__CreateFilterEvent__e filterEvent = new FSL__CreateFilterEvent__e();

filterEvent.FSL__ServiceApptIds__c = JSON.serialize(ids);
system.debug(filterEvent.FSL__ServiceApptIds__c);
filterEvent.FSL__FilterName__c = 'My Temporary List';
filterEvent.FSL__Description__c = 'a temporary list that was created to demonstrate this feature';
filterEvent.FSL__FilterCategory__c = 'GENERAL';
filterEvent.FSL__LoadFilterImmediately__c = true;

Database.SaveResult eventRunningResult = EventBus.publish(filterEvent);
```

SEE ALSO:

[Salesforce Field Service: Set Custom Permissions for Field Service](#)

[Salesforce Field Service: Create Custom Actions for the Field Service Dispatcher Console](#)

Field Service Mobile App

Configure mobile-specific features, such as deep linking, for the Field Service app.

IN THIS SECTION:

[Get Started with Lightning Web Components in the Field Service Mobile App](#)

Learn how to create and use Lightning web components (LWCs) in the Field Service mobile app. LWCs offer a versatile and efficient framework to meet the needs of your Field Service operations, from enhancing user interfaces to extending functionality.

[Build Custom Lightning Web Components for Service Documents](#)

Customize templates for your specific business and branding needs by creating custom components with Lightning web components (LWCs).

[Configure Deep Linking for the Field Service Mobile App](#)

Your field technicians rely on their mobile devices to perform complex tasks, and some of those tasks require them to navigate between multiple tabs, flows, or actions. With deep linking, you can send users directly to the right spot in the app.

[Add Lightning Web Components for Plug-Ins to the Field Service Mobile App](#)

Enhance the Field Service mobile app by creating Lightning web components (LWCs) for plug-ins.

Get Started with Lightning Web Components in the Field Service Mobile App

Learn how to create and use Lightning web components (LWCs) in the Field Service mobile app. LWCs offer a versatile and efficient framework to meet the needs of your Field Service operations, from enhancing user interfaces to extending functionality.

This guide is a companion document to the [Mobile and Offline Developer guide](#). Together these guides help you create seamless and intuitive experiences for your field service mobile workers. Familiarize yourself with the content in the Mobile and Offline Developer Guide first, then return here to learn about and get set up for working with the Field Service mobile app.

IN THIS SECTION:

[Considerations for LWC Offline in the Field Service Mobile App](#)

The following considerations apply to LWC Offline when you run your components in the Field Service mobile app.

[Field Service Org Setup](#)

Define and assign the required permission sets.

[iOS Simulator Setup](#)

During development it's convenient to test your code in a virtual device. Use Xcode to create a device simulator, and install the virtual device build of your mobile app into it.

[Android Emulator Setup](#)

During development it's convenient to test your code in a virtual device. Use Android Studio to create a device emulator, and install the virtual device build of your mobile app into it.

[Debug Lightning Web Components in the Field Service Mobile App](#)

The best way to develop and debug your Lightning web components is the same way you develop and debug *anything* built with HTML, CSS, and JavaScript: with the debugging tools built into your web browser.

[What Can I Do with Lightning Web Components in the Field Service Mobile App?](#)

You can create custom components that are accessed from the Actions menu. Actions built with LWCs can be simple or complex, depending on the problem you're solving, and can work even when mobile devices are offline. Use them to add and annotate an image to attach to a service report, order a part, contact a SME, or create a custom dashboard for your mobile workers. Use LWCs in Field Service to do pretty much anything Salesforce can do.

[Configure Offline Mode in the Field Service Mobile App](#)

Learn what to expect when Automatic Offline Mode is turned on in the Field Service mobile app and how you can adjust its settings.

Design with Lightning Web Components to Look Like Salesforce

Elevate your field service operations with visually cohesive mobile experiences. The Field Service mobile app seamlessly integrates the following Lightning web components with the Salesforce UI style.

Considerations for LWC Offline in the Field Service Mobile App

The following considerations apply to LWC Offline when you run your components in the Field Service mobile app.

-  **Note:** Developing with Lightning web components is likely best suited for seasoned developers, although very advanced admins with coding experience can also be successful.
- Global quick actions are available on all pages that have the Actions menu. They do **not** receive the record ID of the current record when invoked from a record detail page.
- A Community license user who opens a service appointment can experience a missing record error. This is a known issue with this specific user type.
- Task objects added to a briefcase aren't primed for offline use. If a briefcase contains a Task object and priming is attempted, an error message is displayed. However, other objects in the briefcase are primed. To resolve the error, remove Task objects from the briefcase.
- There's a conflict between Appointment Assistant and LWC Offline that can result in missing URLs. [See this known issue for details.](#)
- (iOS only) Changes to Lightning web components are loaded into the app only when the app is fully quit and then relaunched (a "cold start").

Field Service Org Setup

Define and assign the required permission sets.

-  **Note:** This step is required for Field Service orgs. If you're using LWC Offline in the Salesforce mobile app, skip to the next step.

Lightning web components for LWC Offline-enabled mobile apps is an opt-in feature. To enable it for your org, you must define and assign the permissions in the steps provided below. These are required to access and run LWCs within the Field Service mobile app. (While not technically required to develop LWCs, you'll have a hard time developing components when you can't run them.)

IN THIS SECTION:

[Define a Permission Set for Your Org](#)

Create a permission set that applies the permissions required to enable Lightning Data Service for Field Service mobile app users.

[Assign the Permission Set to a Mobile User](#)

Assign Lightning Data Service permissions to Field Service mobile app users.

Define a Permission Set for Your Org

Create a permission set that applies the permissions required to enable Lightning Data Service for Field Service mobile app users.

1. From Setup, enter *Permission Sets* in the Quick Find Box and select **Permission Sets**.
2. Click **New**.
 - For **Label**, enter *Field Service - Lightning Data Service*.
 - For **Description**, enter *Assign to Field Service Mobile users to give them permission to use features that use Lightning Data Service*.

- For **License**, select **Field Service Mobile**.
3. Click **Save**.
 4. In the Find Settings box, enter *Lightning SDK for Field Service Mobile* and click it.
 5. Click **Edit**.
 6. Select **Lightning SDK for Field Service Mobile** and save your work.

Assign the Permission Set to a Mobile User

Assign Lightning Data Service permissions to Field Service mobile app users.

Create your permission set.

1. From Setup, in the Quick Find box, enter and select **Permission Sets**.
2. Scroll to the permission set you created, in this case **Field Service - Lightning Data Service**, and click on it.
3. Click **Manage Assignments**, then **Add Assignments**.
4. Select the users you'd like to assign this permission set to.



Note: You can assign an expiration date at which time the permission set is removed for the selected users. If you want to assign the permission set indefinitely, leave **No expiration date** selected.

5. Click **Assign**.
6. On the Assignment Summary screen, click **Done**.

iOS Simulator Setup

During development it's convenient to test your code in a virtual device. Use Xcode to create a device simulator, and install the virtual device build of your mobile app into it.

IN THIS SECTION:

[Configure Minimum Required iOS Simulator Settings](#)

Ensure your virtual device meets the minimum device and iOS version requirements to run your mobile app.

[Install the Field Service App for iOS](#)

To run the Field Service mobile app in an iOS simulator, download and install a virtual device build of the app. After it's installed, open the app and log into your development org.

Configure Minimum Required iOS Simulator Settings

Ensure your virtual device meets the minimum device and iOS version requirements to run your mobile app.

Review the minimum requirements for your mobile app.

- [Field Service Mobile App Requirements](#)
 - [Salesforce Mobile App Requirements](#)
1. Download and install the latest version of [Xcode](#). If you already have Xcode installed, there's no need to reinstall it.
 2. Launch Xcode.
 3. In the menu bar, select **Xcode > Open Developer Tool > Simulator**.
A new program called Simulator opens, displaying a mobile screen.

4. Go to **File > Open Simulator** to choose your preferred device.

If you want to create a simulator, go to **File > New Simulator** and follow the prompts.

After the device simulator launches you can close Xcode, but keep the Simulator app's window open to install the Field Service app in the next section.

Install the Field Service App for iOS

To run the Field Service mobile app in an iOS simulator, download and install a virtual device build of the app. After it's installed, open the app and log into your development org.

1. Download the iOS virtual device build of the Salesforce Field Service mobile app, in the form of a zip file.

A link to download the latest version of the app is available in [this Trailblazer Community post](#). If you encounter an insufficient permissions page, log in to Trailhead and try the link again.

2. Double-click the downloaded zip file to extract the app file.
3. Drag the downloaded `.app` file into the Simulator window.
4. In Simulator, click (a simulated tap in this case) the newly installed Field Service app to open it.

If you don't see the app on the first page, it's probably installed on a different app page. You can swipe the page with your mouse to see additional pages to find the app.

5. Click (simulator tap, you get the idea) the Field Service app. Click **Get Started**.
6. Click the screen to run through the tutorial or click **Skip**.
7. Click **Log In**.
8. Click **I Agree** to agree to the Order Form Supplement agreement.

9. Click  and select a connection.



Warning: If you're logging in as a community user for the first time, click  to add a new connection.

- For **Host**, enter your org's URL in the following format: `https://[yourURL].my.salesforce.com`.
- For **Label**, enter a nickname for your connection.

10. Click **Done**.
11. Enter your username and password for your org.
12. Click **Log In**.
13. Click **Allow** to allow the app to access your Salesforce information.
14. Click through the various permissions screens and allow them the appropriate access.
When finished, you arrive at the app's home screen.

Android Emulator Setup

During development it's convenient to test your code in a virtual device. Use Android Studio to create a device emulator, and install the virtual device build of your mobile app into it.

IN THIS SECTION:

[Configure Minimum Required Android Emulator Settings](#)

Ensure your virtual device meets the minimum device and Android API version requirements to run your mobile app.

[Install the Field Service App for Android](#)

To run the Field Service mobile app in an Android emulator, download and install a virtual device build of the app. After it's installed, open the app and log into your development org.

Configure Minimum Required Android Emulator Settings

Ensure your virtual device meets the minimum device and Android API version requirements to run your mobile app.

Review the minimum requirements for your mobile app.

- [Field Service Mobile App Requirements](#)
- [Salesforce Mobile App Requirements](#)

1. Download and install the latest version of [Android Studio](#).

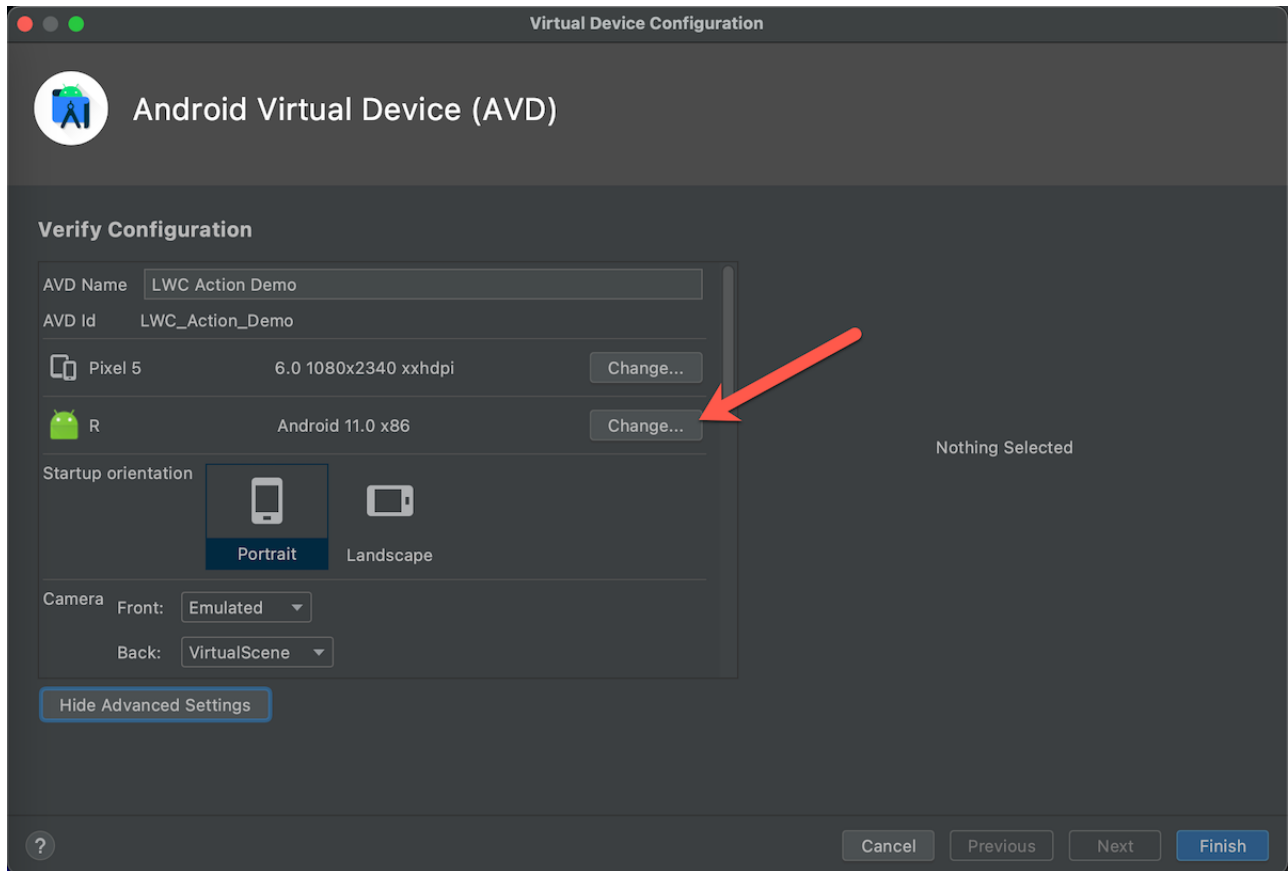
If you already have Android Studio installed, there's no need to reinstall it.

2. Launch Android Studio.

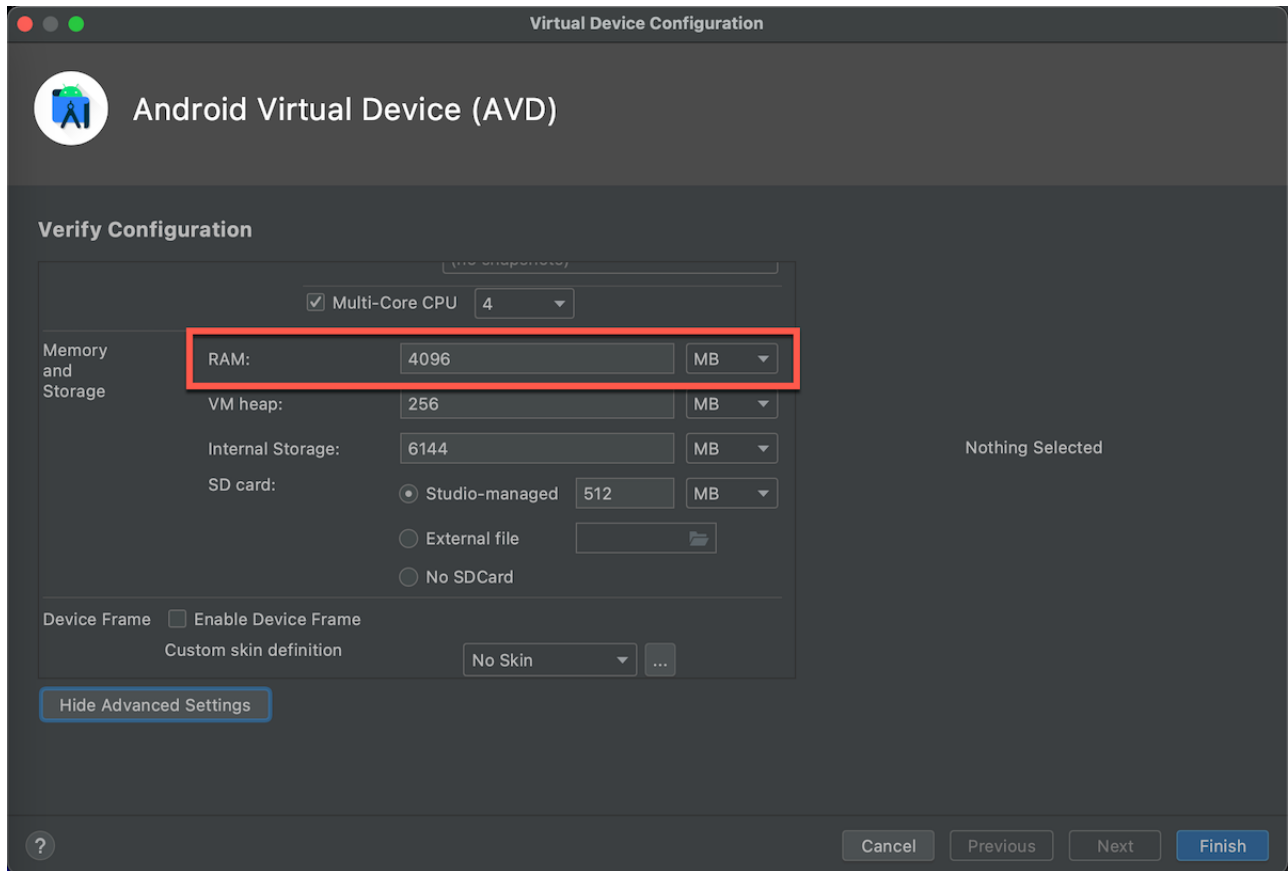
3. Click **More Actions** or  in the top left, depending on your version of Android Studio, and then select **Virtual Device Manager** from the dropdown.

4. Click the  in the Actions column of the device you'd like to edit.
Or, if you want to create an emulator, click **Create Device** and follow the prompts.

5. Click **Change** in the line that displays the version number.



6. In the pop-up window, select API 30 version or later.
7. Click **OK**.
8. Scroll down in the window to the Memory and Storage section.



9. Click in the **RAM** field and enter `4096`.

If you can't change the RAM value, use a newer device for the emulator.

10. Click **Finish**.

11. Click  in the Actions column of the device to launch the Android Emulator.

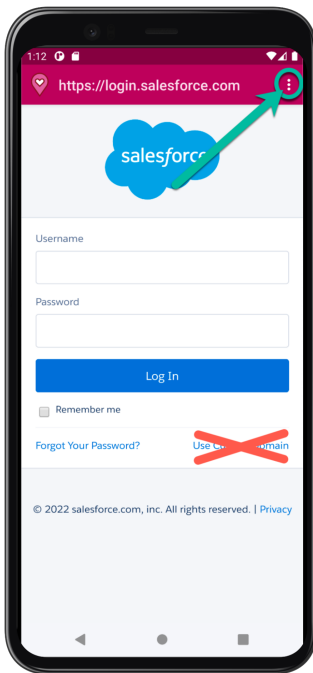
After the device emulator launches you can close Android Studio, but keep the Android Emulator app window open to install the Field Service app in the next section.

Install the Field Service App for Android

To run the Field Service mobile app in an Android emulator, download and install a virtual device build of the app. After it's installed, open the app and log into your development org.

1. Download the Android virtual device build of the Salesforce Field Service mobile app in the form of an APK file.
A link to download the latest version of the app is available in [this Trailblazer Community post](#). If you encounter an insufficient permissions page, log in to Trailhead and try the link again.
2. Drag the downloaded `.apk` file into the Android Emulator window.
3. Click a blank space on the Android Emulator's screen and drag up to view the installed apps.
4. Click the newly installed Field Service app.
5. Click **I Agree** to accept the Order Form Supplement agreement.

- Click the vertical dots button on the top right and select **Change Server**.



- Select a connection.



Warning: If you're logging in as a community user for the first time, click **Add New Connection**, fill in the form as indicated in the following bullet list, and then click **Apply** to save the changes.

- For **Name**, enter a nickname for your connection.
- For **URL**, enter your org's URL. It must be in the following format: `https://[yourURL].my.salesforce.com`

- Click the arrow in the top left to go back to the login screen.
- Enter your username and password for your org.
- Click **Log In**.
- Click **Allow** to allow the app to access your Salesforce information.
- Click through the various permissions screens and allow them the appropriate access.
When finished, you arrive at the app's home screen.

Debug Lightning Web Components in the Field Service Mobile App

The best way to develop and debug your Lightning web components is the same way you develop and debug *anything* built with HTML, CSS, and JavaScript: with the debugging tools built into your web browser.

- For debugging on Android, use [Chrome DevTools](#)
- For debugging on iOS, use [Safari Web Inspector](#)

Next, this section walks you through connecting your desktop browser's developer tools to the WebView within the Field Service mobile app. From there, debugging an LWC running on a mobile device is like any other web app debugging session.

IN THIS SECTION:

[Install Local Development Server Plugin](#)

Whether you're debugging your component for iOS or Android, you must first install the LWC Development Server for mobile.

[Debug in iOS](#)

Connect Safari Web Inspector on your desktop to the WebView in the Field Service mobile app where your LWC is running.

[Debug in Android](#)

Connect Chrome DevTools on your desktop to the WebView in the Field Service mobile app where your LWC is running.

Install Local Development Server Plugin

Whether you're debugging your component for iOS or Android, you must first install the LWC Development Server for mobile.

1. In a terminal window in VS Code or Terminal, run the following command to ensure you're using the latest version of Salesforce CLI.

```
sf update
```



Note: If you encounter an error when updating Salesforce CLI, see [Update Salesforce CLI](#) for troubleshooting instructions.

2. In the same terminal window, run the following command to install the LWC Development Server for mobile.

```
sf plugins install @salesforce/lwc-dev-server
```

Debug in iOS

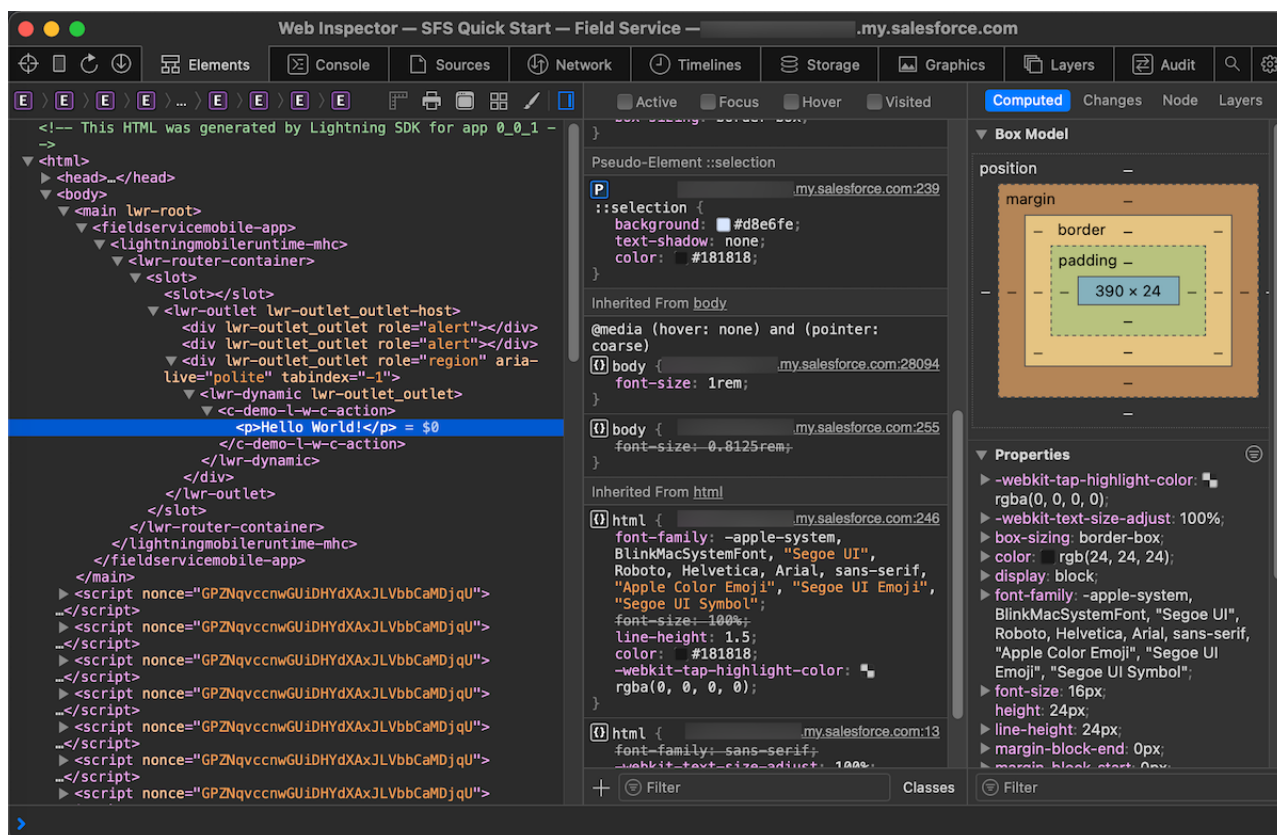
Connect Safari Web Inspector on your desktop to the WebView in the Field Service mobile app where your LWC is running.



Note: Debugging in iOS currently only works with Big Sur or later, and requires using the Safari Technology Preview browser.

1. Launch Safari on your desktop.
2. Select **Safari > Preferences**.
3. Select **Advanced**.
4. Enable **Show Develop menu in menu bar**. Close the Preferences panel.
5. Select **Develop > Simulator - device - version**, where the Simulator is the one that you've opened with the Field Service app for testing your new Lightning web component.

A window appears that shows the Safari Web Inspector developer tools connected to your simulator. You can use these tools and standard web development techniques to refine and improve your component.



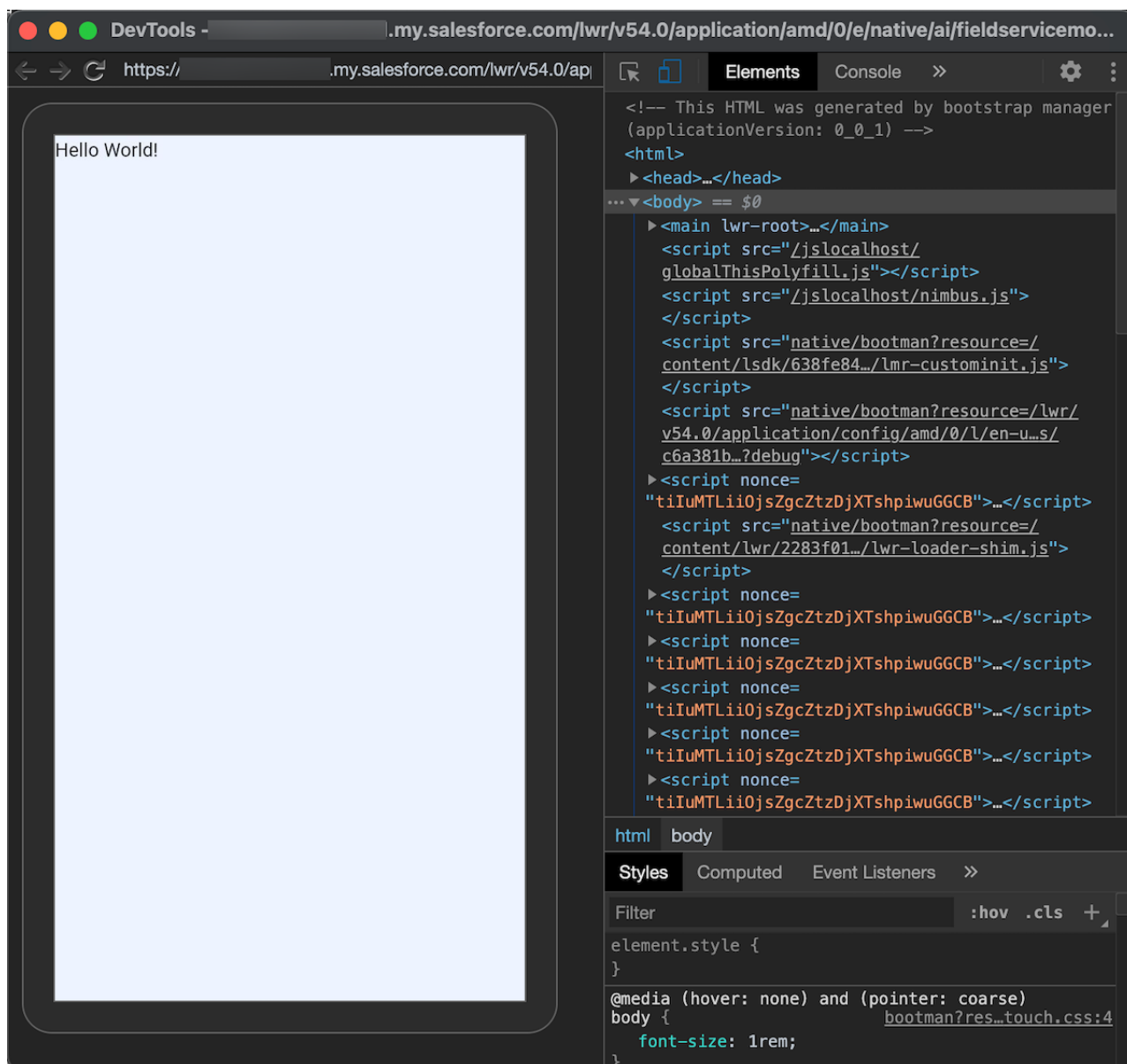
See [Apple Web Development Tools](#) for additional information about the Web Inspector and how to use it.

Debug in Android

Connect Chrome DevTools on your desktop to the WebView in the Field Service mobile app where your LWC is running.

1. On your emulated Android device, open the Settings app.
2. Enter *About emulated device* into the search bar and click it.
3. Scroll to the bottom of the page and click **Build number** seven times.
The message “You are now a developer!” appears when you click it enough, indicating that developer mode is enabled for the emulator.
4. Launch Chrome on your desktop.
5. In the location bar, enter *chrome://inspect/#devices*.
6. Click **Inspect** for the WebView under the Remote Target emulator you’re using.

A window appears containing your emulator with Chrome DevTools connected to it. You can use Chrome DevTools to inspect the LWC element, set breakpoints, and see the console output. As with iOS, you can use the standard web development techniques for working with HTML, JavaScript, and CSS to run, test, debug, and improve your LWC.



See [Remote debugging WebViews](#) for additional basics, and [Debugging embedded JavaScript in an Android app using Chrome DevTools](#) for a deep dive on debugging JavaScript in an embedded WebView of an Android mobile app.

What Can I Do with Lightning Web Components in the Field Service Field Service Mobile App?

You can create custom components that are accessed from the Actions menu. Actions built with LWCs can be simple or complex, depending on the problem you're solving, and can work even when mobile devices are offline. Use them to add and annotate an image to attach to a service report, order a part, contact a SME, or create a custom dashboard for your mobile workers. Use LWCs in Field Service to do pretty much anything Salesforce can do.

IN THIS SECTION:

[Work with LWCs](#)

Use the resources in the following table to learn more about developing with LWCs for the Field Service mobile app.

Work with LWCs

Use the resources in the following table to learn more about developing with LWCs for the Field Service mobile app.

Task	Where to Find It
Learn about and develop with LWCs.	Salesforce Developers: Lightning Web Components Developer Guide
Use the open-source LWCs to find nearby resources, reschedule appointments, and upload and annotate images in the Field Service mobile app.	GitHub: LWC-Mobile-Samples - FindNearby - RescheduleAppointments - UploadAndAnnotateImages
Use the unlocked packages to schedule follow-up appointments and view a mobile dashboard in the Field Service mobile app.	GitHub: LWC-Mobile-Samples - FollowupAppointment - MobileDashboard
Create LWCs that use plug-ins.	Field Service Developer Guide: Add Lightning Web Components for Plug-Ins to the Field Service Mobile App
Find a Partner to help you develop with LWCs.	Find the right Salesforce expertise: Salesforce Partners

Configure Offline Mode in the Field Service Mobile App

Learn what to expect when Automatic Offline Mode is turned on in the Field Service mobile app and how you can adjust its settings.

The Field Service mobile app switches offline when no cellular signal and no wifi connection are available. When the app is in offline mode, it uses cached data, and it refreshes its data when it returns online.

The app determines whether to stay in offline mode by sending regular connection requests to the Salesforce API. After repeated failed requests, it enters or remains in offline mode. This means that in certain situations, the app can be in offline mode even when a cellular signal is available, wifi connection, or both.

1. As an admin, go to the Manage Connected Apps page in Setup.
2. Click **Salesforce Field Service for iOS** or **Salesforce Field Service for Android**. If your team is using both platforms, add a configuration for each.
3. In the custom attribute list, click **New**.
4. Add custom attributes to configure the way the app uses offline mode.
5. After you add custom attributes to configure the app, ask your mobile workers to log out and then log in to the app for the changes to take effect.

You can change the following settings related to offline mode.

Setting	Default Value	Setting Name
Interval of the next Salesforce API connection check after a successful check when the app is online	15 seconds	AO_DELAY_FOR_ONLINE_STATUS
Maximum number of failed requests before the app goes offline	3	AO_CONNECTION_FAILURE_THRESHOLD
Maximum number of successful requests for app to return online	3	AO_CONNECTION_SUCCESS_THRESHOLD
Interval of the next Salesforce API connection check after a failed check	1 second	AO_DELAY_FOR_CHANGING_STATUS
The time the connection request can take before being considered a failed attempt	5 seconds	AO_CONNECTION_CHECK_TIMEOUT

Considerations

- The app can switch between online and offline mode frequently depending on your settings. For example, if you set the Salesforce API connection check to ping every second and configure the app to go offline after one failed check, it can repeatedly switch between online and offline mode.
- Connection checks don't accurately reflect data retrieval calls. As a result, this feature keeps the app online, but data retrieval calls can still be slow.
- If a flow is in progress when the app switches between offline and online mode, the flow can be interrupted.
- If the app is online and you use it in areas of low network connectivity, some users experience long wait times for the data to refresh.

Design with Lightning Web Components to Look Like Salesforce

Elevate your field service operations with visually cohesive mobile experiences. The Field Service mobile app seamlessly integrates the following Lightning web components with the Salesforce UI style.

- Labels for Lightning elements
 - font size
 - Max length of label is three lines
- `lightning-input`
 - Text + Number
 - width
 - height
 - background-color-focus
 - border-color-focus
 - border-width
 - caret-color-focus
 - margin
 - padding
 - background-color-error-focus

- [Datetime Picker](#)
 - width
 - height
 - background-color-focus
 - border-color
- [Checkbox Toggle](#)
 - background-color-checked-focus
 - background-color-checked
 - shadow
- [Checkbox Button](#)
 - background-color
- [lightning-textarea](#)
 - width
 - height
 - background-color-focus
 - border-color-focus
 - border-width
 - caret-color-focus
 - margin
 - padding
 - background-color-error-focus
 - font size
- [lightning-combobox](#) (picklist)
 - width
 - height
 - background-color-focus
 - border-color-focus
 - border-width
 - margin
 - padding
- [lightning-checkbox-group](#)
 - border around checkbox
 - label font size
 - checkbox size
 - checkbox color border
 - checkbox color background
 - checkbox mark color
 - error state background focus

- [lightning-radio-group](#)
 - border around checkbox
 - label font size
 - radio size
 - radio color border thickness
 - radio mark color
- [lightning-progress-bar](#)
 - progress bar color
- [lightning-progress-indicator](#)
 - progress indicator bar color
- [lightning-button](#) (variant - base, neutral, brand)
 - border
 - background-color
 - background-color-focus

Build Custom Lightning Web Components for Service Documents

Customize templates for your specific business and branding needs by creating custom components with Lightning web components (LWCs).

If you need a component for your document template that isn't available in Document Builder standard components, you can easily customize templates for your specific business and branding needs. Creating custom components is simple with LWCs. LWCs for service documents require specific print styling, so if you built custom components for other platforms, such as Field Service Mobile, you must make components for Document Builder.

IN THIS SECTION:

[Lightning Web Components for Document Builder: Requirements](#)

Here are the requirements for creating Lightning web components (LWCs) for Document Builder.

[Lightning Web Components for Document Builder: Considerations](#)

Considerations and limitations for building Lightning web components (LWCs) for the Document Builder feature, including a list of supported and unsupported utilities, HTML components, and APIs.

[Lightning Web Components for Document Builder: Offline Priming](#)

Ensure that your custom Lightning web components (LWCs) for Document Builder work when your mobile workers are offline.

[Build Custom Tables for Service Documents with LWCs](#)

Build custom tables for Document Builder with Lightning web components (LWCs).

[Code Examples for Document Builder](#)

Build custom components for Document Builder with Lightning web components (LWCs).

Lightning Web Components for Document Builder: Requirements

Here are the requirements for creating Lightning web components (LWCs) for Document Builder.

Use Salesforce Lightning Design System (SLDS) wire adapters

- Don't use aura controllers.
- Don't use Apex. While Apex works online, it returns an empty array when offline.
- Calls to a third-party data source to retrieve data don't work offline.

Use wire adapters for create, update, and delete operations

You can use only these operations, which are logically primed: create, update, delete, `getRecord`, `getRecords`, `getObjectInfo`, and `graphql`. Use of `graphql` is recommended. `getRelatedListRecords` is not fully supported for offline use.

You can use these wire adapters to create, update, delete, and get records.

- `import { getRecord } from 'lightning/uiRecordApi'`
- `import { createRecord } from 'lightning/uiRecordApi'`
- `import { deleteRecord } from 'lightning/uiRecordApi'`
- `import { updateRecord } from 'lightning/uiRecordApi'`
- `import { getObjectInfo } from 'lightning/uiObjectInfoApi'`
- `import { gql, graphql } from 'lightning/uiGraphQLApi'`

Remove Aura dependencies

Field Service developers using the Lightning Software Developers Kit (LSDK) can't use any code that relies on Aura because Field Service must be able to execute offline. To do so, use only LWCs.

LWCs that depend on Aura run into trouble at run time. To avoid problems, you must fully evaluate your component's dependencies and go all the way down the chain, ensuring that no code uses Aura.

Ensure LWCs Are Statically Analyzable

If your LWC isn't statically analyzable, also known as Komaci-friendly, Komaci doesn't prime the LWC. Komaci is a suite of packages that provides data flow graph generation and processing capabilities. It defines a schema for an Abstract Data Graph (ADG), which is a serializable metadata-based representation of a data flow graph. Komaci includes an ADG generation service. It also includes a framework for building plug-ins to read, process, and transform an ADG and a runtime engine for hosting an ADG with live inputs and full reactivity.

Komaci provides an abstraction layer for the data flow through an LWC component tree. It includes a utility to analyze an LWC module and generate an ADG and runtime helpers for runtime integration with LWC wires. It also provides an extension for LWC applications to apply progressive data resolution and integrate with the Komaci engine.

 **Important:** For custom LWCs to appear on mobile document preview, you must include this code in your `js-meta.xml` file:

```
<supportedFormFactor type="Small" />
```

Lightning Web Components for Document Builder: Considerations

Considerations and limitations for building Lightning web components (LWCs) for the Document Builder feature, including a list of supported and unsupported utilities, HTML components, and APIs.

Supported Utilities

Here are the supported utilities for building LWCs for Document Builder.

- [Borders](#)

- [Box](#)
- [Description List](#)
- [Grid](#)
- [Horizontal List](#)
- [Hyphenation](#)
- [Layout](#)
- [Margin](#)
- [Name Value List](#)
- [Padding](#)
- [Print](#)
- [Text](#)
- [Themes](#)
- [Vertical List](#)

Supported Base Components

Here are the supported base components for building LWCs for Document Builder.

- [Combobox](#)
- [Formatted Date-Time](#)
- [Formatted Location](#)
- [Formatted Phone](#)
- [Formatted Time](#)
- [Formatted URL](#)
- [Relative Date-Time](#)
- [Formatted Address](#)



Tip: You can't use `showMapLink` to show Google maps in an `iframe`

- [Formatted Name](#)
- [Formatted Number](#)
- [Formatted Rich Text](#)
- [Formatted Text](#)
- [Progress Indicator](#)
- [Progress Ring](#)
- [Tile](#)

Unsupported Salesforce APIs

For consistency between desktop, mobile, and print, don't use [Form Factor GVP](#) or [Width Aware](#) Salesforce APIs.

Unsupported HTML Concepts

You can't use these HTML concepts when building LWCs for Document Builder.

- Don't use iframes.
Iframes don't work offline, and page numbers don't show up properly.
- Don't use CSS Screen media queries.
CSS Screen media queries cause discrepancies in how your custom component looks on desktop and mobile.
- Don't use CSS `position: fixed`.
When content is printed, fixed positioning is irrelevant, and elements with `position: fixed` don't behave as expected. If you want repeating elements, drag an object into the header or footer sections of the canvas.
- Don't use CSS to show objects side by side.
To show objects side by side, use the standard grid component. Avoid CSS properties such as `float`, `display: flex`, `display: inline`, `display: inline-block`, `display: inline-flex`, `display: grid`, `display: inline-grid`, `display: table`, or `display: table-row`.
- Don't use `:nth-of-*` CSS Selectors.
When working with content that spans multiple pages, pseudo-selectors such as `:nth-of-type` or `:nth-child` typically don't work. As a workaround, use `lwc:if` templates and application of classes to customize the style of your service document. Learn more about `lwc:if` in the [Lightning Web Components Developer Guide](#).

Pagination Considerations

When creating LWCs for Document Builder, consider these pagination limitations.

- Service documents are limited to 250 pages. Document Builder can't process a document with more than 250 pages into a PDF.
- Avoid CSS page declarations. Don't declare `@page` on custom components. The ability to modify page formatting for print is built into Document Builder.
- If an image is too large to fit onto the remainder of a page, it's pushed to the next page. If an image, by itself on a page, is still too large to fit, it's resized to fit on one page.
- If the page number component is dragged onto the body section of the page, it shows the page number in which it shows up. We recommend that you use this component in header or footer regions.
- The standard signature component isn't split across pages.
- If the field component spans pages, the field labels are inline and repeat.
- Column headers repeat if related list content spans across pages.
- Use the standard grid component for creating layouts because it provides comprehensive layout options. If you must use SLDS grid classes, don't apply `slds-col` and `slds-grid` CSS classes to a single element. Instead, apply each layout class to its own element. Here's an example.

```
<!-- Bad --><div class="slds-col slds-grid"></div><!-- Good --><div class="slds-col"><div class="slds-grid">
```

The simpler the grid, the better. Features such as column wrapping, reordering, gutters, and vertical grids aren't supported.

- PDF conversion doesn't compress static resources. Don't place large static resources (`contentAssets`) on the page. Resize and compress them before using them on a service document.

Lightning Web Components for Document Builder: Offline Priming

Ensure that your custom Lightning web components (LWCs) for Document Builder work when your mobile workers are offline.

To ensure that your custom components have offline support, install the Komaci VS Code Plug-in. With the plug-in, you can analyze and identify which of your custom LWC components mobile workers can use offline. Without the plug-in, you can't tell whether a component works while a mobile device is offline. When a component can be used offline, mobile workers can still capture data, even when they don't have access to a network. Before the user goes offline, we cache the data required to make the component functional. Then, when the user has wireless access again, we sync the data that's captured while mobile devices are offline.

Install the Komaci ESLint Visual Studio Code (VS Code) Plug-in

We suggest that you use VS Code. If you're using VS Code, install the ESLint VS Code plug-in, and then press CTRL+SHIFT+P to open the command palette to ESLint: Restart ESLint Server. Then, in the bottom output tab (with ESLint) selected, you can debug it.

You can also run `yarn lint:lwc` inside VS Code's terminal.

If you're not using VS Code, check out the [public repository](#) of the ESLint Komaci Plug-in that can configure your tool.

If you want a deep dive, check out [npm](#) for information about the plug-in, including code. But if you're ready to implement the code you can use either of these commands.

- Command 1: `yarn add --dev @salesforce/eslint-plugin-lwc-graph-analyzer`
- Command 2: `npm install --save-dev @salesforce/eslint-plugin-lwc-graph-analyzer`

After you input the code, check your project directory and update the `.eslintrc` file to reference our recommended configuration.

The Komaci engine is fully portable to any JavaScript runtime environment and has no dependencies on web-based technologies or APIs, such as the DOM. You can use Komaci as an independent data fetching service for LWC applications. The engine can analyze LWC modules and prefetch their respective data dependencies in a host environment outside of a runtime window, such as a [Web Worker](#), [JavaScriptCore](#), or [V8](#) host environment.

Build Custom Tables for Service Documents with LWCs

Build custom tables for Document Builder with Lightning web components (LWCs).

While Document Builder offers a standard component to create tables for related records, you can also build your own custom tables to suit your company's needs.

Design Your Table to Show Data Properly

To ensure that your tables show data properly, use HTML tables to build your LWC for custom tables. The right HTML table formatting organizes your data properly, even when the table overflows or breaks between pages. If information in the table does overflow, column headers repeat on the next page.

Here's a sample markup that calls out the header row, ensuring that the header is repeated across pages.



Example:

```
<table>
  <thead>
    <tr>
      <th>Company</th>
      <th>Contact</th>
      <th>Country</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Alfreds Futterkiste</td>
```

```

        <td>Maria Anders</td>
        <td>Germany</td>
    </tr>
    <tr>
        <td>Centro comercial Moctezuma</td>
        <td>Francisco Chang</td>
        <td>Mexico</td>
    </tr>
</tbody>
</table>

```

Code Examples for Document Builder

Build custom components for Document Builder with Lightning web components (LWCs).

With custom components, the sky's the limit, but these code samples can get you started on your own Document Builder customization journey.



Important: For custom LWCs to appear on mobile document preview, you must include this code in your `js-meta.xml` file:

```
<supportedFormFactor type="Small" />
```

Static Text

The header and medium header base components come with color and padding configuration. This code creates a medium header component.

```

// Below is the code for the HTML block:
<template>
    <h2 class="slds-text-heading_medium" style={inlineStyle}>{text}</h2>
</template>

// Below is the code for the javascript block:
import { LightningElement, api } from "lwc";

export default class MediumHeaderText extends LightningElement {
    @api text;
    @api colorhex;
    @api topPadding;
    @api leftPadding;
    @api rightPadding;
    @api bottomPadding;

    get inlineStyle() {
        return `color:#{this.colorhex};
padding:${this.topPadding}px ${this.rightPadding}px ${this.bottomPadding}px
${this.leftPadding}px`;
    }
}

// Below is the extensible markup language block
<?xml version="1.0" encoding="UTF-8" ?>
<LightningComponentBundle xmlns="http://soap.sforce.com/2006/04/metadata">
    <apiVersion>57.0</apiVersion>

```

```

<isExposed>true</isExposed>
<targets>
  <target>lightning__ServiceDocument</target>
</targets>
<targetConfigs>
  <targetConfig targets="lightning__ServiceDocument">
    <supportedFormFactors>
      <supportedFormFactor type="Large" />
    </supportedFormFactors>

    <property name="text" type="string" default="A medium header" />
    <property name="colorhex" type="string" default="1B3971" />
    <property name="topPadding" type="string" default="0" />
    <property name="rightPadding" type="string" default="0" />
    <property name="leftPadding" type="string" default="0" />
    <property name="bottomPadding" type="string" default="0" />
  </targetConfig>
</targetConfigs>

</LightningComponentBundle>

```

Static Images

Here's some sample code to create custom static images. This code creates a Static Image component.

Important: Don't use static resources, as they aren't supported offline and are deprecated. Only use ContentAsset in your LWC. See [Using Assets in LWC](#), [How to Create Assets](#), and [Viewing and Editing Assets](#).

```

// Below is the HTML block
<template>
  <div>
    <div class="image" style={inlineStyle}>
      <img src={salesforceUrl} />
    </div>
  </div>
</template>

// Below is the JavaScript block
import { LightningElement, api } from "lwc";
import SALESFORCE_LOGO from "@salesforce/contentAssetUrl/salesforce";

export default class DemoSalesforceLogo extends LightningElement {
  @api width;
  @api height;

  // Expose the static resource URL for use in the template
  get salesforceUrl() {
    return SALESFORCE_LOGO;
  }

  get inlineStyle() {
    return `width: ${this.width}px;height: ${this.height}px`;
  }
}

```

```
// Below is the Extensible Markup Language block
<?xml version="1.0" encoding="UTF-8" ?>
<LightningComponentBundle xmlns="http://soap.sforce.com/2006/04/metadata">
  <apiVersion>54.0</apiVersion>
  <isExposed>true</isExposed>
  <targets>
    <target>lightning__ServiceDocument</target>
  </targets>
  <targetConfigs>
    <targetConfig targets="lightning__ServiceDocument">
      <supportedFormFactors>
        <supportedFormFactor type="Large" />
        <supportedFormFactor type="Small" />
      </supportedFormFactors>

      <property name="width" type="integer" />
      <property name="height" type="integer" />
    </targetConfig>
  </targetConfigs>
</LightningComponentBundle>
```

This code uses SFDX to deploy contentAssets.

```
// Below is the XML block:
<?xml version="1.0" encoding="UTF-8" ?>
<ContentAsset xmlns="http://soap.sforce.com/2006/04/metadata">
  <isVisibleByExternalUsers>true</isVisibleByExternalUsers>
  <language>en_US</language>
  <masterLabel>salesforce</masterLabel>
  <relationships>
    <organization>
      <access>VIEWER</access>
    </organization>
    <workspace>
      <access>INFERRED</access>
      <isManagingWorkspace>true</isManagingWorkspace>
      <name>sfdc_asset_company_assets</name>
    </workspace>
  </relationships>
  <versions>
    <version>
      <number>1</number>
      <pathOnClient>salesforce.png</pathOnClient>
    </version>
  </versions>
</ContentAsset>
```

Configure Deep Linking for the Field Service Mobile App

Your field technicians rely on their mobile devices to perform complex tasks, and some of those tasks require them to navigate between multiple tabs, flows, or actions. With deep linking, you can send users directly to the right spot in the app.

Deep linking enables mobile apps to interact with each other using links called URIs, or Uniform Resource Identifiers. URIs have the power to point users to a specific location in a mobile app. Field Service has a pre-defined URI scheme that allows users to launch the Field Service app from a link in an email, website, or third-party application. An administrator or developer can create custom URIs to display specific tabs in the Field Service app. For example, you could integrate a third-party app that your field technicians use for billing.

If an unauthenticated user taps on a Field Service URI, they will be taken to the login screen. Once they log in, they need to tap the URI again to view the linked page.

IN THIS SECTION:

[Deep Linking Schema for the Field Service Mobile App](#)

Use the predefined URI schema for the Field Service mobile app on Android and iOS devices.

[Hide Deep Linking Security Dialog](#)

Each time a user opens a deep link to an action from the Field Service mobile app, a security dialog prompts the user to confirm the action. You can hide this Launch action? dialog by configuring the deep link URL with a security key.

[Pass Data to an LWC with Deep Linking](#)

Pass data between Lightning web components (LWC) or from external apps to an LWC with deep linking. LWC with deep linking is available for the Field Service mobile app on both Android and iOS devices.

Deep Linking Schema for the Field Service Mobile App

Use the predefined URI schema for the Field Service mobile app on Android and iOS devices.

URI Schema Format

The available URI schema formats are:

- `<com.salesforce.fieldservice>://v1/sObject/<id>/<action>?<params>`
- `<com.salesforce.fieldservice>://v1/globalaction/<api_name>?<params>`

Let's break down each parameter of the URI scheme format.

- `com.salesforce.fieldservice`—The supported Salesforce schema name.
- `v1`—The current version of deep linking. The version is a static value.
- `sObject`—Indicates a single instance of a Salesforce object. The schema supports all object types that are supported in the Field Service app, such as work order, service appointment, and assets.
- `id`—The unique single instance of a Salesforce object. `id` lengths of 15 and 18 characters are supported.
- `action`—A supported URI schema operation. Valid actions are listed in the Supported URI Schemes section. If no action is specified, URIs direct the work orders, service appointments, and other object types to the Overview or Details tab.
- `globalaction`—The name of the global quick action.
- `api_name`—Some deep links require an API name in order to perform their function. For instance, the quick action URI requires a quick action API name, and the flow URI requires a flow API name.
- `params`—The field and value pairs used in the global quick action.

Supported URI Schemes

`com.salesforce.fieldservice://v1/sObject/<id>`

Links to the Overview or Details tab.

- Work order and work order line item IDs are directed to the Overview tab.

- Service appointment IDs are directed to the Details tab, with these exceptions.
 - If the service appointment parent is a work order, the deep link is directed to the work order Overview tab for iOS and the service appointment Details tab for Android.
 - If the service appointment parent is a work order line item, the deep link is directed to the work order line item Overview tab for iOS and the service appointment Details tab for Android.
- All other object types are directed to the object's Details tab.

Example: `com.salesforce.fieldservice://v1/sObject/0W05500000001UZGAA`

This example links to the Overview tab for a work order.

com.salesforce.fieldservice://v1/sObject/<id>/details

Links to the Details tab for all object types.

Example: `com.salesforce.fieldservice://v1/sObject/0W05500000001UZGAA/details`

This example links to the Details tab for a work order.

com.salesforce.fieldservice://v1/sObject/<id>/related

Links to the Related tab for all object types.

Example: `com.salesforce.fieldservice://v1/sObject/0W05500000001UZGAA/related`

This example links to the Related tab for a work order.

com.salesforce.fieldservice://v1/sObject/<id>/products

Links to the Products tab for all applicable object types. If an object doesn't have a Products tab, it links to the Details tab instead.

Example: `com.salesforce.fieldservice://v1/sObject/0W05500000001UZGAA/products`

This example links to the Products tab for a work order.

com.salesforce.fieldservice://v1/sObject/<id>/feed

Links to the Feed tab for all applicable object types. If an object doesn't have a Feed tab, it links to the Details tab instead.

Example: `com.salesforce.fieldservice://v1/sObject/0W05500000001UZGAA/feed`

This example links to the Feed tab for a work order.

com.salesforce.fieldservice://v1/sObject/<id>/location

Links to the Location tab for all applicable object types. If an object doesn't have a Location tab, it links to the Details tab instead.

Example: `com.salesforce.fieldservice://v1/sObject/0W05500000001UZGAA/location`

This example links to the Location tab for a work order.

com.salesforce.fieldservice://v1/sObject/<id>/edit

Links to the edit page for the specified Salesforce record.

Example:

`com.salesforce.fieldservice://v1/sObject/0W05500000001UZGAA/edit?Subject=Example%20Subject`

This example links to the edit page for this record and fills the Subject field with the Example Subject value.

com.salesforce.fieldservice://v1/sObject/<id>/flow/<api_name>

Launches a Field Service Mobile Flow from within the mobile app. For example, you can use this URL to launch a custom service closure flow. To use this feature, the app extension must reference this flow. You can find the extension in Setup under Field Service Mobile Settings. Verify that an extension in the App Extension section includes this flow. To learn more, see [Connect a Flow to the Field Service Mobile App](#). Use the URL-encoded name of the extension in the URI.

Example:

`com.salesforce.fieldservice://v1/sObject/0W05500000001UZGAA/flow/service_response_flow?Subject=Example%20Subject`

This example launches the “service_response_flow” flow passing in “Example Subject” as the value for the “Subject” input variable.

com.salesforce.fieldservice://v1/sObject/<id>/quickaction/<api_name>

Launches a quick action from within the mobile app. These quick action types are supported: Create a Record, Update a Record, and Field Service Mobile Extension. You can pass inputs into the quick action using parameters on this URL.

Example:

`com.salesforce.fieldservice://v1/sObject/0W05500000001UZGAA/quickaction/close_order`

This example launches the `close_order` quick action.

com.salesforce.fieldservice://v1/globalaction/<api_name>?<params>

Launches a global quick action from within the mobile app. These global quick action types are supported: Create a Record, Update a Record, and Field Service Mobile Extension. You can pass inputs into the global quick action using parameters on this URL.

Example:

`com.salesforce.fieldservice://v1/globalaction/Create_Work_Order?Subject=Example%20Subject`

This example launches a global quick action to create a work order.

com.salesforce.fieldservice://v1/login/<server_name>/<server_url>

Adds a connection to a server using the supplied URL. To access the server, users click the deep link and enter their existing login credentials. The server is saved to the Change Server list, which gives users quick access because it eliminates the need for repeated log-in attempts.

Parameters:

- `<server_name>` is a user-friendly name used to identify the server. Avoid spaces, punctuation, and other “illegal” characters, especially if workers must log in to multiple sites.
- `<server_url>` is the URL used by the Salesforce Field Service Mobile app to log in. Don’t include “http” or “https” when specifying the URL.

Example: `com.salesforce.fieldservice://v1/login/employee/my.site.com/contractor/login`

This example sets the login URI, sends users directly to `https://my.site.com/contractor/login`, and saves it in the Change Server list as `employee`.

Security Considerations:

- Ensure that you trust the source of a link before clicking it, especially when adding usernames and passwords. For this reason, we recommend that your workers are trained on best practices and that the login link isn’t sent directly via email.
- In many email clients and messaging apps, regardless of the mobile operating system, the text body doesn’t support active links to mobile apps. Opening the app when a link is launched is supported, but it depends on your technology and distribution choices. If you encounter issues with your link, first, try testing it in a web browser. If the link opens in the app, it is formatted properly. In cases where the web link works but your chosen distribution method doesn’t, find a different way to share the link. For a seamless experience, use a web link that redirects to the app works. Other options like QR codes, website link, PDF, or text message work well, depending on the app and the operating system used to open the link. It isn’t guaranteed that all third-party apps on iOS and Android support opening the links.

Parameter Passing for Deep Linking

To pass parameters in a deep linking schema such as a quick action or flow URL, these parameter types are valid.

- URL-encoded text parameters. For example, `Short%20text%20input`.
- Numbers or currency. For example, `1` or `1.45`.
- Boolean values. For example, `true` or `false`. These values aren’t case-sensitive. If no parameter is passed, the default is `false`.

- URL-encoded date and dateTime parameters. For example, `2019-12-11T17%3A01%3A00.000%2B0000`. Include a URL-encoded timezone, such as +0000 for GMT or -0400 for AST.

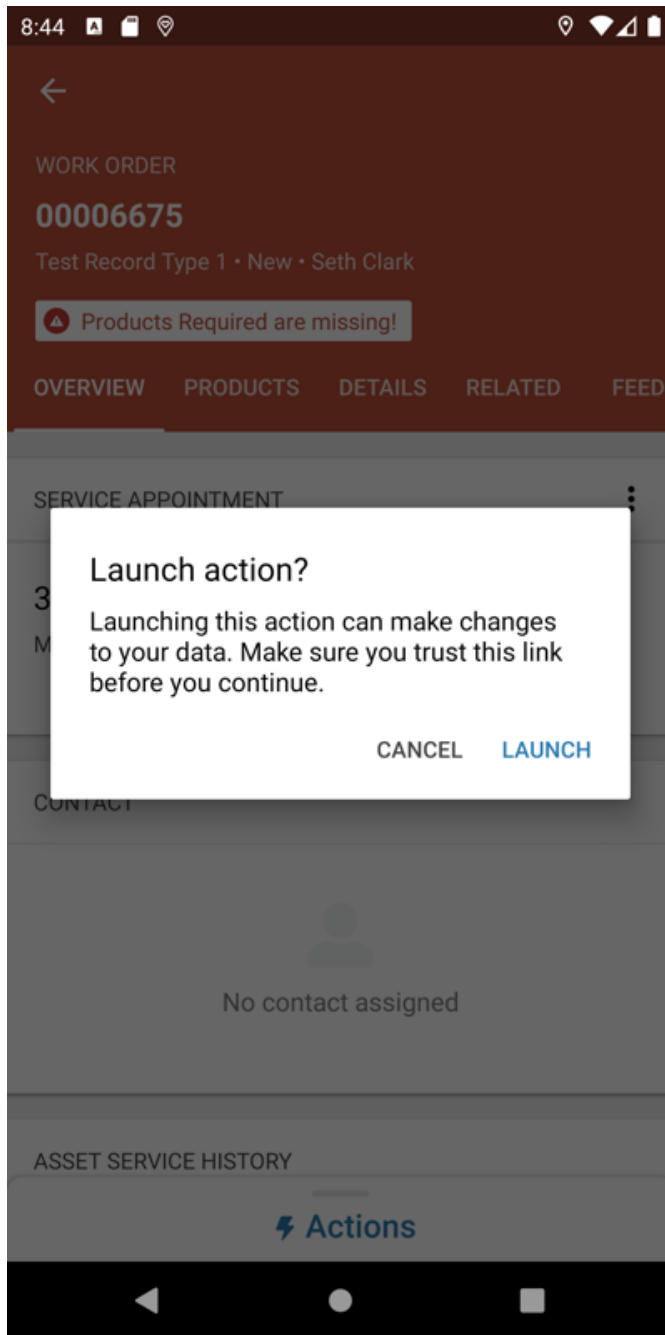
Other parameters aren't supported in deep linking URI schema.

To pass more than one parameter, use an ampersand (&) between them. For example, `firstname=John&lastname=Doe`.

We don't limit the number of parameters, and each parameter value can contain up to 100,000 characters. While deep links can encode up to 1 MB, it's best practice to encode smaller amounts of data.

Hide Deep Linking Security Dialog

Each time a user opens a deep link to an action from the Field Service mobile app, a security dialog prompts the user to confirm the action. You can hide this Launch action? dialog by configuring the deep link URL with a security key.



Complete these steps to configure the security key.

Step 1: Create a private and public key

1. For Linux/Mac, run these commands in the terminal to create a private and public key. The .pem key files are created in the folder where you execute the commands.



Note: We don't officially support Apex for signing the keys, but if you're signing deep links in Apex, try using the ECDSA-SHA256 algorithm.

```
openssl ecparam -genkey -name prime256v1 -noout -out private.pem
openssl ec -in private.pem -pubout -out public.pem
```

- Keep a copy of the private.pem and public.pem files so that you can sign URLs in the future without generating new key pairs. The private.pem file contains the security key.
- Open the public.pem file, and copy the public key, excluding the header and footer and without spaces or a new line. Here's a sample public key.

```
MFkwEwYHKoZIzj0CAQYIKoZIzj0DAQcDQgAEkvkDcFieJenYABN8wOLlE2VomNt2
9/tcTyj+B06ZndRkTjs7+XwrjHe/wOZvjkdYvewhIBYLI6uDTYZixDhO1A==
```

- Copy the public key into the Advanced Permissions section of the Field Service Settings UI. This step is required to turn on this feature.

Step 2: Sign the deep link URL

- In the terminal, change the directory to the folder containing the private.pem file.
- Generate the deep link URL, and copy it to your clipboard.

```
// Base URL.
com.salesforce.fieldservice://v1/sObject/<id>/<action>

// URL with additional query parameters.
com.salesforce.fieldservice://v1/sobject/<id>/<action>?param1=value1&param1=value2
```

- To create a signature using the private key, run this command.

```
pbpaste | openssl dgst -sha256 -sign private.pem | openssl base64 | tr '/+' '_-' | tr
-d '=' | tr -d '\n' | pbcopy
```

The command creates a signature using OpenSSL and Base64 encoding. Here's a breakdown of the command parts.

- The `pbpaste` command pastes from the pasteboard.
- The `openssl dgst` command creates a hash with the sha256 algorithm signed by the private key.
- The `openssl` command makes that the output base64 encoded.
- The `tr` command replaces "/" characters with "_-", and then deletes all "=" characters and new line characters "\n".

- The `pbcopy` command copies the output back to the pasteboard.

If you're using a dynamic deep link URL, generate a new signature for each unique set of URL parameter values. The signature is unique for a URL string. For example, if the `<id>` parameter is dynamic, generate a new signature when the `<id>` value changes.

4. Sign the deep link URL by adding the query parameter `__signature=<sig>` to the end of the URL. Replace `<sig>` with the signature, which you can paste from your clipboard. If you have additional query parameters, place the signature parameter after them.

```
// Signed URL.
com.salesforce.fieldservice://v1/sObject/<id>/<action>?__signature=<sig>

// Signed URL with additional query parameters.
com.salesforce.fieldservice://v1/sObject/<id>/<action>?param1=value1&param1=value2&__signature=<sig>
```

Now you can send the deep link URL with the signature as before. When the technician clicks the URL, the deep link opens in the Field Service app. The app verifies the signature using the public key. If the public key hash matches, the URL is loaded without the Launch action? dialog. If the public key hash doesn't match, the user must confirm the action.

Pass Data to an LWC with Deep Linking

Pass data between Lightning web components (LWC) or from external apps to an LWC with deep linking. LWC with deep linking is available for the Field Service mobile app on both Android and iOS devices.

To pass parameters using deep linking, such as passing data into an LWC form, use the quick actions URI schema with your LWC quick action as the `<api_name>`, followed by the parameter key and value pairs.

```
com.salesforce.fieldservice://v1/sObject/<id>/quickaction/<api_name>?<parameterKey1>=<parameterValue1>&<parameterKey2>=<parameterValue2>&...
```

Here's an example of a deep link URL that passes the first and last name of Jane Doe to an LWC using a quick action called `LWC_Pass_Fields`.

```
/quickaction/LWC_Pass_Fields?FirstName=Jane&LastName=Doe
```

To pass the data, also update the LWC source code of the current page reference, the LWC that the deep link opens. Here's an example of an LWC that passes the parameter value into the corresponding parameter key of the current page reference.

```
import {CurrentPageReference} from 'lightning/navigation';

// Declare the variable for the parameter value.
parameterValue;

// Call the page reference that describes the current LWC page.
@wire(CurrentPageReference)
setCurrentPageReference(currentPageReference) {
  // Pass parameter values using the currentPageReference state attribute.
  // Replace <parameterKey> with the parameter key name used in the deep link URL.
  this.parameterValue = currentPageReference.state.<parameterKey>;
}
```

Add Lightning Web Components for Plug-Ins to the Field Service Mobile App

Enhance the Field Service mobile app by creating Lightning web components (LWCs) for plug-ins.

- [Access a Mobile Device's Biometrics Capabilities](#) (Mobile and Offline Developer Guide)

A Lightning web component can use a device's biometrics functionality to prompt a user to confirm their identity. When these biometrics-related actions occur, the result is returned to the Lightning web component that invoked it. Biometrics checks are managed locally on the mobile device, and don't need a network connection. However, `BiometricsService` requires access to platform-specific APIs that are available only within compatible Salesforce mobile apps.

- **Scan Documents on a Mobile Device** (Mobile and Offline Developer Guide)

A Lightning web component can use a device's camera and optical character recognition (OCR) to scan documents. When a document is successfully scanned, text data extracted from the scanned document is returned to the Lightning web component that invoked it. `DocumentScanner` recognizes machine printed text, not handwriting.

- **Monitor Geofence Regions on a Mobile Device** (Mobile and Offline Developer Guide)

A Lightning web component can use a mobile device's location features to determine a user's current location to the user's proximity to areas that may be of interest, or to perform location-related tasks. The longitude, latitude, and radius define a geofence around the regions of interest. Geofence location is determined locally on the mobile device, and doesn't need a network connection. However, `GeofencingService` does require a GPS signal from the device. For Android devices, Google Location Accuracy must be enabled in the system settings. `GeofencingService` does require access to platform-specific APIs that are available only within compatible Salesforce mobile apps.

- **Use Location on a Mobile Device** (Mobile and Offline Developer Guide)

A Lightning web component can use a mobile device's location features to determine the current location of the device and, by association, the person who is holding it. You can access the device's current location at a moment in time, or you can subscribe to location changes, and receive updates to the device's location when it changes significantly. Location is determined locally on the mobile device, and doesn't need a network connection. `LocationService` does require access to platform-specific APIs that are available only within compatible Salesforce mobile apps.

- **Interact with NFC Tags on a Mobile Device** (Mobile and Offline Developer Guide)

A Lightning web component can use a device's native NFC functionality to read, erase, and write to NFC tags. When an NFC operation is successful, the text data extracted from the NFC tag or a simple success message is returned to the Lightning web component that invoked it.

- **Accept On-Site Payments with Tap-to-Pay** (Mobile and Offline Developer Guide)

A Lightning web component can use a mobile device to let your customers use the Tap-to-Pay capability of the Payments plug-in to pay mobile workers directly. The Field Service mobile app then integrates with Pay Now to connect the Lightning web component to a secure payment system that processes the interaction. The `PaymentsService` plugin allows Field Service mobile workers to collect payments from their customers using Tap to Pay. This service integrates with Salesforce Payments and Stripe as a payment provider. Tap to Pay is supported on iOS and Android devices with Stripe as the payment provider.

IN THIS SECTION:

[Scan Barcodes on a Mobile Device](#)

A Lightning web component can use a mobile device's camera and mobile OS platform features to scan a barcode, such as a UPC symbol or QR code. When a barcode is successfully scanned, the data that was read from the barcode is returned to the Lightning web component that invoked it.

[Capture a Space on a Mobile Device](#)

Create a Lightning web component (LWC) that uses the AR SpaceCapture plugin to create a 2D or 3D model of a room. The model is created by interactively scanning the physical environment using the device's Lidar capabilities.

Scan Barcodes on a Mobile Device

A Lightning web component can use a mobile device's camera and mobile OS platform features to scan a barcode, such as a UPC symbol or QR code. When a barcode is successfully scanned, the data that was read from the barcode is returned to the Lightning web component that invoked it.

Scanning is performed locally on the mobile device and doesn't need a network connection. BarcodeScanner does require access to platform-specific APIs that are available only within compatible Salesforce mobile apps.

 **Important:** BarcodeScanner does not and cannot function when running in a web browser, whether running on a desktop or mobile device.

BarcodeScanner provides to your component a string value of the data encoded in a scanned barcode. It doesn't attempt to interpret or process the decoded value.

 **Note:** This barcode scanner plugin is for use only in the Field Service mobile app.

Use the BarcodeScanner API

To develop a Lightning web component with barcode scanning features, use the BarcodeScanner API.

- Import BarcodeScanner into your component definition to make the BarcodeScanner API functions available to your code.
- Test to make sure BarcodeScanner is available before you call scanning lifecycle functions.
- Use the scanning lifecycle functions to start, continue, and stop scanning.

 **Note:** We recommend using the modern `scan()` and `dismiss()` APIs in your LWC scanning code to streamline your development experience. The legacy APIs `beginCapture()`, `resumeCapture()`, and `endCapture()` are still available, but will be retired in a future release.

Add BarcodeScanner to a Lightning Web Component

In your component's JavaScript file, import BarcodeScanner using the standard JavaScript `import` statement. Specifically, import the `getBarcodeScanner` factory function from the `lightning/mobileCapabilities` module, like so:

```
import { getBarcodeScanner } from "lightning/mobileCapabilities";
```

After it's imported into your component, use the factory function to get an instance of BarcodeScanner. With your BarcodeScanner instance, use the utility functions and constants to verify scanner availability and to configure scans. Use the scanning lifecycle functions to perform scanning operations.

Test BarcodeScanner Availability

BarcodeScanner depends on physical device hardware and platform features. A component that uses BarcodeScanner renders without errors on a desktop computer, but scanning functions fail. To avoid these errors, test if BarcodeScanner functionality is available before you use it.

```
import { LightningElement } from 'lwc';
import { getBarcodeScanner } from 'lightning/mobileCapabilities';
export default class ImplementBarcodeScanner extends LightningElement {
  const myScanner = getBarcodeScanner();

  barcodeResults = 'Nothing scanned yet!';

  handleBeginScanClick(event) {
```

```

    if(myScanner.isAvailable()) {
        // Perform scanning operations
        let scanningOptions = {
            "barcodeTypes": ["code128", "code39", "code93", "ean13", "ean8", "upca", "upce",
"qr", "datamatrix", "itf", "pdf417"], "instructionText": "Position barcode in the scanner
view.\nPress x to stop.",
"successText": "Successful Scan!"
        };
        myScanner.scan(scanningOptions)
            .then((results) => {
                // Do something with the results of the scan
                this.barcodeResults = '';
                results.forEach(result => {
                    this.barcodeResults += 'type: ' + result.type + ', value: ' + result.value +
'\n';
                });
            })
            .catch((error) => {
                // Handle cancellation and scanning errors here
                this.barcodeResults = 'Error code: ' + error.code + '\nError message: ' +
error.message;
            })
            .finally(() => {
                myScanner.dismiss();
            });
    } else {
        // Scanner not available
        // Not running on hardware with a scanner
        // Handle with message, error, beep, and so on
        this.barcodeResults = 'Problem initiating scanner. Are you using a mobile device?';
    }
}
}

```

Supported Barcode Types

- aztec
- code39
- code93
- code128
- dataMatrix
- ean8
- ean13
- interleaved2of5
- pdf417
- qr
- upce

Scan a Barcode

Scan with BarcodeScanner using the scanning lifecycle functions.

- Start a scan with `scan(options)`.
- Handle the result of the scan, which is returned in the form of a promise.
- End the scan with `dismiss()`.

For example:

```
myScanner
  .scan(scanningOptions)
  .then((result) => {
    // Do something with the result of the scan
    console.log(result);
    this.scannedBarcode = result.value;
  })
  .catch((error) => {
    // Handle cancellation and scanning errors here
    console.error(error);
  })
  .finally(() => {
    myScanner.dismiss();
  });
```

See [scan\(options\)](#) for more details of how to handle scan results, handle errors, and so on.

SEE ALSO:

[BarcodeScanner API](#)

Capture a Space on a Mobile Device

Create a Lightning web component (LWC) that uses the AR SpaceCapture plugin to create a 2D or 3D model of a room. The model is created by interactively scanning the physical environment using the device's Lidar capabilities.

-  **Note:** This AR SpaceCapture plug-in is supported only by iOS devices that have Lidar (iPhone 12 pro+ and iPad Pro and above), and that are running on iOS 16.0+ and iPadOS 16.0+.
-  **Note:** This AR SpaceCapture plugin is for use only in the Field Service mobile app.

IN THIS SECTION:

[Use the AR SpaceCapture API](#)

To develop an LWC with AR Space Capture features, use the AR SpaceCapture API as your method for accessing a device's native AR functionality.

[AR SpaceCapture User Experience](#)

Your component can deliver any user experience you desire, but you must follow a common flow for any component that calls for an AR SpaceCapture scan.

[AR SpaceCapture Example](#)

The component's HTML template is minimal and includes a button that initiates the room scan.

[AR SpaceCapture API](#)

Access a device's camera and AR capabilities from an LWC.

SEE ALSO:

[Apple's RoomPlan](#)

Use the AR SpaceCapture API

To develop an LWC with AR Space Capture features, use the AR SpaceCapture API as your method for accessing a device's native AR functionality.

1. Add AR SpaceCapture to an LWC

In your component's JavaScript file, import ARSpaceCapture using the standard JavaScript import statement. Specifically, import the `getARSpaceCapture()` factory function from the `lightning/mobileCapabilities` module.

```
import {getARSpaceCapture} from 'lightning/mobileCapabilities';
```

After it's imported into your component, use the factory function to get an instance of ARSpaceCapture. With your ARSpaceCapture instance, use the utility functions and constants to verify availability. Then use the feature functions to perform the associated functionality.

2. Test AR SpaceCapture Availability

AR SpaceCapture depends on physical device hardware and platform features. A component that uses SpaceCapture renders without errors on a desktop computer or in a mobile browser, but SpaceCapture functions fail. To avoid these errors, test if SpaceCapture functionality is available before you use it.

```
handleCheckARSpaceCaptureClick(event) {  
    const myARSpaceCapture = getARSpaceCapture();  
    if(myARSpaceCapture.isAvailable()) {  
        // Perform next operations  
    } else {  
        // AR SpaceCapture isn't available, or consuming app hasn't implemented it  
        // Not running on hardware with AR functionality, etc.  
        // Handle with message, error, beep, and so on  
    }  
}
```

3. Start an AR SpaceCapture Scan

Start a room scan with the `scanRoom()` function. You can then handle the JSON file that represents the USDZ file as required.

For example:

```
handleScanRoomClicked(event) {  
    if (this.myARSpaceCapture !== null && this.myARSpaceCapture.isAvailable()) {  
        this.myARSpaceCapture.scanRoom()  
            .then((arSpaceCaptureResult) => {  
                console.log(JSON.stringify(arSpaceCaptureResult, undefined, 2))  
            })  
            .catch((error) => {  
                console.log(error);  
            });  
    }  
};
```



```
}
}
```

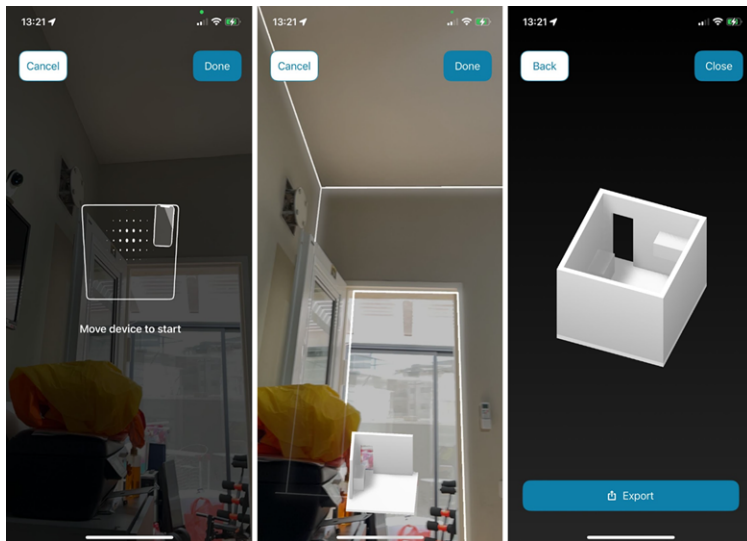
Considerations:

- The floor must be leveled when starting the scan.
- A scanned space includes only items that are on the same level.

AR SpaceCapture User Experience

Your component can deliver any user experience you desire, but you must follow a common flow for any component that calls for an AR SpaceCapture scan.

The user performs an action that triggers a space capture scan. The operating system provides messages for the user to aim the camera in the right direction and instructs the user to move the camera to the floors or ceilings. If the room scan isn't successful, the operating system provides an error message.



AR SpaceCapture Example

The component's HTML template is minimal and includes a button that initiates the room scan.

```
<template>
  <div style="height: 100%; padding: 0px;margin: 0px;">
    <table class="rootTable" style="width: 100%; height: 100%; padding: 0px; border-spacing:
5px">
      <thead>
        <tr>
          <th colspan="3">
            <h1>AR SpaceCapture Demo</h1>
          </th>
        </tr>
      </thead>
      <tbody>
        <tr style="height: 1px;">
          <td><input type="button" class="lightningButton" onclick={handleBeginScanRoomClick}
value='Scan Room' style="width: 100%; height: 50px; border: none; color: white; background:
```

```

    #0072d9; border-radius: 6px; font-size: medium; white-space: normal; word-wrap:
break-word;"/></td>
    </tr>
    <tr>
      <td colspan="3">
        <div lwc:ref="previewDivSummary" class="previewDivSummary" style="width: 100%;
height: 100%; border: 1px solid #c3c3c3; border-radius: 6px;">
          Summary: <br>
          <div lwc:ref="outputDivSummary" id="outputDivSummary" style="margin:
10px;">Results will be shown here soon...</div>
        </div>
      </td>
    </tr>
    <tr>
      <td colspan="3">
        <div lwc:ref="previewDiv" class="previewDiv" style="width: 100%; height: 100%;
border: 1px solid #c3c3c3; border-radius: 6px;">
          Full JSON: <br>
          <div lwc:ref="outputDiv" id="outputDiv" style="margin: 10px;">Results will
be shown here soon...</div>
        </div>
      </td>
    </tr>
  </tbody>
</table>
</div>
</template>

```

This example uses AR SpaceCapture to let the user scan a room.

```

import { LightningElement } from 'lwc';
import { ShowToastEvent } from 'lightning/platformShowToastEvent';
import { getARSpaceCapture } from 'lightning/mobileCapabilities';

export default class ARSpaceCaptureNew extends LightningElement {
  myARSpaceCaptureScanner;
  scanRoomDisabled = false;
  capturedRoomsData = '';

  // When the component is initialized, determine whether to enable the Scan Room button
  connectedCallback() {
    this.myARSpaceCaptureScanner = getARSpaceCapture();
    if (this.myARSpaceCaptureScanner?.isAvailable() !== true) {
      this.scanRoomDisabled = true;
    }
  }

  handleBeginScanRoomClick() {
    // Reset capturedRoomsData to empty string before starting a new scan
    this.capturedRoomsData = '';
  }
}

```

```

// Make sure AR SpaceCapture is available before trying to use it.
// Scan Room button also disabled when scanner unavailable
if (this.myARSpaceCaptureScanner?.isAvailable()) {
  let options = {};

  // Starting the scanning process
  this.myARSpaceCaptureScanner.scanRoom(options)
    .then((results) => {
      try {
        this.refs.outputDiv.innerHTML = "";
      } catch (ex) {

      }

      const capturedRooms = results.capturedRooms ?? [];
      if (capturedRooms) {
        // Array of Rooms
        let summary = {};

        try {
          summary.isSuccess = results.isSuccess;
          summary.wallsCount = results.capturedRooms[0].walls.length;
          summary.floorsCount = results.capturedRooms[0].floors.length;
          summary.openingsCount = results.capturedRooms[0].openings.length;
          summary.doorsCount = results.capturedRooms[0].doors.length;
          summary.windowsCount = results.capturedRooms[0].windows.length;

          if (results.capturedRooms[0].floors.length > 0) {
            summary.roomSizeWidth = results.capturedRooms[0].floors[0].dimensions[0];
            summary.roomSizeHeight = results.capturedRooms[0].floors[0].dimensions[1];
            summary.roomVolume = summary.roomSizeWidth * summary.roomSizeHeight;
          }

          if (results.capturedRooms[0].openings.length > 0) {
            summary.openingSizeWidth = results.capturedRooms[0].openings[0].dimensions[0];
            summary.openingSizeHeight = results.capturedRooms[0].openings[0].dimensions[1];
            summary.openingVolume = summary.openingSizeWidth * summary.openingSizeHeight;
          }

          if (results.capturedRooms[0].doors.length > 0) {
            summary.doorSizeWidth = results.capturedRooms[0].doors[0].dimensions[0];
            summary.doorSizeHeight = results.capturedRooms[0].doors[0].dimensions[1];
            summary.openingVolume = summary.doorSizeWidth * summary.doorSizeHeight;
          }

          this.refs.outputDivSummary.innerHTML = "<pre><code>" + JSON.stringify(summary,

```

```

undefined, 2) + "</code></pre>";
    this.refs.outputDiv.innerHTML = "<pre><code>" + JSON.stringify(results, undefined,
2) + "</code></pre>";
  } catch (ex) {
    this.refs.outputDivSummary.innerHTML = ex.code + "<br>" + ex.message;
  }
  } else {
// Single Room

    try {
      let summary = {};

      summary.isSuccess = results.isSuccess;
      summary.wallsCount = results.capturedRoom.walls.length;
      summary.floorsCount = results.capturedRoom.floors.length;
      summary.openingsCount = results.capturedRoom.openings.length;
      summary.doorsCount = results.capturedRoom.doors.length;
      summary.windowsCount = results.capturedRoom.windows.length;

      if (results.capturedRoom.floors.length > 0) {
        summary.roomSizeWidth = results.capturedRoom.floors[0].dimensions[0];
        summary.roomSizeHeight = results.capturedRoom.floors[0].dimensions[1];
        summary.roomVolume = summary.roomSizeWidth * summary.roomSizeHeight;
      }

      if (results.capturedRoom.openings.length > 0) {
        summary.openingSizeWidth = results.capturedRoom.openings[0].dimensions[0];
        summary.openingSizeHeight = results.capturedRoom.openings[0].dimensions[1];
        summary.openingVolume = summary.openingSizeWidth * summary.openingSizeHeight;
      }

      if (results.capturedRoom.doors.length > 0) {
        summary.doorSizeWidth = results.capturedRoom.doors[0].dimensions[0];
        summary.doorSizeHeight = results.capturedRoom.doors[0].dimensions[1];
        summary.doorVolume = summary.doorSizeWidth * summary.doorSizeHeight;
      }

      this.refs.outputDivSummary.innerHTML = "<pre><code>" + JSON.stringify(summary,
undefined, 2) + "</code></pre>";
      this.refs.outputDiv.innerHTML = "<pre><code>" + JSON.stringify(results, undefined,
2) + "</code></pre>";
    } catch (ex) {
      this.refs.outputDivSummary.innerHTML = ex.code + "<br>" + ex.message;
    }
  }
  })

```

```
.catch((error) => {  
  // There was an error while scanning  
  this.refs.outputDivSummary.innerHTML = error.code + "<br>" + error.message;  
})  
.finally({  
  // Close capture process regardless of whether we completed successfully or had an  
error  
  // this.myARSpaceCaptureScanner.dismiss();  
});  
} else {  
  this.refs.outputDivSummary.innerHTML = 'AR SpaceCapture is not available on your device.';  
}  
}  
}
```

AR SpaceCapture API

Access a device's camera and AR capabilities from an LWC.

```
function scanRoom(options) {  
  return new Promise((resolve, reject) => {  
    // Implement the logic to scan the room using the provided options  
    // and return the ARSpaceCaptureResult  
    // When the scan is complete, resolve the promise with the result  
    resolve({  
      // Populate the ARSpaceCaptureResult object with the appropriate data  
    });  
  });  
}
```

INDEX

C

Classes

- FSL.AdvancedGapMatrix [567](#)
- FSL.AppointmentBookingService [569](#)
- FSL.AppointmentBookingSlot [579](#)
- FSL.AppointmentInsightsResult [582](#)
- FSL.GradeSlotsService [580](#)
- FSL.OAAS [588](#)
- FSL.OAASRequest [598](#)
- FSL.PolygonUtils [601](#)
- FSL.RecurringAppointmentsManager [603](#), [608](#)
- FSL.RecurringPattern [606](#)
- FSL.ResourceScheduleData [609](#)
- FSL.ScheduleJobsApi [610](#)
- FSL.ScheduleJobsApi.GroupData [587](#)
- FSL.ScheduleResult [612](#)
- FSL.ScheduleService [615](#)
- FSL.SchedulingOption [619](#)
- FSL.TimeInterval [620](#)

F

- Field Service Flow API [498](#)
- FieldServiceMobileSettings [503](#)

FSL

- namespace [565](#)
- FSL.AdvancedGapMatrix
 - classes [567](#)
- FSL.AppointmentBookingService
 - classes [569](#)
- FSL.AppointmentBookingSlot
 - classes [579](#)
- FSL.AppointmentInsightsResult
 - classes [582](#)
- FSL.GradeSlotsService
 - classes [580](#)
- FSL.OAAS
 - classes [588](#)

- FSL.OAASRequest
 - classes [598](#)
- FSL.PolygonUtils
 - classes [601](#)
- FSL.RecurringAppointmentsManager
 - classes [603](#), [608](#)
- FSL.RecurringPattern
 - classes [606](#)
- FSL.ResourceScheduleData
 - classes [609](#)
- FSL.ScheduleJobsApi
 - classes [610](#)
- FSL.ScheduleJobsApi.GroupData
 - classes [587](#)
- FSL.ScheduleResult
 - classes [612](#)
- FSL.ScheduleService
 - classes [615](#)
- FSL.SchedulingOption
 - classes [619](#)
- FSL.TimeInterval
 - classes [620](#)

N

- Namespaces
 - FSL [565](#)

O

- Objects
 - TimeSheetTemplate [561](#)

S

- ServiceReportTemplate [506](#)

T

- TimeSheetTemplate object [561](#)