

---

# Deploy Enhancements from Sandboxes

Salesforce, Spring '18





# CONTENTS

<a href="#">Deploy Enhancements from Sandboxes</a> .....	1
Sandboxes: Staging Environments for Customizing and Testing .....	1
Create, Clone, or Refresh a Sandbox .....	5
Manage Your Sandboxes .....	11
Deploy Your Changes .....	20
<a href="#">Index</a> .....	49



# DEPLOY ENHANCEMENTS FROM SANDBOXES

*Sandboxes* create copies of your Salesforce org in separate environments. Use them for development, testing, and training, without compromising the data and applications in your production org.

## IN THIS SECTION:

### [Sandboxes: Staging Environments for Customizing and Testing](#)

Want to customize your organization in a staging environment where you can test changes without affecting your production organization or its users? Want to have an organization that users can log into and test new features before they're production-ready? Or maybe you just want to log into a Salesforce organization for training or development that mirrors your production organization.

### [Create, Clone, or Refresh a Sandbox](#)

Create a sandbox to use for development, testing, and training. Clone a sandbox to copy its data and metadata into another sandbox. Refresh an existing sandbox to update its contents.

### [Manage Your Sandboxes](#)

In Setup, enter *Sandboxes* in the **Quick Find** box, then select **Sandboxes**. Sandboxes displays the available sandboxes that you purchased and a list of your sandboxes in use.

### [Deploy Your Changes](#)

Migrate metadata changes between Salesforce orgs by using the deployment tools available in Setup.

## Sandboxes: Staging Environments for Customizing and Testing

---

Want to customize your organization in a staging environment where you can test changes without affecting your production organization or its users? Want to have an organization that users can log into and test new features before they're production-ready? Or maybe you just want to log into a Salesforce organization for training or development that mirrors your production organization.

Salesforce offers sandboxes and a set of deployment tools, so you can:

- Isolate customization and development work from your production environment until you're ready to deploy changes.
- Test changes against copies of your production data and users.
- Provide a training environment.
- Coordinate individual changes into one deployment to production.

Whether you're an administrator adding features to an organization, a solo developer writing code, or a team of developers working to enhance your organization, you should work with the right tools in the right environment to build and deploy change successfully to your production organization. For a broad overview of the development process and recommendations for organizing your work, see the [Development Lifecycle Guide](#).

## IN THIS SECTION:

### [Sandbox Types and Templates](#)

Sandboxes are isolated from your production org, so operations that you perform in your sandboxes don't affect your production org.

[Sandbox Licenses and Storage by Type](#)

You can create different sandbox environments for your org, depending on your needs for storage, copy configuration, and frequency of refresh.

## SEE ALSO:

[Sandbox Types and Templates](#)[Deploy Your Changes](#)[Choose Your Tools for Developing and Deploying Changes](#)

## Sandbox Types and Templates

Sandboxes are isolated from your production org, so operations that you perform in your sandboxes don't affect your production org.

From Setup, enter *Sandboxes* in the **Quick Find** box, then select **Sandboxes** to view and manage your existing sandboxes or create new ones.

### Sandbox Types

#### Developer Sandbox

A Developer sandbox is intended for development and testing in an isolated environment. A Developer Sandbox includes a copy of your production org's configuration (metadata).

#### Developer Pro Sandbox

A Developer Pro sandbox is intended for development and testing in an isolated environment and can host larger data sets than a Developer sandbox. A Developer Pro sandbox includes a copy of your production org's configuration (metadata). Use a Developer Pro sandbox to handle more development and quality assurance tasks and for integration testing or user training.

#### Partial Copy Sandbox

A Partial Copy sandbox is intended to be used as a testing environment. This environment includes a copy of your production org's configuration (metadata) and a sample of your production org's data as defined by a sandbox template. Use a Partial Copy sandbox for quality assurance tasks such as user acceptance testing, integration testing, and training.

#### Full Sandbox

A Full sandbox is intended to be used as a testing environment. Only Full sandboxes support performance testing, load testing, and staging. Full sandboxes are a replica of your production org, including all data, such as object records and attachments, and metadata. The length of the refresh interval makes it difficult to use Full sandboxes for development.

We recommend that you apply a sandbox template so that your sandbox contains only the records that you need for testing or other tasks.

When you create a Full sandbox, you also have to decide how much field tracking history and Chatter activity to include.

- The default is to omit field tracking, but you can include up to 180 days of field tracking. If you track field history for many objects in your production org, specify fewer days to avoid generating an excessive amount of data.
- Chatter activity data can be extensive, which can add a significant amount of time to your Full sandbox copy.

Limit the amount of field history that you copy, and copy your Chatter data only if you need it for your testing use cases.

#### EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **Professional, Enterprise, Performance, Unlimited,** and **Database.com** Editions

#### USER PERMISSIONS

To view a sandbox:

- View Setup and Configuration

To create, refresh, activate, and delete a sandbox:

- Manage Sandbox

## Sandbox Templates Tab

If you have purchased a license for Partial Copy or Full sandboxes, this tab lists any templates you have created.

Create a Sandbox Data Template, create a sandbox from a template, edit or delete a template, or click the template name for more information. For more information about creating a Sandbox Data Template, see [Create or Edit Sandbox Templates](#) on page 15.

## Sandbox History Tab

This tab displays a log of your sandbox creation and a history of refreshes, including when sandboxes were created and who created them. This tab provides information only. To view or edit an existing sandbox, use the Sandbox tab.

## Sandbox Licenses

You purchase licenses for each sandbox type and can purchase multiple licenses of each type. Sandbox licenses are hierarchical. Specifically, the following table shows the type of sandbox you can create with each license:

	Full Sandbox license	Partial Copy Sandbox license	Developer Pro Sandbox license	Developer Sandbox license
<b>Allows you to create:</b>				
Developer sandbox type	✓	✓	✓	✓
Developer Pro sandbox type	✓	✓	✓	
Partial Copy sandbox type	✓	✓		
Full sandbox type	✓			

License stages are:

### Available

The displayed value represents the number of sandboxes that you've purchased but haven't used.

### In use

The displayed value represents the number of sandboxes that you've purchased and used.

 **Note:** If you don't see a sandbox option or need licenses for more sandboxes, contact Salesforce to order sandboxes for your org.

When your sandbox licenses expire, your existing sandboxes are subject to certain restrictions. See [Unlock a Sandbox](#) on page 13 for resolution of license expiration issues.

## Understanding Sandbox Refresh Intervals

The refresh interval for each sandbox environment is calculated from when the actual sandbox copying process begins. The sandbox status changes from Pending to Processing at the start of copying.

If other sandbox copy requests were made before yours, your sandbox sometimes remains in the Pending status for some time. The refresh interval timer for your sandbox doesn't start until your request leaves this state.

SEE ALSO:

[Create a Sandbox](#)

[Sandbox Licenses and Storage by Type](#)

[Unlock a Sandbox](#)

[General Sandbox and Sandbox Storage Limits](#)

[Development Lifecycle Guide](#)

## Sandbox Licenses and Storage by Type

You can create different sandbox environments for your org, depending on your needs for storage, copy configuration, and frequency of refresh.

Each type has different features to support the activities it's designed for.

**Table 1: Sandboxes Available Per Edition**

Sandbox Type	Professional Edition	Performance Edition	Unlimited Edition	Enterprise Edition
Developer Sandbox	10 (change sets not available)	100	100	25
Developer Pro Sandbox		5	5	
Partial Copy Sandbox		1	1	1
Full Sandbox		1	1	

### EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **Professional, Enterprise, Performance, Unlimited,** and **Database.com** Editions

 **Note:** You can buy more sandboxes for any edition except Developer Sandbox, which is bundled with add-on sandboxes of other types.

- The Developer Pro Sandbox add-on is bundled with 5 Developer Sandboxes.
- The Partial Copy Sandbox add-on is bundled with 10 Developer Sandboxes.
- The Full Sandbox add-on is bundled with 15 Developer Sandboxes.

**Table 2: Sandbox Feature Quick Reference**

Sandbox Type	Refresh Interval	Storage Limit	What's Copied	Sandbox Templates
Developer Sandbox	1 day	Data storage: 200 MB File storage: 200 MB	Metadata only	Not available
Developer Pro Sandbox	1 day	Data storage: 1 GB File storage: 1 GB	Metadata only	Not available

Sandbox Type	Refresh Interval	Storage Limit	What's Copied	Sandbox Templates
Partial Copy Sandbox	5 days	Data storage: 5 GB File storage: 5 GB	Metadata and sample data	Required
Full Sandbox	29 days	Same as your production org	Metadata and all data	Available

## SEE ALSO:

- [Create a Sandbox](#)
- [Create or Edit Sandbox Templates](#)
- [Sandbox Setup Considerations](#)

## Create, Clone, or Refresh a Sandbox

Create a sandbox to use for development, testing, and training. Clone a sandbox to copy its data and metadata into another sandbox. Refresh an existing sandbox to update its contents.

You have a few ways to copy metadata and data to a sandbox.

- **Create a sandbox.**  
When you create a sandbox, Salesforce copies the metadata from your production org to a sandbox org. While creating a Partial Copy or Full sandbox, you can apply a sandbox template, if you have created one. Customers create sandbox templates to define specific object data to copy into the Partial Copy or Full sandbox.
- **Clone an existing sandbox to copy all its data and metadata to a new sandbox.**
- **Refresh and activate an existing sandbox.**  
Refreshing a sandbox updates the sandbox's metadata from its source org. If the sandbox is a clone or if it uses a sandbox template, the refresh process updates the org's data in addition to its metadata.

## IN THIS SECTION:

### [Create a Sandbox](#)

When you create a sandbox, Salesforce copies the metadata from your production org to a sandbox org.

### [Refresh Your Sandbox](#)

Refreshing a sandbox updates the sandbox's metadata from its source org. If the sandbox is a clone or if it uses a sandbox template, the refresh process updates the org's data in addition to its metadata.

### [Activate Your Refreshed Sandbox](#)

If you didn't select **Auto Activate** while refreshing your sandbox, you must activate your sandbox before you can use it.

### EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **Professional, Enterprise, Performance, Unlimited,** and **Database.com** Editions

### USER PERMISSIONS

To view a sandbox:

- View Setup and Configuration

To create, refresh, activate, and delete a sandbox:

- Manage Sandbox

### Sandbox Cloning (Beta)

You can create a sandbox by cloning an existing sandbox rather than using your production org as your source. Save time by customizing a sandbox with a set of data and metadata and then replicating it. Sandbox cloning simplifies having multiple concurrent streams of work in your application life cycle. You can set up a sandbox for each type of work, such as development, testing, and staging. Your colleagues can easily clone individual sandboxes instead of sharing one sandbox and avoid stepping on each other's toes.

### Monitor Your Sandbox's Progress

From Setup, enter *Sandboxes* in the **Quick Find** box, then select **Sandboxes**. The list of your sandboxes displays a progress bar for items in the queue, in progress, or recently completed.

## Create a Sandbox

When you create a sandbox, Salesforce copies the metadata from your production org to a sandbox org.

1. From Setup, enter *Sandboxes* in the **Quick Find** box, then select **Sandboxes**.
2. Click **New Sandbox**.
3. Enter a name (10 characters or fewer) and description for the sandbox.

We recommend that you choose a name that:

- Reflects the purpose of this sandbox, such as QA.
- Has only a few characters, because Salesforce appends the sandbox name to usernames on user records in the sandbox environment. Names with fewer characters make sandbox logins easier to type.

4. Select the type of sandbox you want.

If you don't see a sandbox option or need licenses for more, contact Salesforce to order sandboxes for your org.

If you reduce the number of sandboxes you purchase, you are required to match the number of your sandboxes to the number you purchased. For example, if you have two Full sandboxes but purchased only one, you can't create a Full sandbox. Instead, convert a Full sandbox to a smaller one, such as a Developer Pro or Developer sandbox, depending on which types you have available.

5. Select the data to include in your Partial Copy or Full sandbox.
  - For a Partial Copy sandbox, click **Next**, and then select the template you created to specify the data for your sandbox. If you have not created a template for this Partial Copy sandbox, see [Create or Edit Sandbox Templates](#).
  - For a Full sandbox click **Next**, and then decide how much data to include.
    - To include template-based data for a Full sandbox, select an existing sandbox template. For more information, see [Create or Edit Sandbox Templates](#)
    - To include all data in a Full sandbox, choose whether and how much field tracking history data to include, and whether to copy Chatter data. You can copy from 0 to 180 days of history, in 30-day increments. The default is 0 days. Chatter data includes feeds, messages, and discovery topics. Decreasing the amount of data you copy can significantly speed sandbox copy time.
6. To run scripts after each create and refresh for this sandbox, specify the Apex class you previously created from the SandboxPostCopy interface.

### EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **Professional, Enterprise, Performance, Unlimited,** and **Database.com** Editions

### USER PERMISSIONS

To view a sandbox:

- View Setup and Configuration

To create, refresh, activate, and delete a sandbox:

- Manage Sandbox

## 7. Click **Create**.

 **Tip:** Try to limit changes in your production org while the sandbox copy proceeds.

The process takes from several minutes to several days, depending on the size and type of your org.

When your sandbox is ready for use, you receive a notification email that your sandbox has completed copying.

To access your sandbox, click the link in the notification email. Users can log in to the sandbox at `https://test.salesforce.com` by appending `.sandbox_name` to their Salesforce usernames. For example, if a username for a production org is `user1@acme.com`, and the sandbox is named "test," the modified username to log in to the sandbox is `user1@acme.com.test`.

 **Note:** Salesforce automatically changes sandbox usernames, but not passwords. New sandboxes have the default email deliverability setting `System email only`. The `System email only` setting is especially useful for controlling email sent from sandboxes so that testing and development work doesn't send test emails to your users.

### SEE ALSO:

[Sandbox Types and Templates](#)

[Sandbox Licenses and Storage by Type](#)

[Create or Edit Sandbox Templates](#)

[Sandbox Setup Considerations](#)

[Unlock a Sandbox](#)

## Refresh Your Sandbox

Refreshing a sandbox updates the sandbox's metadata from its source org. If the sandbox is a clone or if it uses a sandbox template, the refresh process updates the org's data in addition to its metadata.

### 1. From Setup, enter *Sandboxes* in the `Quick Find` box, then select **Sandboxes**.

A list of your sandboxes displays. Sandboxes that you can refresh have a Refresh link next to their name.

### 2. Next to the name, click **Refresh**.

### 3. Review the Name, Description, and Create From values, and edit these values if needed.

### 4. Select the type of sandbox environment you want.

A table shows the number and type of sandbox licenses available in your org. You can select a different sandbox type to refresh.

If the sandbox you're refreshing is a clone, this option isn't available. A cloned sandbox refreshes from its source org and retains the source org's sandbox license type. If a sandbox's source org has been deleted, the clone refreshes from your production org.

### 5. Select the data you want to copy.

For a Partial Copy sandbox, click **Next**, and then select a template to specify the data for your sandbox. If you have not created a template for this Partial Copy sandbox, see [Create or Edit Sandbox Templates](#).

For a Full sandbox, click **Next**, and then decide how much object data to include.

To include template-based data in a Full sandbox, select an existing sandbox template. For more information, see [Create or Edit Sandbox Templates](#).

### EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **Professional, Enterprise, Performance, Unlimited,** and **Database.com** Editions

### USER PERMISSIONS

To view a sandbox:

- View Setup and Configuration

To create, refresh, activate, and delete a sandbox:

- Manage Sandbox

To include all object data in a Full sandbox, choose whether and how much field tracking history to include, and whether to copy Chatter data. You can copy from 0 to 180 days of history, in 30-day increments. The default is 0 days. Chatter data includes feeds, messages, and discovery topics. Decreasing the amount of data you copy can speed sandbox copy time.

6. If you want to activate your sandbox immediately after you refresh it, select **Auto Activate**. In this case, you don't receive an activation email.
7. Click **Create**.
8. Refreshing your sandbox can move it to a different Salesforce instance. For example, the sandbox can move from CS40 to CS50. If you're subscribed to [Trust Notifications](#), check your subscription settings to ensure that you continue to receive updates about unforeseen incidents and planned maintenance that affect your sandbox.

Salesforce starts copying data to the sandbox.

If you didn't select **Auto Activate** while refreshing your sandbox, Salesforce sends you an email when your sandbox is ready to activate.

## Activate Your Refreshed Sandbox

If you didn't select **Auto Activate** while refreshing your sandbox, you must activate your sandbox before you can use it.

1. From Setup, enter *Sandboxes* in the **Quick Find** box, then select **Sandboxes**.  
A list of your sandboxes displays. Refreshed sandboxes that haven't been activated yet show an **Activate** link next to their name.

2. Click **Activate** next to the sandbox you want to activate.



**Warning:** Activating a replacement sandbox that was created using the Refresh link deletes the sandbox it is refreshing. The current configuration and data are erased, including application or data changes that you've made. Click the **Activate** link only if you don't need the current contents of the sandbox. Your production org and its data aren't affected.

Salesforce deletes new sandboxes that aren't activated within 30 days. Users who created or most recently refreshed any sandboxes for your org receive at least two email notifications before Salesforce schedules the sandbox for deletion.

## Sandbox Cloning (Beta)

You can create a sandbox by cloning an existing sandbox rather than using your production org as your source. Save time by customizing a sandbox with a set of data and metadata and then replicating it. Sandbox cloning simplifies having multiple concurrent streams of work in your application life cycle. You can set up a sandbox for each type of work, such as development, testing, and staging. Your colleagues can easily clone individual sandboxes instead of sharing one sandbox and avoid stepping on each other's toes.

-  **Note:** As a beta feature, Sandbox Cloning is a preview and isn't part of the "Services" under your master subscription agreement with Salesforce. Use this feature at your sole discretion, and make your purchase decisions only on the basis of generally available products and features. Salesforce doesn't guarantee general availability of this feature within any particular time frame or at all, and we can discontinue it at any time. This feature is for evaluation purposes only, not for production use. It's offered as is and isn't supported, and Salesforce has no liability for any harm or damage arising out of or in connection with it. All restrictions, Salesforce reservation of rights, obligations concerning the Services, and terms for related Non-Salesforce Applications and Content apply equally to your use of this feature. You can provide feedback and suggestions for Sandbox Cloning in the [IdeaExchange](#).

### EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **Professional, Enterprise, Performance, Unlimited, and Database.com** Editions

### USER PERMISSIONS

To view a sandbox:

- View Setup and Configuration

To create, refresh, activate, and delete a sandbox:

- Manage Sandbox

## IN THIS SECTION:

[Clone a Sandbox \(Beta\)](#)

When you clone a sandbox, all its data and metadata are copied to the new sandbox. A cloned sandbox uses the same license type as its source org.

[Refresh a Cloned Sandbox \(Beta\)](#)

Refreshing a cloned sandbox updates the sandbox's metadata and data from its source org.

## Clone a Sandbox (Beta)

When you clone a sandbox, all its data and metadata are copied to the new sandbox. A cloned sandbox uses the same license type as its source org.

 **Note:** You can only clone sandboxes that are on the same major Salesforce version as the production org. During the transition to a major release, some sandboxes get upgraded before others. To determine the Salesforce version for your production and sandbox orgs, see [this knowledge article](#). To determine when your production org or sandbox gets upgraded, see the [Salesforce maintenance schedule](#).

1. From Setup, enter *Sandboxes* in the Quick Find box, then select **Sandboxes**.
2. Click **New Sandbox**, or click **Clone** next to a completed sandbox.

 **Note:** During the beta, some older sandboxes aren't available for cloning. Unavailable source orgs don't have a Clone link next to them.

3. Enter a name (10 characters or fewer) and description for the sandbox.

We recommend that you choose a name that:

- Reflects the purpose of this sandbox, such as QA.
- Has only a few characters, because Salesforce appends the sandbox name to usernames on user records in the sandbox environment. Names with fewer characters make sandbox logins easier to type.

4. If you clicked New Sandbox, from the Create From dropdown, select the name of the sandbox that you want to clone.

 **Note:** During the beta, some older sandboxes aren't available for cloning. Unavailable source orgs don't appear in the Create From dropdown.

5. If you clicked Clone, confirm that the sandbox name selected in the Create From dropdown is the sandbox you want to use as your source org.
6. Make sure that the org you're cloning has the license type that you want for your new sandbox. To use a different license type, choose a different source org from the Create From dropdown.
7. Click **Next**.
8. To run scripts after each creation and refresh for this sandbox, specify an Apex class that extends the `SandboxPostCopy` interface. The Apex class you specify must exist in your source org.
9. Click **Create**.

 **Tip:** Avoid making changes in your source org while the sandbox copy occurs.

When your new sandbox is ready, you can manage it from your production org like any other sandbox.

### EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **Professional, Enterprise, Performance, Unlimited,** and **Database.com** Editions

### USER PERMISSIONS

To view a sandbox:

- View Setup and Configuration

To create, refresh, activate, and delete a sandbox:

- Manage Sandbox

## Refresh a Cloned Sandbox (Beta)

Refreshing a cloned sandbox updates the sandbox's metadata and data from its source org.

 **Note:** When refreshing a sandbox, you can't choose a source sandbox that's on a different major release version than the production org. During the transition to a major release, some sandboxes get upgraded before others. To determine the Salesforce version for your production and sandbox orgs, see [this knowledge article](#). To determine when your production org or sandbox gets upgraded, see the [Salesforce maintenance schedule](#).

1. From Setup, enter *Sandboxes* in the Quick Find box, then select **Sandboxes**.

A list of your sandboxes displays. Sandboxes that you can refresh have a Refresh link next to their name.

2. Next to the name of the sandbox you want to refresh, click **Refresh**.

3. Review the name and description, and edit them if needed.

4. Review the Create From value, which is the source org for the refresh. If you don't want to refresh the cloned sandbox using its original source org, select a different sandbox or your production org.

A cloned sandbox refreshes from its source org and retains the source org's sandbox license type. If a sandbox's source org has been deleted, the clone refreshes from your production org.

5. If you want to activate your sandbox immediately after you refresh it, select **Auto Activate**. In this case, you don't receive an activation email.

6. To run scripts after each creation and refresh for this sandbox, specify an Apex class that extends the `SandboxPostCopy` interface. The Apex class you specify must exist in your source org.

7. Click **Create**.

8. Refreshing your sandbox can move it to a different Salesforce instance. For example, the sandbox can move from CS40 to CS50. If you're subscribed to [Trust Notifications](#), check your subscription settings to ensure that you continue to receive updates about unforeseen incidents and planned maintenance that affect your sandbox.

Salesforce starts copying metadata and data to the sandbox.

If you didn't select **Auto Activate**, Salesforce emails you when your sandbox is ready to activate.

SEE ALSO:

[Activate Your Refreshed Sandbox](#)

### EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **Professional, Enterprise, Performance, Unlimited,** and **Database.com** Editions

### USER PERMISSIONS

To view a sandbox:

- View Setup and Configuration

To create, refresh, activate, and delete a sandbox:

- Manage Sandbox

## Monitor Your Sandbox's Progress

From Setup, enter *Sandboxes* in the **Quick Find** box, then select **Sandboxes**. The list of your sandboxes displays a progress bar for items in the queue, in progress, or recently completed.

- To show the percentage completed of a copy in progress, hover over the progress bar.
- To see information about the sandbox, including copy progress or how much time before the next available refresh, click the name.

If your sandbox status is suspended or stopped for more than 1 hour, contact Salesforce customer support.

### EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **Professional, Enterprise, Performance, Unlimited,** and **Database.com** Editions

### USER PERMISSIONS

To view a sandbox:

- View Setup and Configuration

To create, refresh, activate, and delete a sandbox:

- Manage Sandbox

## Manage Your Sandboxes

In Setup, enter *Sandboxes* in the **Quick Find** box, then select **Sandboxes**. Sandboxes displays the available sandboxes that you purchased and a list of your sandboxes in use.

### IN THIS SECTION:

#### [Sandbox Action and Status Reference](#)

Access the list of your sandboxes in your org from the Sandbox tab. Each entry shows the status of the sandbox environment and the specific actions that you can take.

#### [Unlock a Sandbox](#)

Sandboxes are licensed separately from the Salesforce service and are subject to restrictions. When your sandbox licenses expire, Salesforce decreases the count of available sandbox licenses for the selected sandbox type.

#### [Create or Edit Sandbox Templates](#)

Sandbox templates control which data is copied into a sandbox.

#### [Sandbox Setup Considerations](#)

Sandboxes behave almost the same as your Salesforce production org does, but important differences affect how you configure and test a sandbox org.

### SEE ALSO:

[Development Lifecycle Guide](#)

### EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **Professional, Enterprise, Performance, Unlimited,** and **Database.com** Editions

### USER PERMISSIONS

To view a sandbox:

- View Setup and Configuration

To create, refresh, activate, and delete a sandbox:

- Manage Sandbox

## Sandbox Action and Status Reference

Access the list of your sandboxes in your org from the Sandbox tab. Each entry shows the status of the sandbox environment and the specific actions that you can take.

### Available Sandbox Actions

Click a sandbox name to see the sandbox detail page. On the sandbox detail page, you can perform all the preceding actions: Log In, Refresh, Activate, Discard, and Delete.

Action	Description
Log In	Click <b>Log In</b> to log into the sandbox as an administrator. Salesforce displays this option only for active sandboxes. You must be logged in to your org as an administrator to see the Log In button.
Refresh	Click <b>Refresh</b> to replace a with a new copy. Salesforce activates the <b>Refresh</b> button only for sandboxes that are eligible for refreshing. Your existing copy of the sandbox remains available while you wait for the refresh to be completed. The refreshed copy is inactive until you activate it.
Activate	Click <b>Activate</b> to activate a refreshed sandbox. Salesforce displays this option only for sandbox that aren't activated. You must activate your refreshed sandbox before you can access it.   <b>Warning:</b> Activating a refreshed sandbox replaces the current sandbox with the refreshed version and permanently deletes the older version and all data in it. Your production org and its data aren't affected.
Discard	Click <b>Discard</b> to discard a refreshed sandbox. This action is displayed only for sandboxes that aren't activated. You can discard a sandbox if you don't want to use it or if you plan to do another refresh.
Del	Click <b>Del</b> to remove the sandbox entirely.   <b>Warning:</b> Deleting a sandbox permanently erases the sandbox and all data in it, including any outbound change sets that have been uploaded from the sandbox. Your production org and its data aren't affected.

#### EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **Professional, Enterprise, Performance, Unlimited,** and **Database.com** Editions

#### USER PERMISSIONS

To view a sandbox:

- View Setup and Configuration

To create, refresh, activate, and delete a sandbox:

- Manage Sandbox

 **Note:** The Delete and Refresh operations are available only when the refresh interval for that sandbox type has elapsed.

## Sandbox Statuses

Status	Description
Sampling	The copy engine is determining which object records are sampled and copied from the production org. This status is used only by Partial Copy sandboxes.
Pending	The sandbox is in the queue to be processed by the copy engine. If other sandbox copy requests were made before yours, your sandbox could remain in this state for an extended period of time.
Processing	The copy engine has picked up the copy request and is working to build the sandbox.
Suspended	The copy engine was interrupted while refreshing or creating the sandbox. The copy engine automatically recovers from this state and returns to processing. If this status remains unchanged for more than one hour, contact Salesforce Customer Support.
Stopped	The copy engine couldn't recover from multiple events. If your sandbox is in this state, contact Salesforce Customer Support for specific details and next steps. Salesforce is notified automatically of sandboxes in this state so we can resolve the issues.
Pending Activation	The copy engine created the sandbox and is waiting for you to activate or discard it.
Activating	The copy engine is completing the final steps to make your new sandbox available.
Discarding	The copy engine is marking the refreshed sandbox for deletion, because you clicked <b>Discard</b> . The current sandbox and your production org aren't affected by this process.
Completed	The copy engine has completed the creating or refreshing the sandbox, and the new sandbox is active. You can log in to your new sandbox org.
Deleting	The copy engine is marking the sandbox environment and all sandbox history for deletion. This status is used after you click <b>Delete</b> . This process doesn't affect your production org.
Locking	A background process is locking the sandbox. See the <b>Locked</b> status for more details.
Locked	You can't log into this sandbox because you let some or all your sandbox licenses expire. Contact your account manager to restore the expired licenses. You have 60 days to restore the licenses. If the licenses aren't restored within 60 days, your sandbox is deleted.

## Unlock a Sandbox

Sandboxes are licensed separately from the Salesforce service and are subject to restrictions. When your sandbox licenses expire, Salesforce decreases the count of available sandbox licenses for the selected sandbox type.

If your current license count is lower than the number of your provisioned sandbox orgs, Salesforce removes sandbox services, such as Refresh, Sandbox org accessibility, or Login.

 **Note:** Salesforce doesn't automatically delete sandbox orgs because of a license expiration.

### EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **Professional, Enterprise, Performance, Unlimited,** and **Database.com** Editions

Sandboxes are locked when all the licenses for a type of sandbox expire. Salesforce deletes sandboxes that have been locked for 60 days. Users who created or most recently refreshed any sandbox for your org are notified prior to scheduling the sandbox for deletion. They get at least three email notifications over 30 days.

Salesforce deletes new sandboxes that weren't activated within 30 days. Users who created or most recently refreshed any sandbox for your org get at least two email notifications before we schedule the unactivated sandbox for deletion.

Here are some scenarios and solutions, based on licensing and usage.

Scenario	Cause	Example	Effect	Resolution
Unable to refresh a particular type of sandbox	Your org is using more sandboxes than its sandbox licenses permit.	Your org has three Partial sandboxes but only two Partial sandbox licenses.	You can't refresh any sandbox. When you're over your limit on any type of sandbox, your org isn't allowed to refresh any sandboxes.	Delete sandboxes to comply with the number allowed by your org's sandbox licenses, or purchase more sandbox licenses.
All sandboxes of a particular type are locked	The license count of a given type, including higher hierarchical types, is zero.	Your org has three full sandboxes and zero full sandbox licenses.	You don't have access to the sandboxes.	Purchase the correct sandbox licenses to unlock the sandboxes.
All sandboxes are locked	Your production org is locked.	Your org has one full sandbox and one developer-sandbox, but you can't log in to either sandbox.	If your production org is locked, all sandboxes associated with the org are locked.	Contact your Salesforce representative to unlock your org. When your production org is unlocked, the sandboxes are unlocked as well.

**SEE ALSO:**

[General Sandbox and Sandbox Storage Limits](#)

[Create a Sandbox](#)

[Sandbox Licenses and Storage by Type](#)

[Manage Your Sandboxes](#)

[Sandbox Setup Considerations](#)

## Create or Edit Sandbox Templates

Sandbox templates control which data is copied into a sandbox.

Sandbox templates allow you to pick specific objects and data to copy to your Full or Partial Copy sandbox to control the size and content of each sandbox. Sandbox templates are only available for use with Full or Partial Copy sandboxes.

When you create a sandbox template, you select the object data (standard and custom) to copy during the creation or refresh of a sandbox.

The sandbox template editor understands the relationships that are defined in your Salesforce org's object schema. Some objects are included even before you've selected anything because they're required in any org. As you select objects to copy, the editor ensures that the associated required objects are added. To see which related objects are required by an object, select it in the **Object** table. Required objects are displayed in the **Required Objects** column.

As you change the schema of the objects in your org, Salesforce updates the template by adding or subtracting related required objects. For example, if Object A is a master of Object B, and you add Object B to a template, Salesforce requires Object A in the template and adds Object A.

1. From Setup, enter *Sandboxes* in the **Quick Find** box, select **Sandboxes**, then click the **Sandbox Templates** tab.
2. Click **New Sandbox Template** or click **Edit** next to an existing template you want to modify.
3. Enter a name and description for the sandbox template.
4. To add objects to the template, select the checkbox for each object you want from the available Objects list. The Object Details section shows you the objects to be added automatically with the one you've selected.
5. To remove objects from the template, deselect the checkbox for the object in the available Objects list. If you remove an object you previously selected, dependent objects you didn't explicitly select are removed. If you attempt to remove an object with dependent objects, you receive a warning requesting a confirmation of the removal. After you confirm your choice, those objects are also removed.
6. Click **Save**.

To understand how to use a Sandbox template during sandbox creation or refresh, see [Create a Sandbox](#) on page 6.

To understand how a Sandbox template is used by the sandbox copy engine to create a Full or Partial Copy sandbox, see [Sandbox Licenses and Storage by Type](#) on page 4.

 **Warning:** If you modify your object schema, your sandbox templates could be altered to include objects that are required by relationships. If you make a change to a required relationship in your object schema, review your sandbox templates to ensure that objects that you expect to be selected are still selected.

### SEE ALSO:

[Create a Sandbox](#)

[Sandbox Licenses and Storage by Type](#)

[Sandbox Types and Templates](#)

### EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **Professional, Enterprise, Performance, Unlimited,** and **Database.com** Editions

### USER PERMISSIONS

To view a sandbox:

- View Setup and Configuration

To create, refresh, activate, and delete a sandbox:

- Manage Sandbox

## Sandbox Setup Considerations

Sandboxes behave almost the same as your Salesforce production org does, but important differences affect how you configure and test a sandbox org.

Consider the following before you create a sandbox.

### Servers and IDs

- Sandbox and production orgs always have unique org IDs. The sandbox copy engine creates an org as part of each creation and refresh request. So, the org ID of your sandbox changes each time your sandbox is refreshed. Salesforce inserts the new value in any place the org ID is used, such as text values and metadata.

To find the org ID of the org that you're currently logged in to, from Setup, enter *Company Information* in the *Quick Find* box, then select **Company Information**. Any script or process, such as test scripts or Web-to-Lead, that depends on a "hard coded" org ID must use the current ID for the sandbox. When you deploy your changes to a production org, update those scripts or processes with the production org ID.

- Salesforce creates sandbox orgs on several instances. When a sandbox is created or refreshed, Salesforce selects an instance for your sandbox, so your sandboxes sometimes appear on different instances and have different URLs.
- When data is copied to a sandbox, object IDs (unique identifiers for all objects—the same as the [ID Field Type](#) in the developer API) for records are copied. After being copied, however, object IDs don't synchronize between your production org and your sandbox. The sandbox and its corresponding production org act as independent orgs. Object data (and corresponding object IDs) that are created in the production org after the creation or refresh of a sandbox aren't synchronized into the sandbox. The sandbox has the same behavior—new objects that are created in the sandbox aren't synchronized back to the production org.

### Users and Contacts

- User information is included in a sandbox copy or refresh for all sandbox types. Because all Salesforce usernames must be unique and reference a single org, all copied usernames are modified to ensure uniqueness during the copy process.

For each username, the copy process applies modifications as necessary to generate a unique, new username.

- First, the sandbox name is appended to the username. For example, the username `user@acme.com` for a sandbox named `test` becomes `user@acme.com.test`.
- If the resulting username is not unique, a second modification is performed in which some characters and digits are prepended to the modified username. This second modification results in a username such as `00x7Vquser@acme.com.test`.

When you log in with the modified username, you log in to the corresponding sandbox.

- The copy process doesn't copy Contact data to Developer or Developer Pro sandboxes. Therefore, Customer Portal users aren't copied. However, the copy process does copy the Customer Portal licenses, so you can create Customer Portal users in these sandboxes as needed.
- When you create or refresh a sandbox, user email addresses are modified in your sandbox so that production users don't receive automatically generated email messages from the sandbox. User email addresses are appended with an example domain (`@example.com`), and the original `@` sign is replaced with `=`. This modification ensures that the system ignores these email addresses. For example, a user email of `awheeler@universalcontainers.com` in production becomes

#### EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **Professional, Enterprise, Performance, Unlimited,** and **Database.com** Editions

#### USER PERMISSIONS

To view a sandbox:

- View Setup and Configuration

To create, refresh, activate, and delete a sandbox:

- Manage Sandbox

awheeler=universalcontainers.com@example.com when migrated to sandbox. If you want sandbox users to receive automatically generated emails as part of testing, you can correct the email addresses while logged in to the sandbox.

 **Warning:** Sandboxes change Salesforce user email addresses, but don't change other email addresses in Salesforce, such as email addresses in contact records. To avoid sending unsolicited email from your sandboxes, manually invalidate or delete all email addresses in your sandboxes that don't belong to users of the sandbox. When testing outbound email, change contact email addresses to the addresses of testers or an automated test script.

## Email Deliverability

New and refreshed sandboxes have the default email deliverability setting `System_email_only`. To configure email deliverability settings, in the sandbox org, from Setup, enter `Deliverability` in the `Quick Find` box, then select **Deliverability**. If editable, set the `Access Level` in the Access to Send Email section. You can't edit the Access level if Salesforce has restricted your organization's ability to change this setting.

- `No_access`: Prevents all outbound email to and from users.
- `System_email_only`: Allows only automatically generated emails, such as new user and password reset emails.
- `All_email`: Allows all types of outbound email. Default for new, non-sandbox organizations.

 **Tip:** The `System_email_only` setting is especially useful for controlling email sent from sandboxes so that testing and development work doesn't send test emails to your users.

- Newly created sandboxes default to `System_email_only`.
- Sandboxes created before Spring '13 default to `All_email`.

## Creating, Refreshing, and Deleting Sandboxes

- You can specify a post-copy script to run on a sandbox every time it is refreshed (and the first time it is created). Specify the script when you create the sandbox.
- You can copy Site.com and Site.com community sites to sandboxes.
- Sandbox copy is a long-running operation that occurs in the background. You are notified of the completion of a sandbox copy via email. Sandbox refreshes can complete in hours, days, or even more than a week.
- Several conditions affect the duration of a sandbox copy or refresh, including the number of customizations, data size, numbers of objects and configuration choices, and server load. Also, sandbox refreshes are queued, so your copy might not start immediately after your request.
- A sandbox is not a point-in-time snapshot of the exact state of your data. Furthermore, we recommend that you limit changes to your production org while a sandbox is being created or refreshed. Setup and data changes to your production org during the sandbox creation and refresh operations can result in inconsistencies in your sandbox. You might detect and correct some inconsistencies in your sandbox after it is created or refreshed.
- Creating or refreshing a sandbox occurs over time. Running a large process or updating orgs of 30 GB or more during creation or refresh can cause inconsistencies in your sandbox.
- Some types of sandboxes are not available if you've reached your org's limit. For example, if your org is limited to one full sandbox, and you already have a full sandbox, you can't create another full sandbox. However, you can refresh your existing full sandbox.
- When you're finished with a sandbox, you can refresh it. This process replaces the current sandbox with a new copy of your production org.
- If you have reduced your org's number of sandbox licenses, a **Delete** link displays next to existing sandboxes. Delete a sandbox before creating or refreshing any more sandboxes.

- If you have active Salesforce-to-Salesforce connections in your sandbox, deactivate the connections and then reactivate them after the sandbox is refreshed. The connections and mappings aren't copied to the refreshed sandbox.

## Matching Production Licenses

You can match provisioned licenses in production to your sandbox org without having to refresh your sandbox. Matching updates sandbox license counts to match the counts in production, adds licenses that are in production but not in sandbox, and deletes licenses that aren't in production.

To match licenses, log in to your sandbox. From Setup, enter *Company Information* in the Quick Find box, select **Company Information**, then click **Match Production Licenses**. You get an alert when the matching process starts. After all licenses have been matched in sandbox, you get a confirmation email. The confirmation email is sent to the user who initiates the license copy.

Matching production licenses requires that your sandbox and production are on the same Salesforce release. If your sandbox has been upgraded to the next release—for example, during sandbox preview—but production hasn't been upgraded, you can't match production licenses.

The Match Production Licenses tool isn't available in trial orgs. To use the tool, you must have the "Modify All Data" permission.

## Configuring Full Sandboxes

When you create or refresh a full sandbox, you can configure it to determine what data is copied. Minimizing the amount of data you include speeds up your sandbox copy.

- The Object History, Case History, Content Version History, and Opportunity History options allow you to select the number of days of history from your production org to copy to your sandbox. You can copy from 0 to 180 days of history, in 30-day increments. The default is 0 days.
- By default, Chatter data isn't copied to your sandbox. Chatter data includes feeds and messages. Select **Copy Chatter Data** if you want to include it.
- The setup audit trail history of your production org isn't copied to your sandbox. The audit trail for your sandbox org starts when you begin to use it.
- Archived activities (tasks and events that aren't available in the production org because they're over a year old) and password history (users' previous passwords) aren't copied.



**Note:** Don't increase the default selections unless special circumstances require it. Large amounts of data can significantly lengthen the time it takes to copy your sandbox.

## Accessing Sandbox

- Access changes for sandbox users:
  - A sandbox refresh deletes and recreates the sandbox as a new copy of the production org. In effect, this process reverses any manual access changes you've performed. If you created sandbox-only users, they no longer exist, and a user's profile and permissions revert to their values in the production org. After a refresh, make any access changes in the new copy.
  - You can create users in your production org that are inactive, and then activate them in a sandbox. This method is a good way to create a user that has the appropriate permissions to develop in a sandbox.
  - Many development and testing tasks require the "Modify All Data" permission. If your developers don't have that permission in the production org, increase their permissions in the sandbox. Exercise caution when granting this permission in sandbox orgs that contain sensitive information copied from production (for example, social security numbers).
  - Users added in a production org after creating or refreshing a sandbox don't have access to the production org instance's related sandboxes. To create users in a sandbox, log in as the administrator on the sandbox org and create them in the sandbox instance.

- You can create users for sandbox development, but these new users count against the number of licensed users in your org. To reduce your license count, you can disable production users who don't need access to the sandbox before you create or refresh a sandbox.
- Always log in to your sandbox org using the `https://test.salesforce.com` login URL.
- Remember to log in using the modified username as described in [Users and Contacts](#) on page 16.
- If using the API, after you log in, use the redirect URL that is returned in the `loginResult` object for subsequent access. This URL reflects the instance on which the sandbox is located and the appropriate server pool for API access.
- Sandbox copies are made with federated authentication with SAML disabled. Any configuration information is preserved, except the value for `Salesforce Login URL`. The `Salesforce Login URL` is updated to match your sandbox URL, for example `https://yourInstance.salesforce.com/`, after you re-enable SAML. To enable SAML in the sandbox, from Setup, enter *Single Sign-On Settings* in the *Quick Find* box, then select **Single Sign-On Settings**; then click **Edit**, and select `SAML Enabled`. Change the value of the `Salesforce Login URL` in the certificate for your client application as well.

## Sandbox Limits Notes

Sandboxes don't send email notifications when storage limits are reached. However, if you reach the storage limit of your sandbox, you can't save new data in your sandbox. To check your storage limits, from Setup, enter *Storage Usage* in the *Quick Find* box, then select **Storage Usage** in your sandbox.

## Customization and Data Changes

- Customizations and data changes in your production org aren't reflected in your existing sandboxes. Create or refresh a sandbox to incorporate the most recent customizations made to your org.
- You can only add, edit, or delete Apex using the Salesforce user interface in a Developer Edition or sandbox org. In a Salesforce production org, you can only make changes to Apex by using the `compileAndTestAPI()` call.
- If your sandbox is the same version as Salesforce AppExchange, you can:
  - Install and deploy apps from Salesforce AppExchange in your sandbox.
  - Publish apps from your sandbox to Salesforce AppExchange.

Publishing managed packages from a Lightning Platform Sandbox is not advised, as refreshing or deleting the sandbox prevents any revisions to that managed package.

The version of your sandboxes can differ from Salesforce AppExchange around the time of a Salesforce release. Check the logo in the upper left corner of your sandbox home page for version information.

- If your org uses quote templates, and you create a Developer Pro sandbox, templates that contain `Text/Image` fields can't be opened for editing within the sandbox.
- If your production org uses an image in quote templates and you copy the org to your sandbox, the image path is not correct and the image appears as a broken link. To display the image, reinsert it from the correct location on your sandbox.

## Service Exclusions

- The following features are disabled and can't be enabled in sandboxes.
  - Contract expiration warnings
  - Case escalation

Contract expiration warnings and case escalation are disabled because they automatically send email to contacts, customers, and production org users.

- Subscription summary
- Data exports (by clicking *Export Now* or *Schedule Export* on the Weekly Export Service page in Setup)
- The ability to create Salesforce sandboxes
- The ability to copy email service addresses that you create in your sandbox to your production org
- The ability to publish Site.com sites

## Other Service Differences

- Only custom links created as relative URLs, such as `/00Oz0000000EVpU&pv0={!Account_ID}`, work when copied to your sandboxes. Custom links created as absolute URLs, such as `https://yourInstance.salesforce.com/00Oz0000000EVpU&pv0={!Account_ID}`, don't work in your org's sandboxes. We recommend that you use only relative URLs in your production org. Otherwise, correct the URLs in each sandbox.
- Salesforce has a background process that permanently deletes records in the Recycle Bin that are older than 15 days. This process runs at different times on different servers, so its timestamp in your sandbox differs from its timestamp in your production org. Applications and integrations that depend on this timestamp can fail if they are first connected to one environment, such as your production org, and then connected to another environment, such as your sandbox. Keep this behavior in mind when developing applications and integrations that depend on this timestamp.

The time of the latest execution of the background delete process is available through the `getDeleted()` API call.

- For Salesforce authentication providers set up in the Summer '14 release and earlier, the user identity provided by a sandbox does not include the org ID. The destination org can't differentiate between users with the same user ID from two sources (such as two sandboxes). To differentiate users, edit the existing Salesforce Auth. Provider settings in the destination org, and select the checkbox to include the org ID for third-party account links. After you enable this feature, your users must reapprove the linkage to their third-party links. Salesforce authentication providers created in the Winter '15 release and later have this setting enabled by default.
- After an org's sandbox refresh is completed, a user has login access to a sandbox for 10 years after the sandbox refresh date if the user:
  - Is a system administrator.
  - Is copied into the sandbox from the production org, not created directly in the sandbox.
- To log in as any user, access your sandbox via [test.salesforce.com](https://test.salesforce.com). The option to log in as any user isn't available when users access a sandbox from production by using the Login link.

SEE ALSO:

[Create a Sandbox](#)

[Sandbox Types and Templates](#)

[Sandbox Licenses and Storage by Type](#)

[Unlock a Sandbox](#)

## Deploy Your Changes

---

Migrate metadata changes between Salesforce orgs by using the deployment tools available in Setup.

To access these pages, use the `Quick Find` box.

**Deployment Settings**

To use the change sets feature, a deployment connection is required. You can specify connection permissions for both outbound and inbound change sets on the Deployment Connections page.

**Deployment Status**

Monitor the progress of deployments made through the Metadata API.

**Outbound Change Sets**

Make changes in the org you are logged into, and upload those changes to another org.

**Inbound Change Sets**

Accept, modify, or reject change sets uploaded from other orgs.

## IN THIS SECTION:

[Choose Your Tools for Developing and Deploying Changes](#)

Whether you're an admin using point-and-click tools or a developer writing code, you can pick the right tool, work in a sandbox, and deploy complete changes to a production org. You can customize and code changes for your org in a sandbox using one, or more, of the tools provided by Salesforce.

[Connect Organizations for Deployment](#)

Deploy connections for change sets and authorize a deployment connection.

[Change Sets](#)

Use change sets to send customizations from one Salesforce org to another. For example, you can create and test a new object in a sandbox org, then send it to your production org using a change set. Change sets can only contain modifications you can make through the Setup menu. For example, you can't use a change set to upload a list of contact records. Change sets contain information about the org. They don't contain data, such as records.

[Special Behavior in Deployments](#)

When deploying changes to a Salesforce org, consider how individual components in your deployment behave so you're including all the necessary changes. Use the following information to determine what to include in your deployment, and how the changes appear in the destination org.

[Monitor Deployments](#)

You can monitor deployments that are in progress, check which deployments are waiting for execution, and view the results of completed deployments on the Deployment Status page.

[General Sandbox and Sandbox Storage Limits](#)

Limits for sandbox environments.

## SEE ALSO:

[Change Sets](#)[Monitor Deployments](#)

## Choose Your Tools for Developing and Deploying Changes

Whether you're an admin using point-and-click tools or a developer writing code, you can pick the right tool, work in a sandbox, and deploy complete changes to a production org. You can customize and code changes for your org in a sandbox using one, or more, of the tools provided by Salesforce.

## IN THIS SECTION:

- [Develop and Deploy Apex in the Developer Console](#)
- [Develop and Deploy Using the Force.com IDE](#)
- [Develop and Deploy Using SOAP API](#)
- [Deploy Using the Ant Migration Tool](#)
- [Deploy Using Change Sets](#)

## SEE ALSO:

- [Extend Salesforce with Clicks, Not Code](#)
- [Enhance Salesforce with Code](#)

## Develop and Deploy Apex in the Developer Console

The Developer Console is an integrated development environment with a collection of tools you can use to create, debug, and test applications in your Salesforce org.

## SEE ALSO:

- [Open the Developer Console](#)
- [Developer Console Functionality](#)

## Develop and Deploy Using the Force.com IDE

You can download the Force.com IDE to help you code projects for your organization. Using this tool, you can also compile and test the code you write, synchronize changes in a sandbox, and deploy your code to a production organization.

For more information, see the [Force.com IDE page](#).

 **Note:** The Force.com IDE is a free resource provided by Salesforce to support its users and partners, but is not considered part of our Services for purposes of the Salesforce Master Subscription Agreement.

## SEE ALSO:

- [Develop and Deploy Using SOAP API](#)
- [Choose Your Tools for Developing and Deploying Changes](#)

## EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **Performance, Unlimited, Developer, Enterprise,** and **Database.com** Editions

## USER PERMISSIONS

To use the Apex Deployment Tool:

- Author Apex

## EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **Performance, Unlimited, Developer, Enterprise,** and **Database.com** Editions

## Develop and Deploy Using SOAP API

You can use the SOAP API to develop and deploy changes to a development or sandbox organization, programmatically.

For more information about the SOAP API, and other API, see the [SOAP API Developer Guide](#).

SEE ALSO:

[Choose Your Tools for Developing and Deploying Changes](#)

## Deploy Using the Ant Migration Tool

Download the Ant Migration Tool if you want to perform a file-based deployment of metadata changes and Apex classes from a Developer Edition or sandbox organization to a production organization using Apache's Ant build tool.

1. [Download the .zip file of the Winter '18 Ant Migration Tool](#). (You can also download a .zip file containing a preview version of the [Spring '18 Ant Migration Tool](#).) The download link doesn't require authentication to Salesforce. If you're logged in to Salesforce, we recommend you log out before accessing the link in your browser.
2. Save the `salesforce_ant.zip` file and unzip its contents to the location of your choice.

The `salesforce_ant.zip` file contains the files you need to run an ant task that exercises the `compileAndTest` API call, including:

- A `Readme.html` file that explains how to use the tools
- A Jar file containing the ant task: `ant-salesforce.jar`
- A sample folder containing:
  - A `codepkg\classes` folder that contains `SampleDeployClass.cls` and `SampleFailingTestClass.cls`
  - A `codepkg\triggers` folder that contains `SampleAccountTrigger.trigger`
  - A `mypkg\objects` folder that contains the custom objects used in the examples
  - A `removecodepkg` folder that contains XML files for removing the examples from your organization
  - A sample `build.properties` file that you must edit, specifying your credentials, in order to run the sample ant tasks in `build.xml`
  - A sample `build.xml` file, that exercises the `deploy` and `retrieve` API calls

 **Note:** The Ant Migration Tool is a free resource provided by Salesforce to support its users and partners, but is not considered part of our Services for purposes of the Salesforce Master Subscription Agreement.

SEE ALSO:

[Force.com Migration Tool Guide](#)

[Choose Your Tools for Developing and Deploying Changes](#)

### EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **Performance, Unlimited, Developer, Enterprise,** and **Database.com** Editions

### EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **Performance, Unlimited, Developer, Enterprise,** and **Database.com** Editions

### USER PERMISSIONS

To use the Apex Deployment Tool:

- Author Apex

## Deploy Using Change Sets

You can deploy workflows, rules, Apex classes and triggers, and other customization from a sandbox organization to your production organization. You can create an outbound change set in the Salesforce user interface and add the components that you would like to upload and deploy to the target organization. To access change sets, from Setup, enter *Outbound Change Sets* in the **Quick Find** box, then select **Outbound Change Sets**.

SEE ALSO:

[Change Sets](#)

[Choose Your Tools for Developing and Deploying Changes](#)

## Connect Organizations for Deployment

Deploy connections for change sets and authorize a deployment connection.

IN THIS SECTION:

[Deployment Connections for Change Sets](#)

A deployment connection is required between two Salesforce orgs to send change sets from one org to another. You can't create deployment connections between arbitrary orgs. Instead, you create connections between all orgs affiliated with a production org. For example, if you have a production org and two sandboxes, a deployment connection is created between production and each sandbox. Also, a deployment connection is created between the two sandboxes.

[Authorizing a Deployment Connection](#)

Authorize inbound changes so that another Salesforce org can send change sets to the org you are logged into.

[Viewing Available Deployment Connections](#)

A deployment connection enables customizations to be copied from one Salesforce org to another. The deployment connections list shows which orgs are authorized to upload changes to this org, and which orgs allow this org to upload changes to them.

[Viewing Details of a Deployment Connection](#)

A deployment connection enables customizations to be copied from one Salesforce org to another. The deployment connections list shows which orgs are authorized to upload changes to this org, and which orgs allow this org to upload changes to them.

## Deployment Connections for Change Sets

A deployment connection is required between two Salesforce orgs to send change sets from one org to another. You can't create deployment connections between arbitrary orgs. Instead, you create connections between all orgs affiliated with a production org. For example, if you have a production org and two sandboxes, a deployment connection is created between production and each sandbox. Also, a deployment connection is created between the two sandboxes.

A deployment connection alone doesn't enable change sets to be sent between orgs. Each org must be authorized to send and receive change sets. This added level of security enforces code promotion paths and keeps orgs' setup metadata from being overwritten by mistake.

### EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in **Enterprise, Performance, Professional, Unlimited,** and **Database.com** Editions

### EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in **Enterprise, Performance, Professional, Unlimited,** and **Database.com** Editions

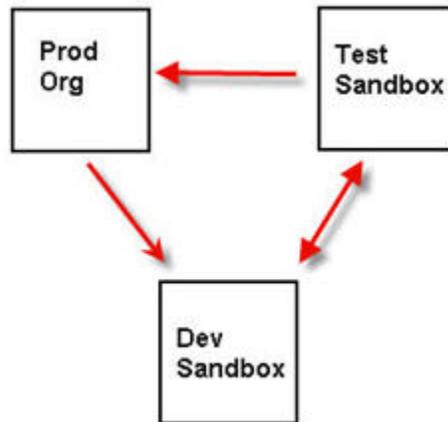
### USER PERMISSIONS

To edit deployment connections:

- **Deploy Change Sets AND Modify All Data**

For example, the following figure illustrates one possible migration path for a production org (Prod) and two sandboxes (Dev and Test). In this example, the production org can only receive changes that have been fully tested, so only the Test sandbox is authorized to upload change sets to production. To synchronize development projects with the production org, the Prod org can send change sets to the Dev sandbox, but not to the Test sandbox. Finally, because the features in development need iterative testing, Dev and Test sandboxes should be able to send change sets back and forth.

#### Change Set Authorization Enforces Code Path



 **Note:** This illustration describes one possible code migration path. Your department must create its own policies for orgs to send and receive change sets to one another.

#### SEE ALSO:

- [Deploy a Change Set](#)
- [Viewing Available Deployment Connections](#)
- [Authorizing a Deployment Connection](#)
- [Viewing Details of a Deployment Connection](#)

## Authorizing a Deployment Connection

Authorize inbound changes so that another Salesforce org can send change sets to the org you are logged into.

1. From Setup, enter *Deployment* in the *Quick Find* box, then select **Deployment Settings**, and then click **Continue**.
2. Click **Edit** next to the org you want to authorize.
3. Select **Allow Inbound Changes**.
4. Click **Save**.

#### SEE ALSO:

- [Viewing Available Deployment Connections](#)
- [Viewing Details of a Deployment Connection](#)
- [Deployment Connections for Change Sets](#)

## Viewing Available Deployment Connections

A deployment connection enables customizations to be copied from one Salesforce org to another. The deployment connections list shows which orgs are authorized to upload changes to this org, and which orgs allow this org to upload changes to them.

To view available connections, from Setup, enter *Deployment* in the **Quick Find** box, then select **Deployment Settings**.

### Action

Click **Edit** next to the org that you want to allow or disallow change sets from.

### Name

A list of orgs that have deployment connections to the org you are currently logged into. Click the name of an org to view more information about the connection.

### Description

A brief description of the connected orgs.

### Type

The type of org you are connected to. Possible values are Production, Full Copy Sandbox, Configuration Only Sandbox, and Developer Sandbox.

### Upload Authorization Direction

The arrows show the direction in which uploads can occur. A broken line means that no change sets are authorized in either direction. To authorize the connected org to send you inbound change sets, edit the deployment connection for this org. If you want to send outbound change sets to a connected org, the administrator for that org must edit the connection for that org.

### SEE ALSO:

[Authorizing a Deployment Connection](#)

[Viewing Details of a Deployment Connection](#)

[Deployment Connections for Change Sets](#)

## Viewing Details of a Deployment Connection

A deployment connection enables customizations to be copied from one Salesforce org to another. The deployment connections list shows which orgs are authorized to upload changes to this org, and which orgs allow this org to upload changes to them.

To view connection details:

1. From Setup, enter *Deployment* in the **Quick Find** box, then select **Deployment Settings**.
2. Click the name of the org you want to view.

### Name

The name of the selected org. This is not the org you are logged into.

### Description

A brief description of the org.

### Type

The type of org you are connected to. Possible values are Production, Full, Partial Copy, Developer Pro, and Developer.

### Allow Inbound Changes

If selected, the named org can send change sets to the org you are currently logged into. This is a read-only field and can only be modified by selecting Allow Inbound Changes in the target org.

**Accepts Outbound Changes**

If selected, the named org allows change sets to be sent to it from the org you are currently logged into.

## SEE ALSO:

[Authorizing a Deployment Connection](#)

[Viewing Available Deployment Connections](#)

[Deployment Connections for Change Sets](#)

## Change Sets

Use change sets to send customizations from one Salesforce org to another. For example, you can create and test a new object in a sandbox org, then send it to your production org using a change set. Change sets can only contain modifications you can make through the Setup menu. For example, you can't use a change set to upload a list of contact records. Change sets contain information about the org. They don't contain data, such as records.

When you want to send customizations from your current org to another org, create an *outbound change set*. Once you send the change set, the receiving org sees it as an *inbound change set*.

Sending a change set between two orgs requires a deployment connection. Change sets can only be sent between orgs that are affiliated with a production org. For example, a production org and a sandbox, or two sandboxes created from the same org can send or receive change sets.

## IN THIS SECTION:

[Permission Sets and Profile Settings in Change Sets](#)

Developers can use permission sets or profile settings to specify permissions and other access settings in a change set. When deciding whether to use permission sets, profile settings, or a combination of both, consider the similarities and differences.

[Components Available in Change Sets](#)

The components available for a change set vary by experience and edition. Also, some components require corresponding features to be enabled in your Salesforce org.

[Change Sets Implementation Tips](#)

Review these tips before you implement your change sets.

[Change Sets Best Practices](#)

Review these best practices about dependencies, validation, and access settings.

[Deploy Inbound Change Sets](#)

An *inbound change set* is a change set that has been sent from another Salesforce org to the org you are logged in to. A change set must be *deployed* for the changes to take effect. You can deploy the contents of an inbound change set as a whole, but not on a component-by-component basis.

**EDITIONS**

Available in: both Salesforce Classic and Lightning Experience

Available in **Enterprise, Performance, Professional, Unlimited,** and **Database.com** Editions

**USER PERMISSIONS**

To edit deployment connections:

- Deploy Change Sets AND Modify All Data

To use outbound change sets:

- Create and Upload Change Sets

To use inbound change sets:

- Deploy Change Sets AND Modify All Data

[Upload Outbound Change Sets](#)

An *outbound change set* is a change set created in the Salesforce org in which you are logged in and that you want to send to another org. You typically use an outbound change set for customizations created and tested in a sandbox and that are then sent to a production org.

SEE ALSO:

[Deployment Connections for Change Sets](#)

[Special Behavior in Deployments](#)

[Release Management Video: From Sandbox to Production How To Series \(Salesforce Classic\)](#)

## Permission Sets and Profile Settings in Change Sets

Developers can use permission sets or profile settings to specify permissions and other access settings in a change set. When deciding whether to use permission sets, profile settings, or a combination of both, consider the similarities and differences.

 **Note:** In API version 40.0 and later, when you deploy the output of a retrieval to another org, the metadata in the deployment replaces the target org metadata. In API version 39.0 and earlier, when you deploy your retrieved permission set output to another org, the deployment contents are merged with your current org data. For example:

- In API version 40.0 and later, if your permission set contains the Manage Roles user permission and you deploy a metadata file without this user permission, Manage Roles is disabled in the target org. Or, let's say you have a permission set with edit access to fields in an object. If the change set fails to include these fields, the permissions are still carried over to the target org and these changes are deployed.
- In API version 39.0 and earlier, if you deploy a metadata file without this user permission, Manage Roles remains enabled.

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in **Enterprise, Performance, Unlimited,** and **Database.com** Editions

Permission sets available in: **Contact Manager, Professional, Group, Enterprise, Performance, Unlimited, Developer,** and **Database.com** Editions

Behavior	Permission Sets	Profile Settings
Included permissions and settings	<ul style="list-style-type: none"> <li>• Standard object permissions</li> <li>• Standard field permissions</li> <li>• User permissions (such as "API Enabled")</li> </ul> <p> <b>Note:</b> Assigned apps and tab settings are <i>not</i> included in permission set components.</p>	<ul style="list-style-type: none"> <li>• Tab settings</li> <li>• Page layout assignments</li> <li>• Record type assignments</li> <li>• Login IP ranges</li> <li>• User permissions</li> </ul>
Included permissions and settings that require supporting components	<ul style="list-style-type: none"> <li>• Apex class access</li> <li>• Visualforce page access</li> </ul>	<ul style="list-style-type: none"> <li>• Assigned apps</li> <li>• Custom object permissions</li> <li>• Custom field permissions</li> <li>• Apex class access</li> <li>• Visualforce page access</li> </ul>

Behavior	Permission Sets	Profile Settings
Added as a component?	Yes	No. Profiles are added in a separate setting.

For Visualforce page access and Apex class access, always include supporting components in the change set.

 **Note:** Login IP ranges included in profile settings overwrite the login IP ranges for any matching profiles in the target org.

SEE ALSO:

[Deploy Inbound Change Sets](#)

[Upload Outbound Change Sets](#)

[Change Sets Best Practices](#)

## Components Available in Change Sets

The components available for a change set vary by experience and edition. Also, some components require corresponding features to be enabled in your Salesforce org.

 **Note:**

- If you create or modify components unavailable in a change set, you can't send those components from one org to another in a change set. In this case, migrate the changes manually by repeating the steps you performed when you created or modified the component.
- By default, list views are visible to all users when you deploy a change set. You can deploy a change set with Restricted Visibility, or change the visibility in the destination org if necessary.
- Deployed custom tabs are hidden by default for all users. They're visible only if the change set also contains profiles that set the visibility property appropriately. Professional Edition orgs are an exception—deployed custom tabs in those orgs are always visible by default.
- Reports stored in the My Personal Custom Reports folder (private reports) don't appear in the list of reports that can be added to the change set. Reports stored in the Unfiled Public Reports folder appear in the list of reports that can be added to the change set, but they aren't deployed even if added to the change set. To deploy a private or unfiled report using a change set, first copy or move the report to a different report folder.

The following types of components can be added to a change set.

- Action
- Action Link Group Template
- Apex Class
- Apex Sharing Reason
- Apex Trigger
- App
- Approval Process
- Asset File
- Assignment Rule
- Auth. Provider
- Auto-Response Rule

- Button or Link
- CORS Whitelist Origin
- Call Center
- Chatter Extension
- Communication Channel Layout
- Compact Layout
- Content Security Policy Trusted Site
- Custom Feed Filter
- Custom Field
- Custom Label
- Custom Metadata Type
- Custom Object
- Custom Permission
- Custom Report Type
- Custom Setting
- Dashboard
- Data Service
- Document
- Duplicate Rule
- EclairNG Map GeoJson
- Email Template
- Escalation Rule
- External Data Source
- External Service Registration
- Field Mapping
- Field Set
- Flow Definition
- Folder
- Global Value Set
- Group
- Home Page Component
- Home Page Layout
- Letterhead
- Lightning Component Bundle
- Lightning Page
- List View
- Matching Rule
- Microsoft® Outlook® Web App Domain
- Named Credential
- Network

- Page Layout
- Path Assistant
- Permission Set
- Platform Cache Partition
- Post Template
- Queue
- Record Type
- Remote Site
- Report
- Reporting Snapshot
- Role
- S-Control
- Security Custom Baseline
- Send Action
- Sharing Criteria Rule
- Sharing Owner Rule
- Site.com
- Static Resource
- Tab
- User Provisioning Config
- Validation Rule
- Visualforce Component
- Visualforce Page
- Whitelisted Site
- Workflow Email Alert
- Workflow Field Update
- Workflow Outbound Message
- Workflow Rule
- Workflow Task
- Zone

#### IN THIS SECTION:

[Restrictions for Approval Processes in Change Sets](#)

Understand these restrictions before you include approval processes in change sets.

#### SEE ALSO:

[Validate a Change Set](#)

[Create an Outbound Change Set](#)

[Select Components for an Outbound Change Set](#)

[Special Behavior in Deployments](#)

## Restrictions for Approval Processes in Change Sets

Understand these restrictions before you include approval processes in change sets.

- If the approval page fields include any custom fields on standard objects, you need to manually add those custom fields to outbound change sets. The `View/Add Dependencies` option for selecting change set components won't include these fields.
- If the approval process references any post templates that contain custom fields, then you need to resave those post templates in the originating organization before adding them to the change set. From Setup, enter `Post Templates` in the `Quick Find` box, then select **Post Templates**. For each post template, click **Edit** and then **Save**.
- Change sets don't include the order of active approval processes from the source org. You might need to reorder the approval processes in the destination org after deployment.
- If you change the `Unique Name` of an approval process that was previously included in a change set and deployed in another organization, and you resend the approval process via a change set, a new approval process will be created upon deployment in the other organization. The previously deployed approval process will not be modified.

## Change Sets Implementation Tips

Review these tips before you implement your change sets.

### Authorization required to upload changes

Before you can deploy a change set from one org to another, an administrator in the target org must authorize uploads across the deployment connection between the two orgs.

### Deployment Connections list displays all connections

The Deployment Connections list is automatically populated with your production org and all sandboxes. It is possible to deploy between any of these orgs, but no other orgs.

### Change set connections unavailable during maintenance

Authorizing deployment connections and uploading pages require information from the production org, and are unavailable when production is undergoing maintenance. During this time you can construct outbound change sets but not upload them.

### Sandboxes must be available

If an org has no sandboxes provisioned, the user may see an Insufficient Privileges error on the Deployment Connections page.

### Deployment doesn't automatically restart

If an error occurs during change set validation or deployment, you must manually restart the process. Be sure your org is not locked, undergoing maintenance, or otherwise inaccessible.

### Deployment is a one-way transaction

A change set is deployed in a single transaction. If the deployment is unable to complete for any reason, the entire transaction is rolled back. After a deployment completes successfully, all changes are committed to your org and the deployment can't be rolled back.

### Deployments maintain user references

If a component in a change set refers to a specific user, such as recipients of workflow email notifications or dashboard running users, then during deployment the system attempts to locate a matching user in the destination org by comparing usernames.

When you copy data to a sandbox, the fields containing usernames from the production org are altered to include the sandbox name. For example, in a sandbox named `test`, the username `user@acme.com` becomes `user@acme.com.test`. During a deployment using change sets, the `.test` in the username is ignored. This process transfers a user added to a component in one sandbox to other sandboxes or production orgs.

### Change sets with many dependent components

Opening a change set in Salesforce can take several minutes if it contains a component with many dependencies or if a component's parent has many dependencies. The delay is because Salesforce checks component dependencies before displaying the change set

page. An example of a component with many dependencies is a custom field that belongs to a custom object with 2,500 dependent components.

SEE ALSO:

[Change Sets Best Practices](#)

[Special Behavior in Deployments](#)

## Change Sets Best Practices

Review these best practices about dependencies, validation, and access settings.

### **Deploy all dependent components**

Make sure each outbound change set contains all interdependent components that don't exist in the target org. If you try to deploy a component that refers to another component missing from the target org and from the change set, the deployment fails.

Change sets give you fine-grained control over what you deploy. For example, you can migrate custom fields individually. To deploy a custom object and all of its fields, you must add the custom object and every field to the change set; adding just the custom object to the change set won't cause deployment to fail, but results in an empty custom object.

### **Add permissions and access settings to outbound change sets**

Adding profiles or permission sets to outbound change sets allows administrators to migrate permissions for users so they can access the new functionality. There are significant differences between permission sets and profile settings in change sets. For details, see ["Permission Sets and Profile Settings in Change Sets"](#) on page 28.

### **Clone a change set to add dependent components to an uploaded change set**

After you upload a change set, you can't change its contents. If you need to add dependent components to a change set you already uploaded, clone the change set, add the dependent components, and then upload it again.

### **Use distinct names for global publisher layouts and Outlook publisher layouts**

When you add page layouts to an outbound change set, the type for global publisher layouts and Outlook publisher layouts isn't displayed. Make sure that you provide unique names for your global publisher layouts and Outlook publisher layouts so that you can differentiate them in an outbound change set.

### **Plan deployments around maintenance schedule**

Plan your deployment activities around the maintenance schedule for both your production and sandbox orgs. Some features require information from your production org when accessed from a sandbox. In addition, the originating org is locked while validating an outbound change set, and the target org is locked while deploying an inbound change set. (When change sets lock an org, you can still read and write data to the org, but you can't make any setup changes that would modify the metadata.)

### **Validate change sets before deployment**

You can perform a test deployment of an inbound change set to view the success or failure messages that would occur with an actual deployment. This is a good idea if you are planning a deployment on a schedule (for example during low-use hours) and want to determine if the deployment will succeed ahead of time. However, you don't need to perform a test deployment every time you deploy, as this process takes time to complete and the org is locked for the duration. (You can still read and write data to the org, but you can't make any setup changes that would modify the metadata.) To test deploy an inbound change set, click its name and then click **Validate**.

### **View component details**

You can view the XML representation of a component after you upload an outbound change set or before you deploy an inbound change set.

### **Limit change sets to 10,000 files**

Change sets are limited to 10,000 files. If your change set exceeds this limit, you can create separate change sets for email templates, dashboards, and reports. These components are often the most numerous and have fewer dependencies.

**Delete or rename components using the Web interface**

You can't use change sets to delete or rename components. To delete components, use the Web interface on the target org. To rename a component, first delete the component on the target org and then upload the new component in a change set.

**Consider possible delays in deployment time when a change set includes field type changes**

If a change set includes changes to custom field types, the deployment time might be delayed by an extended period of time because custom field type changes might require changes in a large number of records. To avoid long delays in deployment, an alternative is to apply the field type change manually after the change set is deployed.

**Plan for tests to run in the target org**

When a change set is deployed to a production org, all local Apex tests in that org are run by default if you're deploying any Apex classes or triggers. If the target org is a sandbox, however, tests aren't automatically run.

SEE ALSO:

[Change Sets Implementation Tips](#)

[Special Behavior in Deployments](#)

## Deploy Inbound Change Sets

An *inbound change set* is a change set that has been sent from another Salesforce org to the org you are logged in to. A change set must be *deployed* for the changes to take effect. You can deploy the contents of an inbound change set as a whole, but not on a component-by-component basis.

Watch a Demo: [▶ Release Management: Deploying Changes Using Change Sets \(Salesforce Classic\)](#)

IN THIS SECTION:

[Viewing Inbound Change Sets](#)

The Inbound Change Sets page lists change sets awaiting deployment, as well as the history of deployed change sets.

[Viewing Change Set Details](#)

The Change Sets detail page lists information about a particular change set.

[Validate a Change Set](#)

You can validate a change set without deploying changes. By validating you can view the success or failure messages you would receive with an actual deploy.

[Deploy a Change Set](#)

[Monitor Deployments of Change Sets](#)

The size and complexity of the change set determines how long it takes for a change set to deploy. During this time, it can be helpful to monitor the deployment.

SEE ALSO:

[Change Sets](#)

## Viewing Inbound Change Sets

The Inbound Change Sets page lists change sets awaiting deployment, as well as the history of deployed change sets.

To view inbound change sets, from Setup, enter *Inbound Change Sets* in the Quick Find box, then select **Inbound Change Sets**.

**USER PERMISSIONS**

To deploy inbound change sets:

- Deploy Change Sets AND Modify All Data

 **Note:** Inbound change sets are permanently deleted six months after the change set is uploaded.

SEE ALSO:

[Viewing Change Set Details](#)

[Validate a Change Set](#)

[Deploy a Change Set](#)

## Viewing Change Set Details

The Change Sets detail page lists information about a particular change set.

1. From Setup, enter *Inbound Change Sets* in the Quick Find box, then select **Inbound Change Sets**.
2. Click the name of a change set.

SEE ALSO:

[Viewing Inbound Change Sets](#)

[Validate a Change Set](#)

[Deploy a Change Set](#)

## Validate a Change Set

You can validate a change set without deploying changes. By validating you can view the success or failure messages you would receive with an actual deploy.

1. From Setup, enter *Inbound Change Sets* in the Quick Find box, then select **Inbound Change Sets**.
2. Click the name of a change set.
3. Click **Validate**.

 **Note:** We recommend starting a validation during off-peak usage time and limiting changes to your org while the validation is in progress. The validation process locks the resources that are being deployed. Changes you make to locked resources or items related to those resources while the validation is in progress can result in errors.

4. After the validation completes, click **View Results**.

 **Note:** If you change a field type from Master-Detail to Lookup or vice versa, the change isn't supported when using the **Validate** option to test a deployment. This change isn't supported for test deployments to avoid data loss or corruption. If a change that isn't supported for test deployments is included in the deployment package, the test deployment fails and issues an error.

If your deployment package changes a field type from Master-Detail to Lookup or vice versa, you can still validate the changes before you deploy to production. Perform a full deployment to another test sandbox. A full deployment includes a validation of the changes as part of the deployment process.

Change sets that have been successfully validated might qualify for a quick deployment. For more information, see [Quick Deployments](#).

SEE ALSO:

[Viewing Inbound Change Sets](#)

[Viewing Change Set Details](#)

[Deploy a Change Set](#)

## Deploy a Change Set

1. From Setup, enter *Inbound Change Sets* in the *Quick Find* box, then select **Inbound Change Sets**.
2. In the Change Sets Awaiting Deployment list, click the name of the change set you want to deploy.
3. Click **Deploy**.

Alternatively, you can perform a quick deployment to shorten your deployment time to production. Change sets that have been successfully validated might qualify for a quick deployment. For more information, see [Quick Deployments](#).

A change set is deployed in a single transaction. If the deployment is unable to complete for any reason, the entire transaction is rolled back. After a deployment completes successfully, all changes are committed to your org and the deployment can't be rolled back.

 **Note:** The Lightning platform requires that at least 75% of your code is covered by unit tests before you can deploy it to a production org. Ideally, strive for 100% coverage. The code coverage restriction is not enforced for sandbox or Developer Edition orgs.

## Deployment Options

To prevent a deployment from failing when components are referenced by Apex jobs, in the Deployment Settings page, click **Allow deployments of components when corresponding Apex jobs are pending or in progress**, and then click **Save**. This option lets you deploy components that are referenced by Apex jobs—including scheduled jobs, batch jobs, and future methods—that are pending or in progress. This option applies to change sets and deployments that are started through the Metadata API.

 **Note:**

- Enabling this option might cause Apex jobs to fail due to unsupported changes.
- This option doesn't affect editing and saving Apex code in the Salesforce user interface (in Setup or the Developer Console), which will fail if there are active jobs associated with the Apex class. Updating Apex code by using the Force.com IDE can succeed if this option is enabled because Metadata API is used for the code update.

### SEE ALSO:

[Viewing Inbound Change Sets](#)

[Viewing Change Set Details](#)

[Special Behavior in Deployments](#)

[Monitor Deployments of Change Sets](#)

## Monitor Deployments of Change Sets

The size and complexity of the change set determines how long it takes for a change set to deploy. During this time, it can be helpful to monitor the deployment.

Track the status of deployments that are in progress in the Deployment Status page. (From Setup, enter **Deployment Status** in the *Quick Find* box, then select **Deployment Status**.) The Deployment Status page also shows completed deployments.

Alternatively, you can check completed deployments on the Change Set Detail page. To access this page from Setup, enter *Inbound Change Sets* in the **Quick Find** box, select **Inbound Change Sets**, and then click the name of a deployed change set. Deployments for the change set are listed under the Deployment History section.

SEE ALSO:

[Deploy a Change Set](#)

[Deployment Connections for Change Sets](#)

[Monitor Deployments](#)

## Upload Outbound Change Sets

An *outbound change set* is a change set created in the Salesforce org in which you are logged in and that you want to send to another org. You typically use an outbound change set for customizations created and tested in a sandbox and that are then sent to a production org.

Watch a Demo: [Release Management: Deploying Changes Using Change Sets \(Salesforce Classic\)](#)

 **Note:** A change set can have up to 10,000 files with a total file size of 400 MB. Change set components are represented as metadata XML files. Make sure that your change set doesn't exceed approximately 5,000 components.

Sending an outbound change set to another org doesn't guarantee that the changes are implemented in that org. The change set must be deployed by the target org before the changes can take effect.

 **Tip:** To help ensure a smooth deployment, review information about permission sets and profile settings in change sets.

After you upload a change set, its status becomes closed, and you can't make changes to its components. Also, the components in a closed change set don't get refreshed. To redeploy the same set of components, clone the change set. The cloned change set includes the latest changes to its components source. You can add and remove components in the cloned change set. Cloning change sets is helpful during the iterative phases of a project.

IN THIS SECTION:

[Select Components for an Outbound Change Set](#)

[View and Add Dependent Components to a Change Set](#)

A dependency is a relationship where one or more components must exist for another component to exist. Add dependent components to a change set, unless the dependent components exist in every org where this change set is deployed.

[Upload an Outbound Change Set](#)

Once you've assembled the components in a change set, you can upload it to another Salesforce org. After you upload a change set, you can't edit it or recall it.

[Create an Outbound Change Set](#)

An outbound change set is a change you send from the Salesforce org you are logged into to another org.

[Clone an Outbound Change Set](#)

You can create a copy of an existing change set by cloning it.

[Delete an Outbound Change Set](#)

[Outbound Change Set Validation Errors](#)

### USER PERMISSIONS

To create, edit, or upload outbound change sets:

- [Create and Upload Change Sets](#)

[Uploading Change Sets During Staggered Salesforce Service Upgrades](#)

SEE ALSO:

[Permission Sets and Profile Settings in Change Sets](#)

[Deploy Inbound Change Sets](#)

[Change Sets](#)

## Select Components for an Outbound Change Set

To select the components in an outbound change set:

1. From Setup, enter *Outbound Change Sets* in the **Quick Find** box, then select **Outbound Change Sets**.
2. In the Change Sets list, click the name of a change set, or create a new one.
3. Click **Add** to add components.
4. Choose the type of component and the components you want to add, and then click **Add to Change Set**.
5. Click **Add Profiles** to add profile settings to the change set.

 **Note:** You can't add profile settings to a change set in Professional Edition.

6. Optionally, click **View/Add Dependencies** to add dependent components.

 **Note:** Dependent components rely on the existence of other components. Unless you're certain that the dependent components exist in every org this change set will be deployed to, add dependent components to the change set.

SEE ALSO:

[Create an Outbound Change Set](#)

[View and Add Dependent Components to a Change Set](#)

[Components Available in Change Sets](#)

## View and Add Dependent Components to a Change Set

A dependency is a relationship where one or more components must exist for another component to exist. Add dependent components to a change set, unless the dependent components exist in every org where this change set is deployed.

To add dependent components to an outbound change set:

1. From Setup, enter *Outbound Change Sets* in the **Quick Find** box, then select **Outbound Change Sets**.
2. In the Change Sets list, click the name of a change set.
3. Click **View/Add Dependencies**.
4. On the Component Dependencies page, select the dependent components you wish to deploy and click **Add to Change Set**.

 **Warning:** If your change set contains more than 2500 dependencies, you can see only the first 2500 in the Component Dependencies page.

SEE ALSO:

[Select Components for an Outbound Change Set](#)

[Upload an Outbound Change Set](#)

[Components Available in Change Sets](#)

## Upload an Outbound Change Set

Once you've assembled the components in a change set, you can upload it to another Salesforce org. After you upload a change set, you can't edit it or recall it.

1. From Setup, enter *Outbound Change Sets* in the **Quick Find** box, then select **Outbound Change Sets**.
2. Click the name of a change set.
3. Select the org you want to send the change set to.
4. Click **Upload**.

 **Note:** Outbound change sets expire six months after upload. Change sets are permanently deleted when they expire.

SEE ALSO:

[Uploading Change Sets During Staggered Salesforce Service Upgrades](#)

[Create an Outbound Change Set](#)

## Create an Outbound Change Set

An outbound change set is a change you send from the Salesforce org you are logged into to another org.

To view outbound change sets, from Setup, enter *Outbound Change Sets* in the **Quick Find** box, then select **Outbound Change Sets**.

- To create a new change set, click **New**.
- To view the details of an existing change set, click its name.

SEE ALSO:

[Clone an Outbound Change Set](#)

[Outbound Change Set Validation Errors](#)

## Clone an Outbound Change Set

You can create a copy of an existing change set by cloning it.

1. From Setup, enter *Outbound Change Sets* in the **Quick Find** box, then select **Outbound Change Sets**.
2. Click the name of the change set you want to clone.

3. Click **Clone**.

SEE ALSO:

[Create an Outbound Change Set](#)

## Delete an Outbound Change Set

1. From Setup, enter *Outbound Change Sets* in the **Quick Find** box, then select **Outbound Change Sets**.
2. Click the name of the change set you want to delete.
3. Click **Delete**.

SEE ALSO:

[Create an Outbound Change Set](#)

## Outbound Change Set Validation Errors

If you receive an error about cross-version validation, the org used to create the outbound change set is running on a different version than the destination org. This error typically occurs during upgrades, because orgs may be upgraded at different times due to Salesforce staggered releases. If you receive this error, you can only deploy those components that are compatible between versions.

SEE ALSO:

[Create an Outbound Change Set](#)

[Upload an Outbound Change Set](#)

## Uploading Change Sets During Staggered Salesforce Service Upgrades

During Salesforce service upgrades, production and sandbox environments may not be running the same version of the platform because the upgrades are staggered. Some components may have new functionality that prevents you from deploying that type of component. You can deploy such components after the production org runs the same version as sandbox.

If you upload a change set with components that can't be deployed because of incompatible versions, the system detects which components can't be deployed. Salesforce gives you the option of uploading the remaining components.

SEE ALSO:

[Upload an Outbound Change Set](#)

## Special Behavior in Deployments

When deploying changes to a Salesforce org, consider how individual components in your deployment behave so you're including all the necessary changes. Use the following information to determine what to include in your deployment, and how the changes appear in the destination org.

The behaviors listed in the Metadata API section apply if you're using the Ant Migration Tool or Force.com IDE.

## Change Set Components

### Approval Processes

- If the approval page fields include any custom fields on standard objects, you need to manually add those custom fields to outbound change sets. The `View/Add Dependencies` option for selecting change set components won't include these fields.
- If the approval process references any post templates that contain custom fields, then you need to resave those post templates in the originating organization before adding them to the change set. From Setup, enter `Post Templates` in the `Quick Find` box, then select **Post Templates**. For each post template, click **Edit** and then **Save**.
- Change sets don't include the order of active approval processes from the source org. You might need to reorder the approval processes in the destination org after deployment.
- If you change the `Unique Name` of an approval process that was previously included in a change set and deployed in another organization, and you resend the approval process via a change set, a new approval process will be created upon deployment in the other organization. The previously deployed approval process will not be modified.

### Apex Classes and Apex Triggers

By default, changes to Apex code that have Apex jobs pending or in progress can't be deployed. To deploy these changes, do one of the following.

- Cancel Apex jobs before deploying changes to Apex code. Reschedule the jobs after the deployment.
- Enable deployments with Apex jobs in the Salesforce user interface in the Deployment Settings page.

### Custom Fields

- To change the data type of a custom field, you can use change sets. However, the deployment might be delayed as many records are updated. Consider making this change to the target org through the user interface, instead.

### Custom Objects

Using API version 29.0 and earlier, you can't change the `sharingModel` of an object using Metadata API. You must manually make this change to the target org through the user interface.

Starting with API version 30.0, you can change the `sharingModel` of an object for internal users using Metadata API and the user interface.

Simultaneously inserting a custom object, updating the `sharingModel` field for an object, and adding a new sharing rule isn't supported. Instead, three separate deployments are required. First, deploy the custom object, then deploy the updated `sharingModel` for the object, and then deploy the new sharing rule.

### Flows

- If you plan to deploy a flow with change sets, consider limitations in migration support. Make sure your flows reference only fields and components that are available in change sets.
- You can include only one version of a flow in a change set.
- If the flow has no active version when you upload the outbound change set, the latest inactive version is used.
- When you view the dependent components for the change set, the Component Dependencies page lists the dependencies for `<i>all</i>` versions of the flow. Add all interdependent components for the relevant flow version to the outbound change set.
- An active flow in a change set is deployed to its destination as inactive. Activate the flow manually after deployment.
- Deploying or redeploying a flow with change sets creates a version of the flow in the destination organization.

### Permissions

See [Permission Sets and Profile Settings in Change Sets](#) on page 28.

**Page Layout**

A deployment containing a profile and record type, but not the assigned page layout for that record type, removes the existing layout assignment from the profile for that record type. Always include all page layouts for all required record types in the change set.

**Metadata API****Apex Classes and Apex Triggers**

By default, changes to Apex code that have Apex jobs pending or in progress can't be deployed. To deploy these changes, do one of the following.

- Cancel Apex jobs before deploying changes to Apex code. Reschedule the jobs after the deployment.
- Enable deployments with Apex jobs in the Salesforce user interface in the Deployment Settings page.

**Approval Processes**

- To use approval processes on Salesforce Knowledge articles with the Metadata API, the article type must be deployed. For article version (`_kav`) in approval processes, the supported action types are: Knowledge Action, Email Alert, Field Update, and Outbound Message.
- If the approval process references any post templates that contain custom fields, then you need to resave those post templates in the originating organization before adding them to the change set. From Setup, enter *Post Templates* in the *Quick Find* box, then select **Post Templates**. For each post template, click **Edit** and then **Save**.
- The metadata doesn't include the order of active approval processes. You might need to reorder the approval processes in the destination org after deployment.
- If you change the `Unique Name` of an approval process that was previously included in a change set and deployed in another organization, and you resend the approval process via a change set, a new approval process will be created upon deployment in the other organization. The previously deployed approval process will not be modified.

**Custom Fields**

Starting in API version 30.0, when deploying a new custom field, the default values for the `editable` and `readable` fields in profile field permissions are `false`. To override the default values, include field permissions for the new field in your profiles.

**Custom Objects**

Using API version 29.0, you can't change the `sharingModel` of an object using Metadata API. You must manually make this change to the target org through the user interface.

Starting with API version 30.0, you can change the `sharingModel` of an object for internal users using Metadata API and the user interface.

**Connected App**

- You cannot set the `consumerKey` in Metadata API. It is included in a retrieve operation for informational purposes. If you try to move the connected app to another org, you must remove the `consumerKey` from the `.zip` file before the deployment to an org. A new key will be generated in the destination org.
- Mobile settings of connected apps are not supported in change sets and must be manually migrated.

**Page Layout**

A deployment containing page layout assignments replaces all existing page layout assignments in the destination org with those specified in the `.zip` file. Existing page layouts in the org disappear if they're not included in the `.zip` file. Always include all page layouts for all required record types in the `.zip` file.

**Profiles**

If a package includes a profile with a name that doesn't exist in the target org, a new profile is created with that name. If the deployed profile doesn't specify any permissions or settings, the resulting profile consists of all the permissions and settings in the Standard Profile.

**Sharing**

- Simultaneously updating the `sharingModel` field for an object and adding a new sharing rule isn't supported in Metadata API, regardless of which object you're updating. For example, you can add a sharing rule when the org-wide default is public, and subsequently update the `sharingModel`. This would result in a single sharing recalculation.
- You might encounter an error if you're deploying a change set with a custom object that has a parent-child relationship without the master/detail field in the same change set. To resolve this error, include the master/detail custom field in the change set, even if you haven't changed the org-wide default.

**Workflow**

Test mode for flow triggers isn't supported in the Metadata API. If you want a flow trigger to run the latest flow version when an administrator causes the workflow rule to fire, enable test mode via the user interface after deployment.

## SEE ALSO:

[Deploy a Change Set](#)

[Change Sets](#)

[Components Available in Change Sets](#)

[Working with the Zip File](#)

## Monitor Deployments

You can monitor deployments that are in progress, check which deployments are waiting for execution, and view the results of completed deployments on the Deployment Status page.

This page lists all deployments—change sets, Metadata API-based deployments, including deployments started from the Force.com IDE and the Ant Migration Tool, and package installations.

The size and complexity of the metadata components affect the deployment time. To track the status of deployments that are in progress or have completed in the last 30 days, from Setup, enter *Deployment* in the *Quick Find* box, then select **Deployment Status**. Deployments are listed in different sections depending on their status.

### In-Progress and Queued Deployments

When running a deployment, the Deployment Status page shows you the real-time progress of your current deployment. This page contains charts that provide a visual representation of the overall deployment progress. The first chart shows how many components have already been deployed out of the total and includes the number of components with errors. For example, the following chart indicates that 302 components were processed successfully out of 450 and there were 45 components with errors.

**EDITIONS**

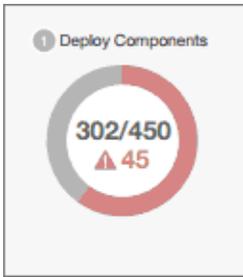
Available in: both Salesforce Classic and Lightning Experience

Available in: **Enterprise, Performance, Unlimited, Developer**, and **Database.com** Editions

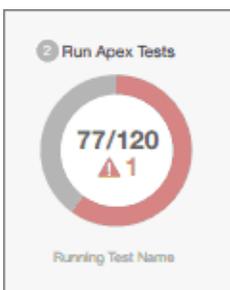
**USER PERMISSIONS**

To view metadata deployments:

- **Modify All Data**



After all components have been deployed without errors, Apex tests start executing, if required or enabled. A second chart shows how many Apex tests have run out of the total number of tests and the number of errors returned. In addition, the chart shows the name of the currently running test. For example, in the following chart, 77 tests have completed execution out of a total of 120, and 1 test failed.



The following information is displayed for the current deployment.

Field	Description
Name	The change set name or a unique identifier that's used to track the Metadata API deployment. For a Metadata API deployment, this value is returned by the <code>deploy()</code> call.
Type	The deployment type: Change Set or API.
Deployed By	The name of the user performing the deployment.
Start Time	The date and time when the deployment actually started, not the time the request was queued. This value is the time the deployment <code>Status</code> is set to In Progress.
Validated	The date and time when the deployment validation finished.

If the current deployment has errors, you can view these errors before the deployment finishes by clicking **View Errors**.

### Pending Deployments

You can initiate multiple deployments, but only one deployment can run at a time. The other deployments will remain in the queue waiting to be executed after the current deployment finishes. Queued deployments are listed under Pending Deployments in the order they will be executed.

### Deployment Validations

A deployment validation is a deployment that is used only to check the results of deploying components and is rolled back. A validation doesn't save any deployed components or change the Salesforce org in any way. You can determine whether a deployment is a validation

only (Validate) or an actual deployment (Deploy) by inspecting the information for pending deployments or the `Status` column of deployments in the Failed and Succeeded sections.

If a validation completed successfully in the last 10 days, and all tests passed with sufficient code coverage, you can perform a quick deployment by deploying this validation to production without running tests. See [Quick Deployments](#).

## Canceling a Deployment

You can cancel a deployment while it's in progress or in the queue by clicking **Cancel** next to the deployment. The deployment then has the status `Cancel_Requested` until the deployment is completely canceled. A canceled deployment is listed in the Failed section.

## Completed Deployments

Deployments that have finished are listed either in the Failed or Succeeded sections depending on their status.

Deployments that have finished but failed, and deployments that were canceled are listed in the Failed section. No changes were committed to the Salesforce org for these deployments because files were missing, components had errors, tests failed, or the deployment was canceled.

Deployments that have completed successfully or have partially succeeded are listed in the Succeeded section. Only deployments to a non-production org can partially succeed. These are deployments that have the `rollbackOnError` field set to `false` in the deployment options and have errors in a subset of components. In a partially succeeded deployment, the failed components aren't committed and the remaining components are committed to the org.

To get more details about a deployment, click **View Details** next to a deployment. Use the information on the Deployment Details page to view the errors and troubleshoot problems for a failed or partially succeeded deployment. The Deployment Details page includes the error messages that occurred during the deployment, errors for Apex tests with stack trace information, code coverage warnings, and information about slow tests. For a successful deployment, the Deployment Details page shows information about the deployment including how many components were deployed and how many Apex tests were run.

### Deployment Status

The `Status` column for completed deployments in the Failed and Succeeded sections lists the type and status of a deployment and has two parts:

- The prefix indicates whether the deployment is a validation only (Validate:) or an actual deployment (Deploy:).
- The second part of the status value contains the status of the deployment: Failed or Canceled for failed deployments, Succeeded for succeeded deployments, or Partially Succeeded for partially succeeded deployments.

## Quick Deployments

As part of a deployment, all Apex tests are run in production. If the production org contains many Apex tests, executing the tests can be time consuming and can delay your deployment. To reduce deployment time to production, you can perform a quick deployment by skipping the execution of tests. Quick deployments are available for change sets and Metadata API components when the following requirements are met.

- The components have been validated successfully for the target environment within the last 10 days.
- As part of the validation, Apex tests in the target org have passed.
- Code coverage requirements are met.
  - If all tests in the org or all local tests are run, overall code coverage is at least 75%, and Apex triggers have some coverage.
  - If specific tests are run with the **Run specified tests** test level, each class and trigger that was deployed is covered by at least 75% individually.

A validation is a deployment that's used only to check the results of deploying components and doesn't save any components in the org. A validation enables you to view the success or failure messages that you would receive with an actual deployment. You can validate change sets or metadata components through the API or the Ant Migration Tool.

To learn how to validate a change set, see [Validate a Change Set](#) in the Salesforce Help.

To validate components with the Ant Migration Tool, set the `checkOnly` option to `true` in the deploy target. See [Deploying Changes to a Salesforce Organization](#) in the *Ant Migration Tool Guide*.

### Performing a Quick Deployment through the User Interface or the API

To perform a quick deployment, first run a validation-only deployment with Apex test execution on the set of components that you need to deploy. If your validation succeeds and qualifies for a quick deployment, you can start a quick deployment.

You can quick-deploy validated change sets and Metadata API components in the user interface. In the Deployment Status page, deploy a recent validation by clicking **Quick Deploy** next to your validation or on the validation's detail page. This button appears only for qualifying validations.

## Deployment Details

Help for this Page ?

← Back to Deployment Status

### Validation Succeeded

Name: 0AfD00000004PvI  
Type: API  
Deployed By: [Test User](#)  
Start Time: 7/29/2014 11:23 AM  
End Time: 7/29/2014 11:23 AM

1 Deploy Components
2 Run Apex Tests

7/7

5/5

**Ready for Quick Deployment**

Enables the quick deployment of recently validated components by skipping Apex tests as part of the deployment.

Quick Deploy

Expires in 3d 23h

To learn how to perform a quick deployment of change sets and run specific tests, check out this video: [▶ Release Management: Deploy Changes Efficiently with Quick Deployments & Test Levels \(Salesforce Classic\)](#).

Alternatively, you can start a quick deployment through Metadata API or the Ant Migration Tool for Metadata API components (excluding change sets). For Metadata API, call `deployRecentValidation()` and pass it the validation ID. For the Ant Migration Tool, use the `<sf:deployRecentValidation>` task.

 **Note:** Quick Deploy is enabled for recent validations in which Apex tests have executed successfully and code coverage requirements have been met. Note the following.

- In production, quick deployments are supported for validations that meet the criteria. You can deploy recent validations of change sets and Metadata API components (including components validated by using the Ant Migration Tool).
- In sandbox, Quick Deploy is supported only for validations that explicitly enable test execution (for example, by choosing a test option when validating inbound sets or through the `testLevel` parameter for the Migration Tool). By default, Apex tests aren't required nor ran in sandbox deployments.
- If you perform a deployment after a validation, whether through Quick Deploy, a package installation, or a regular deployment, all validations no longer qualify for quick deployment. Revalidate the set of components to quick-deploy.

## Performance Tuning Resources for Long-Running Tests

If required or enabled, Apex tests run as part of a deployment after all components are deployed. Apex tests that take a long time to execute delay the entire deployment. The top-five long-running tests, that is the top-five tests that ran longer than two minutes, are flagged for a completed deployment in the Deployment Details page. You can improve the performance of these tests to make them more efficient and speed up future deployments. There can be many causes for slow performance. For example, accessing org data instead of using test data, or exercising SOQL queries or Apex code with poor performance. Here are some resources you can use to learn about performance best practices for Apex and SOQL.

- [Isolation of Test Data from Organization Data in Unit Tests](#)
- [Working with Very Large SOQL Queries](#)
- [Webinar: Inside the Lightning Platform Query Optimizer](#)
- [Query and Search Optimization Cheat Sheet](#)
- [Performance Tuning for Visualforce and Apex Webinar](#)
- [Architect Core Resources](#)

### SEE ALSO:

[Deploy Inbound Change Sets](#)

[Upload Outbound Change Sets](#)

[Metadata API Developer's Guide](#)

[Force.com Migration Tool Guide](#)

## General Sandbox and Sandbox Storage Limits

Limits for sandbox environments.

A sandbox is a copy of your organization in a separate environment, you can use for a variety of purposes, such as testing and training. Sandboxes are completely isolated from your Salesforce production organization and operations you perform in your sandboxes do not affect your Salesforce production organization.

The following limits apply to sandboxes.

- You can refresh a full sandbox 29 days after you created or last refreshed it. If you delete a full sandbox within those 29 days, you need to wait until after the 29 day period, from the date of last refresh or creation, to replace it.
- You can refresh a Partial Copy sandbox 5 days after you created or last refreshed it. If you delete a Partial Copy sandbox within those 5 days, you need to wait until after the 5 day period, from the date of last refresh or creation, to replace it.
- You can refresh a Developer or Developer Pro sandbox once per day.
- Enterprise Edition includes licenses for 25 Developer sandboxes and 1 Partial Copy sandbox.
- Performance Edition includes licenses for 1 full sandbox, 5 Developer Pro sandboxes, 100 Developer sandbox, and 1 Partial Copy sandbox.
- Unlimited Edition includes licenses for 1 full sandbox, 5 Developer Pro sandboxes, 100 Developer sandboxes and 1 Partial Copy sandbox.
- Professional Edition includes licenses for 10 Developer sandboxes. (Change sets aren't available.)
- If you need licenses for more sandboxes, contact Salesforce to order sandboxes for your organization.

The following limits apply to Sandbox storage limits.

- Partial Copy sandboxes have a 5 GB of files and a 5 GB of data storage limit.
- Developer Pro sandboxes have a 1 GB of files and a 1 GB of data storage limit.

- Developer sandboxes have a 200 MB of files and a 200 MB of data storage limit.
- Full sandboxes have the same storage limit as your production organization.
- Sandboxes don't send email notifications when storage limits are reached. However, if you reach the storage limit of your sandbox, you can't save new data in your sandbox. To check your storage limits, from Setup, enter *Storage Usage* in the Quick Find box, then select **Storage Usage** in your sandbox.

# INDEX

## A

Ant task for Apex 21

Apex

tools 21–22

Approval processes

change set restrictions 32

## C

Change sets

approval process restrictions 32

best practices 33

cloning 39

deleting 40

dependent components 38

deploying 36

deployment 34

deployment connections 24–26

details 35

implementation tips 32

inbound 34

outbound 37, 39–40

permission sets and profiles 28

permissions 27

selecting components 38

testing before deployment 35

version errors 40

Cloning a change set 39

Copy

organization 2

## D

Deleting a change set 40

Dependent components 38

Deploying

using Change Sets 24

using SOAP API 23

using the Force.com IDE 22

Deployment

monitoring 36

## I

Inbound change set 24–25, 34

## L

Limits

sandbox 47

## M

Metadata components in change sets 29

Monitoring deployment 36

## O

Organization

copy 2

Outbound change set 24–25, 37, 39–40

Outbound change sets

selecting components 38

version errors 40

## S

Sandbox

activating 8

cloning 10

creating 5–10

licenses 13

limits 47

managing 11

monitoring 11

refreshing 5–10

restrictions 13

status 12

Sandbox data template 15

## T

Test database 15

Tools for Apex 21–22