Salesforce Billing Developer Guide

'25

CONTENTS

Contents

Contents

CHAPTER 1 Salesforce Billing Fields on Standard Objects

Salesforce Billing has added custom fields to some standard Salesforce objects to represent information specific to billing. These fields are available only in orgs where the Salesforce Billing managed package is installed.

Billing Fields on Account

Standard and custom fields extend the standard Account object for use in Billing to represent information about accounts.

Fields

Field	Details
blng__BillToContact__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Primary contact for the billed account. This field is a relationship field. **Relationship Name** blng__BillToContact__r **Relationship Type** Lookup **Refers To** Contact
CustomerProfileId__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Authorize.Net payment gateways require this value to communicate with Salesforce Billing. For accounts with Authorize.Net payment gateways, Salesforce Billing creates a customer profile ID when a user selects **Save for future use** in the Payment Center. It also creates a profile ID when a user creates a credit card payment method on the account. Salesforce Billing then passes the ID to Authorize.Net when it communicates with the payment gateway.
blng__DefaultPaymentType__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Used in payment runs. A payment run picks up an invoice if the run and the invoice have matching Default Payment Type values. Salesforce Billing provides two ways to set Default Payment Type on the invoice: • Relate a payment method to the account and select the payment method's Autopay field. The account's default payment type inherits the value of the payment method's

Field	Details
	`Type` field. All the account's invoices inherit the account's default payment type. Use this setup to quickly set a default payment type for all of the account's invoices.
	• Enter a payment method in the invoice's `Override Autopay Payment Method` field. The invoice's default payment type ignores the account's payment method and inherits the `Type` value from the `Override Autopay Payment Method` instead. Use this setup to temporarily change the payment method used for a specific invoice.

SEE ALSO:

Object Reference for the Salesforce Platform: Account

Billing Fields on Asset

Standard and custom fields extend the standard Asset object for use in Billing to represent information about assets.

Fields

Field	Details
`blng__InvoiceLineBalanceUpdated__c`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date and time when balances of the invoice lines related to the asset were last updated.
`blng__InvoiceLineBalance__c`	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The sum of balances on all the posted invoice lines related to the asset.

SEE ALSO:

Object Reference for the Salesforce Platform: Asset

Billing Fields on Consumption Schedule

Standard and custom fields extend the standard Consumption Schedule object for use in Billing to represent information about consumption schedules.

Fields

Field	Details
`blng__BillingRule__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** This rule is used to invoice usage summaries related to the consumption schedule. This field is a relationship field. **Relationship Name** blng__BillingRule__r **Relationship Type** Lookup **Refers To** blng__BillingRule__r
`blng__RevenueRecognitionRule__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** This rule is used to recognize the revenue of usage summaries based on the summary's related revenue recognition rule. This field is a relationship field. **Relationship Name** blng__RevenueRecognitionRule__r **Relationship Type** Lookup **Refers To** blng__RevenueRecognitionRule__r
`blng__TaxRule__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** This rule is used to tax usage summary invoice lines based on the summary's related tax rule. This field is a relationship field. **Relationship Name** blng__TaxRule__r

Field	Details
	Relationship Type Lookup
	Refers To blng__TaxRule__r

SEE ALSO:

Object Reference for the Salesforce Platform: ConsumptionSchedule

Billing Fields on Finance Balance Snapshot

Standard and custom fields extend the standard Finance Balance Snapshot object for use in Billing to represent information about finance balance snapshots.

Fields

Field	Details
blng__CreditGLAccount__c	**Type** reference
	Properties Filter, Group, Nillable, Sort, Update
	Description The Credit General Ledger account related to the finance transaction. This field is a relationship field.
	Relationship Name blng__CreditGLAccount__r
	Relationship Type Lookup
	Refers To blng__GLAccount__c
blng__DebitGLAccount__c	**Type** reference
	Properties Filter, Group, Nillable, Sort, Update
	Description The Debit General Ledger account related to the finance transaction. This field is a relationship field.

Field	Details
	Relationship Name
	blng__DebitGLAccount__r
	Relationship Type
	Lookup
	Refers To
	blng__GLAccount__c
`blng__FinanceBook__c`	**Type**
	reference
	Properties
	Filter, Group, Nillable, Sort, Update
	Description
	The finance book related to the finance transaction.
	This field is a relationship field.
	Relationship Name
	blng__FinanceBook__r
	Relationship Type
	Lookup
	Refers To
	blng__FinanceBook__c
`blng__FinancePeriod__c`	**Type**
	reference
	Properties
	Filter, Group, Nillable, Sort, Update
	Description
	The finance period related to the finance transaction.
	This field is a relationship field.
	Relationship Name
	blng__FinancePeriod__r
	Relationship Type
	Lookup
	Refers To
	blng__FinancePeriod__c
`blng__GLRule__c`	**Type**
	reference
	Properties
	Filter, Group, Nillable, Sort, Update
	Description
	The General Ledger (GL) rule used to determine the GL accounts for the finance transaction.

Field	Details
	This field is a relationship field.
	Relationship Name blng__GLRule__r
	Relationship Type Lookup
	Refers To blng__GLRule__c
blng__GLTreatment__c	**Type** reference
	Properties Filter, Group, Nillable, Sort, Update
	Description The General Ledger (GL) treatment used to determine the GL accounts for the finance transaction. This field is a relationship field.
	Relationship Name blng__GLTreatment__r
	Relationship Type Lookup
	Refers To blng__GLTreatment__c

Billing Fields on Finance Transaction

Standard and custom fields extend the standard Finance Transaction object for use in Billing to represent information about finance transactions.

Fields

Field	Details
blng__CreditGLAccount__c	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The Credit General Ledger account related to the finance transaction. This field is a relationship field.

Field	Details

	Relationship Name
	blng__CreditGLAccount__r
	Relationship Type
	Lookup
	Refers To
	blng__GLAccount__c

blng__DebitGLAccount__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The Debit General Ledger account related to the finance transaction.
	This field is a relationship field.
	Relationship Name
	blng__DebitGLAccount__r
	Relationship Type
	Lookup
	Refers To
	blng__GLAccount__c

blng__FinanceBook__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The finance book related to the finance transaction.
	This field is a relationship field.
	Relationship Name
	blng__FinanceBook__r
	Relationship Type
	Lookup
	Refers To
	blng__FinanceBook__c

blng__FinancePeriod__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The finance period related to the finance transaction.

Field	Details
	This field is a relationship field.
	Relationship Name blng__FinancePeriod__r
	Relationship Type Lookup
	Refers To blng__FinancePeriod__c
blng__GLRule__c	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The General Ledger (GL) rule used to determine the GL accounts for the finance transaction. This field is a relationship field.
	Relationship Name blng__GLRule__r
	Relationship Type Lookup
	Refers To blng__GLRule__c
blng__GLTreatment__c	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The General Ledger (GL) treatment used to determine the GL accounts for the finance transaction. This field is a relationship field.
	Relationship Name blng__GLTreatment__r
	Relationship Type Lookup
	Refers To blng__GLTreatment__c

Billing Fields on Order

Standard and custom fields extend the standard Order object for use in Billing to represent information about orders.

Fields

Field	Details
`blng__BillNow__c`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Creates an invoice for all order products in the order.
`blng__BilledAmountwithouttax__c`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total amount that has been billed for all the order products in the order, excluding tax. This amount is equal to the sum of Billed Amount (without tax) values of all the order products. This field is a calculated field.
`blng__BilledTax__c`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total amount of taxes that have been billed for all the order products in the order. This amount is equal to the sum of all Tax Amount values of all the order products. This field is a calculated field.
`blng__BillingAccount__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** This field isn't populated by default. If a user or process populates the `Billing Account` field before invoicing the order, the invoice's `Account` field inherits this value instead of the order's `Account` field. This field is a relationship field. **Relationship Name** `blng__BillingAccount__r` **Relationship Type** Lookup

Field	Details
	Refers To Account
blng__BillingDayOfMonth__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The day of the month when the order's products are billed. Possible values are from 1 to 31.
blng__CanceledBillings__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total amount that was canceled and won't be billed for all the order products in the order. This amount is equal to the sum of all Canceled Billing Amount (without tax) values of all the order products. This field is a calculated field.
blng__InvoiceBatch__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Groups orders for invoicing during invoice run evaluation. Null values are assigned to the same invoice run.
blng__PendingBillings__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total amount that hasn't been billed for all the order products in the order. This amount is equal to the sum of all Pending Billing Amount (without tax) values of all the order products. This field is a calculated field.
blng__TotalBilling__c	**Type** currency **Properties** Filter, Nillable, Sort

Field	Details

Description
The total amount that has been billed for all the order products on the order, including tax. This amount is equal to the sum of all Billed Amount (without tax) values and all Billed Tax values of all the order products.

This field is a calculated field.

SEE ALSO:

Object Reference for the Salesforce Platform: Order

Billing Fields on Order Product

Standard and custom fields extend the standard OrderProduct object for use in Billing to represent information about order products.

Fields

Field	Details
blng__AllocatedRevenueAmount__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The revenue amount that has been allocated for the order product.
blng__Asset__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The asset for the order product that's used with the Customer Asset Lifecycle Management (CALM) service. This field is a relationship field. **Relationship Name** blng__Asset__r **Relationship Type** Lookup **Refers To** Asset

Field	Details
blng__BillThroughDateOverride__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The end date of the next invoice line created from the order product. For the upcoming billing cycle, this field overrides the standard invoice line end date calculated by using the charge to date and billing frequency. Salesforce Billing clears this field after it creates the invoice line.
blng__BillableUnitPrice__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** When you order a quote, Salesforce CPQ calculates the billable unit price for one-time and recurring order products on the order. The formula for this value is `[(Amount / Prorate Multiplier) / Pricing Frequency] * Billing Frequency` This field is a read-only field.
blng__BilledAmountwithouttax__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The sum of amounts of all the invoice lines for the order product, excluding tax.
blng__BilledTax__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The sum of taxes of all the invoice lines for the order product.
blng__BillingAccount__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description
	The name of the billing account that the order is related to. You can enter the account name or select the account.
	This field is a relationship field.
	Relationship Name
	blng__BillingAccount__r
	Relationship Type
	Lookup
	Refers To
	Account
blng__BillingGLRule__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The billing rule for including billing transactions in an external general ledger.
	This field is a relationship field.
	Relationship Name
	blng__BillingGLRule__r
	Relationship Type
	Lookup
	Refers To
	blng__GLRule__c
blng__BillingGLTreatment__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The billing treatment for including billing transactions in an external general ledger.
	This field is a relationship field.
	Relationship Name
	blng__BillingGLTreatment__r
	Relationship Type
	Lookup
	Refers To
	blng__GLTreatment__c
blng__BillingRule__c	**Type**
	reference

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The billing rule that's assigned to the order product. This field is a relationship field.
	Relationship Name blng__BillingRule__r
	Relationship Type Lookup
	Refers To blng__BillingRule__c
blng__BillingTreatment__c	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The billing treatment assigned to the order product. This field is a relationship field.
	Relationship Name blng__BillingTreatment__r
	Relationship Type Lookup
	Refers To blng__BillingTreatment__c
blng__CanceledBillings__c	**Type** currency
	Properties Create, Filter, Nillable, Sort, Update
	Description The total balance subtracted from the order product due to cancellations or swaps.
blng__FinanceBookAccounting__c	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The finance book for the order product. This field is auto-populated based on the billing treatment for the order product. This field is a relationship field.

Field	Details
	Relationship Name blng__FinanceBookAccounting__r
	Relationship Type Lookup
	Refers To blng__FinanceBook__c
blng__GroupId__c	**Type** string
	Properties Create, Filter, Group, idLookup, Nillable, Sort, Update
	Description The external ID of the invoice group.
blng__GroupQuantity__c	**Type** double
	Properties Create, Filter, Nillable, Sort, Update
	Description The quantity of order products for the invoice group.
blng__GroupUnitPrice__c	**Type** currency
	Properties Create, Filter, Nillable, Sort, Update
	Description The billable unit price for the invoice group.
blng__HoldBilling__c	**Type** picklist
	Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description Specifies whether invoice schedulers evaluate an active order product for billing. Possible values are: • No: Invoice schedulers evaluate an active order product • Yes: Invoice schedulers don't evaluate an active order product The default value is No.
blng__IncludedUsage__c	**Type** double

Field	Details

Properties

Create, Filter, Nillable, Sort, Update

Description

Salesforce Billing provides this amount of usage for free before charging. For example, if your usage order product has an included quantity of 10 and its usage for one billing period sum to 12, Salesforce Billing invoices only 2 quantities of usage from the usage summary.

The included usage is available only for usage products using price schedules.

`blng__InvoiceGroupId__c`

Type

string

Properties

Create, Filter, Group, Nillable, Sort, Update

Description

After Salesforce Billing groups your order products by their invoice group values, it assigns each group a unique ID.

This field is a read-only field.

`blng__InvoiceGrouping__c`

Type

multipicklist

Properties

Create, Filter, Nillable, Restricted picklist, Update

Description

Use invoice groups to split an account's charges into multiple invoices based on user-defined values. When an invoice scheduler picks up its order products during an invoice run, it looks at each order product's invoice group value. It then groups those order products into invoices by their respective value. You can set any number of invoice group values on an order product.

Valid values are:

- `Contract Number`
- `Invoice Group ID`
- `Legal Entity`
- `Order`
- `Order PO Number`
- `Separate Invoice`

`blng__InvoiceRunProcessingStatus__c`

Type

picklist

Properties

Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Description

The status of an order product as the invoice run evaluates it.

Valid values are:

Field	Details

- `Completed`: The invoice run created and posted an invoice line for the order product. The order product doesn't have any more invoice lines.
- `Error`: The invoice run encountered an error while trying to invoice the order product. For more information, review the invoice scheduler's error log.
- `In Progress`: The invoice run has created an invoice line but hasn't posted it yet.
- `Pending Billing`: A future invoice run creates another invoice line from the order product during its upcoming billing period.
- `Will Not Invoice`: The invoice run evaluated the order product and determined that it's not invoiceable.

`blng__InvoiceRunStartDate__c`

Type
dateTime

Properties
Create, Filter, Nillable, Sort, Update

Description
The start date and time of the invoice run that's evaluating the order product.

`blng__InvoiceRunStatusIndex__c`

Type
string

Properties
Create, Filter, Group, idLookup, Nillable, Sort, Update

Description
The stage of the invoice run that's evaluating the order product for billing. Users must not update this system field. This field is automatically updated with a text representation of the picklist value when the `Invoice Run Processing Status` field is updated.

`blng__InvoiceRun__c`

Type
reference

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The invoice run that's evaluating the order product for billing.

This field is a relationship field.

Relationship Name
blng__InvoiceRun__r

Relationship Type
Lookup

Refers To
blng__InvoiceRun__c

`blng__LastChargeToDate__c`

Type
date

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** The date the order product was billed through. This date is equal to the end date of the last invoice line for the order product. Salesforce Billing sets this field automatically and it's used in the calculation of invoice charge amounts. This field is a read-only field.
`blng__LastInvoiceTargetDate__c`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The target date of the most recent invoice run that generated an invoice line for the order product. Salesforce Billing sets this field automatically and it's used for auditing purposes. This field is a read-only field.
`blng__LegalEntityReference__c`	**Type** string **Properties** Create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The legal entity that covers the order product. When you bill the order product, Salesforce CPQ uses the legal entity to apply billing rules, tax rules, revenue recognitions rules, and their related treatments.
`blng__LegalEntity__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The legal entity that covers the order product. When you bill the order product, Salesforce CPQ uses the legal entity to apply billing rules, tax rules, revenue recognitions rules, and their related treatments. This field is a relationship field. **Relationship Name** blng__LegalEntity__r **Relationship Type** Lookup **Refers To** blng__LegalEntity__c

Field	Details
`blng__NextBillingDate__c`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The next date that Salesforce Billing evaluates and prices the order product for billing. This field is a system field that users and Salesforce admins mustn't change.
`blng__NextChargeDate__c`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The next date that Salesforce Billing will charge the order product. Salesforce Billing sets this field based on the order product's billing frequency, billing type, and date of invoice. This field is a read-only field.
`blng__OverrideBillableUnitPrice__c`	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** A new billable unit price for unactivated and uninvoiced evergreen subscription order products. This field is useful if you're migrating an evergreen subscription order product from outside Salesforce Billing. You can define an Override Billable Unit Price rather than rerun calculations in Salesforce Billing to determine the billable unit price.
`blng__OverrideInitialRevenueEndDate__c`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The end date of revenue recognition. This controls the end date of any revenue schedules created from the order product.
`blng__OverrideInitialRevenueStartDate__c`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The start date of revenue recognition. This controls the start date of any revenue schedules created from the order product.

Field	Details
blng__OverrideNextBillingDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** A one-time custom billing date that invoice schedulers use instead of the `Next Billing Date` field when evaluating order products for invoicing. Salesforce Billing erases this field's value after invoicing the order product.
blng__PendingBillings__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The total amount remaining to be billed from the order product, excluding tax.
blng__RevenueExpectedAmount__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The sum of expected revenue impact of all the credit note lines of the order product.
blng__RevenueLiabilityAmount__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The sum of revenue liability of all the credit note lines of the order product.
blng__RevenueMostLikelyAmount__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The sum of most likely revenue impact of all the credit note lines of the order product. This field is deprecated from Summer '18.
blng__RevenueRecognitionRule__c	**Type** reference

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** The revenue recognition rule that defines whether Salesforce Billing creates a revenue schedule and applies revenue recognition treatments to the order product. This field is a relationship field. **Relationship Name** blng__RevenueRecognitionRule__r **Relationship Type** Lookup **Refers To** blng__RevenueRecognitionRule__c
`blng__RevenueScheduleStatus__c`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The status of recognizing the revenue for the order product. Valid values are: • `Complete` • `Error` • `Queued`
`blng__TaxCity__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The city of the address that Salesforce Billing uses to calculate tax for the order product.
`blng__TaxCountry__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The country of the address that Salesforce Billing uses to calculate tax for the order product.
`blng__TaxCounty__c`	**Type** string

Field	Details
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The county of the address that Salesforce Billing uses to calculate tax for the order product.
`blng__TaxErrorMessage__c`	**Type**
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	A message indicating that Salesforce Billing couldn't calculate tax for the order product.
`blng__TaxGLRule__c`	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The tax rule for recording tax transactions in an external general ledger.
	This field is a relationship field.
	Relationship Name
	blng__TaxGLRule__r
	Relationship Type
	Lookup
	Refers To
	blng__GLRule__c
`blng__TaxGLTreatment__c`	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The tax treatment for recording tax transactions in an external general ledger.
	This field is a relationship field.
	Relationship Name
	blng__TaxGLTreatment__r
	Relationship Type
	Lookup
	Refers To
	blng__GLTreatment__c

Field	Details
blng__TaxPercentageApplied__c	**Type** percent **Properties** Create, Filter, Nillable, Sort, Update **Description** The amount of tax that Salesforce CPQ applied to the order product based on your tax engine's tax calculations.
blng__TaxRule__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The tax rule that's assigned to the order product. Tax rules and their treatments define how Salesforce Billing processes tax for your transactions. This field is a relationship field. **Relationship Name** blng__TaxRule__r **Relationship Type** Lookup **Refers To** blng__TaxRule__c
blng__TaxState__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The state of the address that Salesforce Billing uses to calculate tax for the order product. This field overrides the state inherited from the account.
blng__TaxStatus__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The tax calculation status for the order product. Valid values are: • Completed • Copied • Error

24

Field	Details
	• Processing
	• Queued
	• Warning

Field	Details
`blng__TaxStreet2__c`	**Type** string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The street of the address that Salesforce Billing uses to calculate tax for the order product. This field overrides the street inherited from the account.
`blng__TaxStreet__c`	**Type** string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The street of the address that Salesforce Billing uses to calculate tax for the order product. This field overrides the street inherited from the account.
`blng__TaxTreatment__c`	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The tax treatment assigned to the order product. Tax rules and their treatments define how Salesforce Billing processes tax for your transactions. This field is a relationship field.
	Relationship Name blng__TaxTreatment__r
	Relationship Type Lookup
	Refers To blng__TaxTreatment__c
`blng__TaxZipCode__c`	**Type** string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The postal code of the address that Salesforce Billing uses to calculate tax for the order product. This field overrides the postal code inherited from the account.

Field	Details
blng__TerminatedBillings__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The amount available to be canceled from the order product, which can't be more than the pending billing amount.
blng__TotalBilling__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of all invoice lines, including tax, for the order product. The formula for calculating this amount is `blng__BilledAmountwithouttax__c + blng__BilledTax__c` .
blng__UniqueId__c	**Type** string **Properties** Create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The unique ID of the order product.
blng__UsageFloorQuantity__c	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The minimum usage quantity that Salesforce Billing charges for, even if there's little or no usage. For example, if your order product has a usage floor quantity of 10, and the usages sum to 7, Salesforce Billing invoices for a usage quantity of 10. The included usage is available only for usage products using price schedules.

SEE ALSO:

Object Reference for the Salesforce Platform: OrderItem

Billing Fields on Product2

Standard and custom fields extend the standard Product2 object for use in Billing to represent information about products.

Fields

Field	Details
blng__BillingRule__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** This rule is used to invoice usage summaries related to the product. This field is a relationship field. **Relationship Name** blng__BillingRule__r **Relationship Type** Lookup **Refers To** blng__BillingRule__c
blng__RevenueRecognitionRule__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** This rule is used to recognize the revenue of usage summaries based on the summary's related revenue recognition rule. This field is a relationship field. **Relationship Name** blng__RevenueRecognitionRule__r **Relationship Type** Lookup **Refers To** blng__RevenueRecognitionRule__c
blng__TaxRule__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description
	This rule is used to tax usage summary invoice lines based on the summary's related tax rule.
	This field is a relationship field.
	Relationship Name
	blng__TaxRule__r
	Relationship Type
	Lookup
	Refers To
	blng__TaxRule__c

SEE ALSO:

Object Reference for the Salesforce Platform: Product2

CHAPTER 2   Salesforce Billing Custom Objects

The Salesforce Billing data model provides custom objects to manage invoices, payments, and revenue.

In this chapter ...

- blng__PaymentGateway__c
- blng__PaymentGatewayConfig__c
- blng__PaymentMapping__c
- blng__PaymentMethod__c
- blng__PaymentRun__c
- blng__PaymentScheduler__c
- blng__PaymentTerms__c
- blng__PaymentTransaction__c
- blng__Refund__c
- blng__RefundLineCreditNoteLine__c
- blng__RefundLinePayment__c
- blng__RevenueAdjustment__c
- blng__RevenueAdjustmentLine__c
- blng__RevenueAgreement__c
- blng__RevenueDistributionMethod__c
- blng__RevenueRecognitionRule__c
- blng__RevenueRecognitionTreatment__c
- blng__RevenueSchedule__c
- blng__RevenueTransaction__c
- blng__SubInvoiceLine__c
- blng__TaxConfig__c
- blng__TaxIntegration__c
- blng__TaxRate__c
- blng__TaxRule__c
- blng__TaxTreatment__c
- blng__Usage__c
- blng__UsageSummary__c

blng__AccountBalanceSnapshot__c

Represents a list of outstanding invoice debt within an account, grouped by the number of days past the due date. The account balance snapshot also includes the amounts of unapplied credit notes and payments within the same account.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the legacy entity when multiple currencies are enabled.
LastActivityDate	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The timestamp of when a user had last associated a task or event action with the account.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed an accounting balance snapshot indirectly, for example, through a list view or related record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed an accounting balance snapshot record. If this value is null, it's possible that the user only accessed the accounting period record or

Field	Details
	a related list view (`LastReferencedDate`), but not viewed the accounting balance snapshot record itself.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the account balance snapshot.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. The ID of the record owner. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
blng__AccountBalance__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The number of unpaid balances across all the account's invoices. This field is a calculated field.
blng__Account__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The account for which the invoices are calculated. This field is a relationship field. **Relationship Name** blng__Account__r **Refers To** Account

Field	Details
blng__BalanceSnapshotRunCreatedBy__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The balance snapshot run that created the account balance snapshot record. This field is a relationship field. **Relationship Name** blng__BalanceSnapshotRunCreatedBy__r **Refers To** blng__BalanceSnapShotRun__c
blng__InvoiceAgingBucket10__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** An aging bucket represents a range of dates beyond an invoice's due date. For instance, one bucket might cover invoices that are 1–30 days overdue, while another could cover those that are 31–60 days overdue. Users can set up to 10 aging buckets for each org. When a balance snapshot is generated, the overdue amounts are categorized into these aging buckets based on how long they've been past due. For example, if your account had three invoices that were 45 days overdue, their total debt would be grouped and displayed in the 31 to 60-day aging bucket.
blng__InvoiceAgingBucket1__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** An aging bucket represents a range of dates beyond an invoice's due date.
blng__InvoiceAgingBucket2__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** An aging bucket represents a range of dates beyond an invoice's due date.
blng__InvoiceAgingBucket3__c	**Type** currency

Field	Details
	Properties Create, Filter, Nillable, Sort, Update
	Description An aging bucket represents a range of dates beyond an invoice's due date.
blng__InvoiceAgingBucket4__c	**Type** currency
	Properties Create, Filter, Nillable, Sort, Update
	Description An aging bucket represents a range of dates beyond an invoice's due date.
blng__InvoiceAgingBucket5__c	**Type** currency
	Properties Create, Filter, Nillable, Sort, Update
	Description An aging bucket represents a range of dates beyond an invoice's due date.
blng__InvoiceAgingBucket6__c	**Type** currency
	Properties Create, Filter, Nillable, Sort, Update
	Description An aging bucket represents a range of dates beyond an invoice's due date.
blng__InvoiceAgingBucket7__c	**Type** currency
	Properties Create, Filter, Nillable, Sort, Update
	Description An aging bucket represents a range of dates beyond an invoice's due date.
blng__InvoiceAgingBucket8__c	**Type** currency
	Properties Create, Filter, Nillable, Sort, Update
	Description An aging bucket represents a range of dates beyond an invoice's due date.
blng__InvoiceAgingBucket9__c	**Type** currency

Field	Details
	Properties Create, Filter, Nillable, Sort, Update
	Description An aging bucket represents a range of dates beyond an invoice's due date.
blng__Invoices__c	**Type** textarea
	Properties Create, Nillable, Update
	Description A list of invoices assessed by the balance snapshot run to generate the account balance snapshot, displayed by the record name.
blng__Notes__c	**Type** textarea
	Properties Create, Nillable, Update
	Description Additional information or comments related to the account balance snapshot record.
blng__SnapshotDate__c	**Type** date
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The day that the balance snapshot run created the account balance snapshot record.
blng__TotalInvoiceAmount__c	**Type** currency
	Properties Filter, Nillable, Sort
	Description The sum of all invoice balances on the account, including debit notes. This field is different from the total invoice amount that's described in your contractual terms for Salesforce Billing. This field is a calculated field.
blng__UnappliedCreditNoteAmount__c	**Type** currency
	Properties Create, Filter, Nillable, Sort, Update

Field	Details
	Description
	The sum of all unapplied credit note amounts on the account.
blng__UnappliedCreditNotes__c	**Type**
	textarea
	Properties
	Create, Nillable, Update
	Description
	A list of the unapplied credit notes that the balance snapshot run evaluated, shown by the record name.
blng__UnappliedDebitNoteAmount__c	**Type**
	currency
	Properties
	Create, Filter, Nillable, Sort, Update
	Description
	The sum of all unapplied debit note amounts on this account.
blng__UnappliedDebitNotes__c	**Type**
	textarea
	Properties
	Create, Nillable, Update
	Description
	A list of unapplied debit notes evaluated by the balance snapshot, shown by the record name.
blng__UnappliedPaymentAmount__c	**Type**
	currency
	Properties
	Create, Filter, Nillable, Sort, Update
	Description
	The sum of all unapplied payments in this account.
blng__UnappliedPayments__c	**Type**
	textarea
	Properties
	Create, Nillable, Update
	Description
	A list of the unapplied payments that the balance snapshot run evaluated, shown by the record name.

Field	Details
blng__UniqueId__c	**Type** string **Properties** Create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The unique identification number associated with the account balance snapshot record. This field is unique within your organization.

Associated Objects

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

blng__AccountBalanceSnapshot__ChangeEvent
Change events are available for the object.

blng__AgingBuckets__c

Represents the outstanding invoices sorted into various aging periods.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the aging buckets when multiple currencies are enabled.
LastActivityDate	**Type** date **Properties** Filter, Group, Nillable, Sort

Field	Details

Description
The timestamp of when a user had last associated a task or event action with the aging buckets.

LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed an aging bucket record indirectly, for example, through a list view or related record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed an aging bucket record. If this value is null, it's possible that the user only accessed the aging bucket record or a related list view (LastReferencedDate), but not viewed the aging bucket record itself.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the aging bucket.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. The ID of the record owner. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
blng__Active__c	**Type** boolean

Field	Details
	Properties Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. Indicates whether the aging bucket is active (`true`) or not (`false`). To deactivate the GL treatment, clear the `Active` checkbox. This setting won't affect past or existing transactions or assignments. The default value is `false`.
blng__BucketNumber__c	**Type** double **Properties** Create, Filter, Sort, Update **Description** Required. Classifies the records based on their age.
blng__EndingDaysfromPastDue__c	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The number of days from the past due date to the end of an aging bucket.
blng__Notes__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Additional details or comments related to the aging buckets.
blng__StartingDaysfromPastDue__c	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The starting point for calculating the number of days an invoice is past due.

blng__BalanceSnapShotRun__c

Represents the analysis of an account's invoices on a scheduled basis for past-due balances.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the balance snapshot run when multiple currencies are enabled.
LastActivityDate	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The timestamp of when a user had last associated a task or event action with the balance snapshot run.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the balance snapshot run.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. The ID of the record owner. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User

Field	Details
blng__ApexJobId__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** A unique identifier of an Apex job associated with the balance snapshot run.
blng__BalanceSnapshotScheduler__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Specifies the way to schedule and manage balance snapshots. This field is a relationship field. **Relationship Name** blng__BalanceSnapshotScheduler__r **Refers To** blng__BalanceSnapShotScheduler__c
blng__Notes__c	**Type** textarea **Properties** Create, Nillable, Update **Description** Additional details or comments related to the balance snapshot run.
blng__Status__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The status of the balance snapshot run. Valid values are: • Completed • Credit Note Completed • Credit Note Processing • Debit Note Completed • Debit Note Processing • Error • Invoice Completed

Field	Details
	• Invoice Processing
	• Payment Completed
	• Payment Processing
	• Started

blng__BalanceSnapShotScheduler__c

Represents the Balance Snapshot Scheduler, which creates a Balance Snapshot Run to regularly analyze your account's invoices for past-due balances.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the balance snapshot scheduler record when multiple currencies are enabled.
LastActivityDate	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The timestamp of when a user had last associated a task or event action with the balance snapshot scheduler.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort

Field	Details
	Description
	The timestamp when the current user last accessed a balance snapshot scheduler record indirectly, for example, through a list view or related record.
LastViewedDate	**Type**
	dateTime
	Properties
	Filter, Nillable, Sort
	Description
	The timestamp when the current user last viewed a balance snapshot scheduler record. If this value is null, it's possible that the user only accessed the balance snapshot scheduler record or a related list view (`LastReferencedDate`), but not viewed the balance snapshot scheduler record itself.
Name	**Type**
	string
	Properties
	Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update
	Description
	The name of the balance snapshot scheduler.
OwnerId	**Type**
	reference
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	Required. The ID of the record owner.
	This field is a polymorphic relationship field.
	Relationship Name
	Owner
	Refers To
	Group, User
blng__ApexJobId__c	**Type**
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	A unique identifier of an Apex job associated with the balance snapshot scheduler.
blng__Notes__c	**Type**
	textarea

Field	Details
	Properties Create, Nillable, Update
	Description Additional details or comments related to the balance snapshot scheduler.
`blng__StartDateTime__c`	**Type** dateTime
	Properties Create, Filter, Nillable, Sort, Update
	Description The date and time when the scheduler starts the run.
`blng__Type__c`	**Type** picklist
	Properties Create, Filter, Group, Restricted picklist, Sort, Update
	Description Required. The frequency of the balance snapshot schedule. Valid values are: • `Daily` • `Monthly` • `Once` • `Weekly`

blng__BillingBatchJob__c

Represents a scheduled processing job that triggers recurring invoice batch runs and payment batch runs.

Supported Calls

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

Fields

Field	Details
`CurrencyIsoCode`	**Type** picklist

Field	Details
	Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the billing batch job when multiple currencies are enabled.
LastActivityDate	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The timestamp of when a user had last associated a task or event action with the billing batch job.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the billing batch job.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. The ID of the record owner. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
blng__ApexJobId__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** A unique identifier of an Apex job associated with the billing batch job.

Field	Details
blng__CompleteTime__c	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The timestamp when the batch job was completed.
blng__StartTime__c	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The time when the scheduler triggers the batch processing job.
blng__Status__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The status of the scheduler. Only active schedulers are capable of triggering batch processing jobs. Valid values are: • `Completed` • `Completed with errors` • `Processing` • `Started` • `Step 1/2 Completed` • `Step 1/2 Processing` • `Step 2/2 Completed` • `Step 2/2 Processing`
blng__Type__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The type of billing batch job. Valid values are: • `Auto Renewals` • `MRR Calculation`

Field	Details
	• Usage

blng__BillingPrefix__c

Represents a prefix that can be appended to billing-related records for storage purposes.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the billing prefix record when multiple currencies are enabled.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the billing prefix.
SetupOwnerId	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the record owner. This field is a polymorphic relationship field. **Relationship Name** SetupOwner

Field	Details
	Refers To Organization, Profile, User
blng__BarPrefix__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** A prefix value of barcodes.
blng__DotPrefix__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** A prefix value of dots.
blng__RawPrefix__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** A raw prefix value.

blng__BillingRule__c

Represents how and when billing must occur for products and services.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** how and when billing should occur for products and services. picklist

48

Field	Details
	Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the billing rule record when multiple currencies are enabled.
LastActivityDate	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The timestamp of when a user had last associated a task or event action with the billing rule.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a billing rule record indirectly, for example, through a list view or related record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a billing rule record. If this value is null, it's possible that the user only accessed the billing rule record or a related list view (`LastReferencedDate`), but not viewed the billing rule record itself.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the billing rule.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update

Field	Details
	Description
	Required. The ID of the record owner.
	This field is a polymorphic relationship field.
	Relationship Name
	Owner
	Refers To
	Group, User

blng__Active__c	**Type**
	boolean
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	Required. Indicates whether the billing rule is active (`true`) or not (`false`). To deactivate the billing rule, clear the `Active` checkbox. This setting won't affect past or existing transactions or assignments. The default value is `true`.

blng__AmendmentBillCycleDateAlignment__c	**Type**
	picklist
	Properties
	Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	Specifies how the bill cycle date must be aligned when an amendment is made.
	Valid values are:
	• `Align to original Order Product`
	• `Do not align amended Order Product`
	The default value is `Do not align amended Order Product`.

blng__GenerateInvoices__c	**Type**
	picklist
	Properties
	Create, Filter, Group, Restricted picklist, Sort, Update
	Description
	Required. Specifies if the billing rule creates invoices.
	Valid values are:
	• `No`
	• `Yes`

blng__InitialBillingTrigger__c	**Type**
	picklist

Field	Details
	Properties Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. Specifies the occurrence whose date is used to calculate the initial billing trigger date. Valid value is: • `Order Product Activation Date`
`blng__Notes__c`	**Type** textarea **Properties** Create, Nillable, Update **Description** Additional details or comments related to the billing rule.
`blng__PartialPeriodTreatment__c`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. The process for generating bills for recurring order products that begin billing on a different day than their start date. Valid values are: • `Combine` • `Separate` The default value is `Separate`.
`blng__PeriodTreatmentForBTDO__c`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** This field is relevant only if an order product has a Bill Through Date Override (BTDO) value within a billing cycle. This field specifies the option to merge a partial billing period with the previous full period when generating an invoice line, or to keep the partial period as a separate invoice line. Valid values are: • `Combine` • `Exclude Billing Day of Month` • `Separate`

Field	Details
	The default value is `Exclude Billing Day of Month`.

blng__BillingSchedule__c

Represents information about the order product that's invoiced based on the invoice plan billing frequency.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the billing schedule's order product when multiple currencies are enabled. The default value is `USD`.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the billing schedule.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the user who created the billing schedule. This field is a polymorphic relationship field. **Relationship Name** Owner

Field	Details
	Relationship Type Lookup
	Refers To Group, User
`blng__Amount__c`	**Type** currency
	Properties Create, Filter, Nillable, Sort, Update
	Description The total amount that has been invoiced for the order product, excluding tax.
`blng__BilledAmount__c`	**Type** currency
	Properties Filter, Nillable, Sort
	Description The total amount that has been billed for the order product related to the billing schedule, based on the order product's invoice plan lines. This field is a calculated field.
`blng__InvoicePlan__c`	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The invoice plan associated with the billing schedule. This field is a relationship field.
	Relationship Name blng__InvoicePlan__r
	Relationship Type Lookup
	Refers To blng__InvoicePlan__c
`blng__NumberOfBilledBillingTransactions__c`	**Type** double
	Properties Filter, Nillable, Sort

Field	Details
	Description
	The number of billing transactions that are associated with the billing schedule with the status as `Billed`.
	This field is a calculated field.
blng__NumberOfBillingTransactions__c	**Type**
	double
	Properties
	Filter, Nillable, Sort
	Description
	The number of billing transactions that are associated with the billing schedule.
	This field is a calculated field.
blng__NumberOfCanceledBillingTransactions__c	**Type**
	double
	Properties
	Filter, Nillable, Sort
	Description
	The number of billing transactions associated with the billing schedule with the status as `Canceled`.
	This field is a calculated field.
blng__OrderProduct__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The order product associated with the billing schedule.
	This field is a relationship field.
	Relationship Name
	blng__OrderProduct__r
	Relationship Type
	Lookup
	Refers To
	OrderItem
blng__Source__c	**Type**
	picklist
	Properties
	Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details
	Description The source of the billing schedule. Valid values are: • `External` • `Standard`
`blng__Status__c`	**Type** string **Properties** Filter, Nillable, Sort **Description** The status of the billing schedule. The formula for calculating the status is ```\nIF(blng__BilledAmount__c == blng__Amount__c, "Billed",\n IF((blng__NumberOfBillingTransactions__c != 0 &&\nblng__NumberOfBillingTransactions__c =\nblng__NumberOfCanceledBillingTransactions__c), "Cancelled",\n"Pending")\n)\n``` . This field is a calculated field.
`blng__TaxCity__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The city of the address that Salesforce Billing uses to calculate tax for the billing schedule's order product.
`blng__TaxCountry__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The country of the address that Salesforce Billing uses to calculate tax for the billing schedule's order product.
`blng__TaxState__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description
	The state of the address that Salesforce Billing uses to calculate tax for the billing schedule's order product.
blng__TaxStreet__c	**Type**
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The street of the address that Salesforce Billing uses to calculate tax for the billing schedule's order product.
blng__TaxZipCode__c	**Type**
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The postal code of the address that Salesforce Billing uses to calculate tax for the billing schedule's order product.

blng__BillingTransaction__c

Represents information about each invoice plan line of a billing schedule. A billing transaction stores the billed amount based on the relationship between the invoice plan line's commencement date and the order product's service start and end dates. When a billing transaction is invoiced, Salesforce Billing copies data from the billing transaction to the invoice line.

🛑 **Important:** Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Supported Calls

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

Fields

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

Field	Details
	Description
	The three-letter ISO 4217 currency code associated with the billing transaction's schedule when multiple currencies are enabled.
	The default value is USD.
Name	**Type**
	string
	Properties
	Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update
	Description
	The name of the billing transaction.
blng__AmountToBeBilled__c	**Type**
	currency
	Properties
	Filter, Nillable, Sort
	Description
	The amount to be billed for the order product related to the billing transaction's billing schedule. The formula for calculating this amount when its blank is

```
IF( ISBLANK( blng__OverrideAmount__c ) , blng__Amount__c ,
blng__OverrideAmount__c )
```

.

This field is a calculated field.

blng__Amount__c	**Type**
	currency
	Properties
	Create, Filter, Nillable, Sort, Update
	Description
	The amount that has been invoiced for the order product by the billing transaction.
blng__BillingSchedule__c	**Type**
	reference
	Properties
	Create, Filter, Group, Sort
	Description
	The billing schedule related to the billing transaction.
	This field is a relationship field.
	Relationship Name
	blng__BillingSchedule__r

Field	Details
	Relationship Type Master-detail
	Refers To blng__BillingSchedule__c
blng__BillingTargetDate__c	**Type** date
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The date when the invoice line is created for the order product.
blng__CalculatedStatus__c	**Type** string
	Properties Filter, Nillable, Sort
	Description The status of the billing transaction by using the formula

```
IF(ISPICKVAL(blng__OverrideStatus__c, "--None--"),
TEXT(blng__Status__c), TEXT(blng__OverrideStatus__c))
```

.

This field is a calculated field.

Field	Details
blng__InvoicePlanLine__c	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The invoice plan associated with the billing transaction. This field is a relationship field.
	Relationship Name blng__InvoicePlanLine__r
	Relationship Type Lookup
	Refers To blng__InvoicePlanLine__c
blng__OverrideAmount__c	**Type** currency

Field	Details

	Properties
	Create, Filter, Nillable, Sort, Update
	Description

blng__OverrideBillingTargetDate__c	**Type**
	date
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The date that overrides the billing target date of the billing transaction.

blng__OverrideStatus__c	**Type**
	picklist
	Properties
	Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	The value that overrides the Status value of the billing transaction.
	Valid value is Cancelled.

blng__Quantity__c	**Type**
	double
	Properties
	Create, Filter, Nillable, Sort, Update
	Description
	The quantity of the order product that the billing transaction invoices.

blng__ServicePeriodEnd__c	**Type**
	date
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The start date of the service period of the order product associated with the billing schedule.

blng__ServicePeriodStart__c	**Type**
	date
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The end date of the service period of the order product associated with the billing schedule.

Field	Details
blng__Source__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The source of the billing transaction. Valid values are: - `External` - `Standard`
blng__Status__c	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The status of the billing transaction. Valid values are: - `Billed` - `Cancelled` - `Pending`

blng__BillingTreatment__c

Represents information about how Salesforce Billing records billing rule actions in your finance books and general ledgers. Use billing treatments to specify an invoice plan and cancellation invoice plan for dynamic invoice plans. You can also set up the cancellation rule for amendment orders to override the package setting for the default cancel order rule.

🛇 **Important:** Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the billing treatment when multiple currencies are enabled. The default value is USD.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp of when the current user had last accessed the billing treatment, or a record or list view related to the billing treatment.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed the billing treatment, a record related to this record, or a record or list view related to the billing treatment. If this value is null, it's possible the user accessed this record or list view (LastReferencedDate) but didn't view it.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the billing treatment.
blng__Active__c	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the billing treatment is active (true) or not (false).

Field	Details
	The default value is `true`.
blng__BillingFinanceBook__c	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The finance book that has finance periods to all the invoice lines that the billing rule creates from the order product. This field is a relationship field. **Relationship Name** blng__BillingFinanceBook__r **Relationship Type** Lookup **Refers To** blng__FinanceBook__c
blng__BillingGLRule__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The General Ledger (GL) rule used to determine the GL accounts for the finance transaction. This field is a relationship field. **Relationship Name** blng__BillingGLRule__r **Relationship Type** Lookup **Refers To** blng__GLRule__c
blng__BillingLegalEntity__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The legal entity related to the billing treatment. When your billing rule evaluates an order product, Salesforce Billing applies one of the rule's treatments if the order product and treatment have matching legal entities. This field is a relationship field.

Field	Details

Relationship Name
blng__BillingLegalEntity__r

Relationship Type
Lookup

Refers To
blng__LegalEntity__c

---

blng__BillingRule__c

Type
reference

Properties
Create, Filter, Group, Sort

Description
The billing rule related to the billing treatment.

This field is a relationship field.

Relationship Name
blng__BillingRule__r

Relationship Type
Master-detail

Refers To
blng__BillingRule__c (the master object)

---

blng__CancellationInvoicePlan__c

Type
reference

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The cancellation invoice plan related to the billing treatment.

This field is a relationship field.

Relationship Name
blng__CancellationInvoicePlan__r

Relationship Type
Lookup

Refers To
blng__InvoicePlan__c

---

blng__CancellationRule__c

Type
picklist

Properties
Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update

Field	Details
	Description The cancellation rule for amendment orders. This rule overrides the package setting for the default cancel order rule. Valid values are: • `LIFO by Order Product Creation Date` • `Legacy` The default value is `Legacy`.
`blng__InvoicePlan__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The invoice plan associated with the billing treatment. This field is a relationship field. **Relationship Name** blng__InvoicePlan__r **Relationship Type** Lookup **Refers To** blng__InvoicePlan__c
`blng__Notes__c`	**Type** textarea **Properties** Create, Nillable, Update **Description** Additional details about the billing transaction.

blng__CreditNote__c

Represents information about the amount that's used to reduce the amount that a buyer owes a seller under the terms of an earlier invoice. A credit note always decreases the balance of an invoice. Users can apply positive credit notes to positive invoices, for example, a $10 credit note reduces the balance of a $100 invoice line to $90.

🛑 **Important:** Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the credit note. The default value is USD.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp of when the current user had last accessed the credit note, or a record or list view related to the credit note.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed the credit note, or a record or list view related to the credit note. If this value is null, it's possible the user accessed this record or list view (LastReferencedDate) but didn't view it.
Name	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the credit note.
blng__Account__c	**Type** reference

65

Field	Details
	Properties Create, Filter, Group, Sort
	Description The customer account associated with the credit note. This field is a relationship field.
	Relationship Name blng__Account__r
	Relationship Type Master-detail
	Refers To Account
blng__Allocations__c	**Type** currency
	Properties Filter, Nillable, Sort
	Description The credit note's balance that's allocated to invoice lines. This field is a calculated field.
blng__Balance__c	**Type** currency
	Properties Filter, Nillable, Sort
	Description Amount of the credit note that's available for allocation or refund. The balance amount is calculated using the formula ``` blng__TotalAmount__c - blng__Allocations__c + blng__Unallocations__c - blng__Debits__c - blng__Refunds__c ``` . This field is a calculated field.
blng__BaseCurrencyAmount__c	**Type** currency
	Properties Create, Filter, Nillable, Sort, Update
	Description The credit note's total amount in the base currency.

Field	Details
blng__BaseCurrencyFXDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date used to determine the foreign exchange rate when calculating the base currency amount.
blng__BaseCurrencyFXRate__c	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The foreign exchange rate that's used to convert the credit note's total amount to the base currency amount.
blng__BaseCurrency__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The base currency that's used for converting currency during data export.
blng__BillToAccount__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The account that receives information about the credit note. This field is a relationship field. **Relationship Name** blng__BillToAccount__r **Relationship Type** Lookup **Refers To** Account
blng__BillToContact__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description
	The user who receives information about the credit note. This value is inherited from the account's `PersonContactId` field.
	This field is a relationship field.
	Relationship Name
	blng__BillToContact__r
	Relationship Type
	Lookup
	Refers To
	Contact
`blng__BillingFinanceBook__c`	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The finance book associated with the credit note.
	This field is a relationship field.
	Relationship Name
	blng__BillingFinanceBook__r
	Relationship Type
	Lookup
	Refers To
	blng__FinanceBook__c
`blng__BillingFinancePeriod__c`	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The finance period associated with the credit note.
	This field is a relationship field.
	Relationship Name
	blng__BillingFinancePeriod__r
	Relationship Type
	Lookup
	Refers To
	blng__FinancePeriod__c
`blng__CreditNoteDate__c`	**Type**
	date

Field	Details

Properties

Create, Filter, Group, Sort, Update

Description

The date when the credit note was posted.

`blng__CreditNoteSourceAction__c`

Type

picklist

Properties

Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Description

The action that created the credit note.

Valid values are:

- `Cancel & Rebill`
- `Credit`
- `Invoice Line Credit`
- `Manual`
- `Negative Lines Conversion`

The default value is `Manual`.

`blng__Debits__c`

Type

currency

Properties

Filter, Nillable, Sort

Description

The sum of allocations of the debit notes associated with the credit note.

This field is a calculated field.

`blng__EffectiveTaxDate__c`

Type

date

Properties

Create, Filter, Group, Nillable, Sort, Update

Description

The date when the tax is applied to the credit note.

`blng__EstimateTaxProcessTime__c`

Type

double

Properties

Create, Filter, Nillable, Sort, Update

Description

The estimated time for processing tax for the credit note.

Field	Details
blng__ImpactAmount__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The credit note's financial impact against the customer's accounts receivable. When the credit note is applied, the impact amount is the negative equivalent of the credit note's total amount. This field is a calculated field.
blng__NetAllocations__c	**Type** double **Properties** Filter, Nillable, Sort **Description** The difference between the allocated and unallocated amounts of the credit note. This field is a calculated field.
blng__Notes__c	**Type** textarea **Properties** Create, Nillable, Update **Description** Additional details about the credit note.
blng__Refunds__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of refund amount of all the related credit note lines. This field is a calculated field.
blng__RelatedInvoice__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The invoice that this credit note was applied to. This field is a relationship field.

Field	Details
	Relationship Name blng__RelatedInvoice__r
	Relationship Type Lookup
	Refers To blng__Invoice__c
`blng__Status__c`	**Type** picklist
	Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description The status of the credit note. Valid values are: • `Cancelled` • `Draft` • `Posted` The default value is `Draft`.
`blng__Subtotal__c`	**Type** currency
	Properties Filter, Nillable, Sort
	Description The sum of subtotals of all the related credit note lines. This field is a calculated field.
`blng__TaxAmount__c`	**Type** currency
	Properties Filter, Nillable, Sort
	Description The amount of tax for the credit note. This field is a calculated field.
`blng__TaxCity__c`	**Type** string
	Properties Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The city of the address that Salesforce Billing uses to calculate tax for the credit note.
blng__TaxCountry__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The country of the address that Salesforce Billing uses to calculate tax for the credit note.
blng__TaxErrorMessage__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The error message when the external tax engine fails to calculate tax for the credit note.
blng__TaxState__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The state of the address that Salesforce Billing uses to calculate tax for the credit note.
blng__TaxStreet2__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The street of the address that Salesforce Billing uses to calculate tax for the credit note.
blng__TaxStreet__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The street of the address that Salesforce Billing uses to calculate tax for the credit note.
blng__TaxZipCode__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description
	The postal code of the address that Salesforce Billing uses to calculate tax for the credit note.
`blng__TotalAmount__c`	**Type**
	currency
	Properties
	Filter, Nillable, Sort
	Description
	The sum of amounts of the credit note's lines and adjustment lines.
	This field is a calculated field.
`blng__Unallocations__c`	**Type**
	currency
	Properties
	Filter, Nillable, Sort
	Description
	The amount that's been unallocated from the credit note.
	This field is a calculated field.

blng__CreditNoteAllocation__c

Represents allocation information about a credit note line that's been allocated to an invoice line.

Supported Calls

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

Fields

Field	Details
`CurrencyIsoCode`	**Type**
	picklist
	Properties
	Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	The three-letter ISO 4217 currency code associated with the allocated credit note.
	The default value is `USD`.

Field	Details
Name	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the credit note allocation.
blng__Amount__c	**Type** currency **Properties** Create, Filter, Sort, Update **Description** The amount that's allocated from the credit note line.
blng__BillingFinanceBook__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The finance book associated with the allocated credit note line. This field is a relationship field. **Relationship Name** blng__BillingFinanceBook__r **Relationship Type** Lookup **Refers To** blng__FinanceBook__c
blng__BillingFinancePeriod__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The finance period associated with the allocated credit note line. This field is a relationship field. **Relationship Name** blng__BillingFinancePeriod__r **Relationship Type** Lookup **Refers To** blng__FinancePeriod__c

Field	Details
blng__BillingGLRule__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The General Ledger (GL) rule that's used to determine the GL accounts for the invoice line to which the credit note line is allocated. This field is a relationship field. **Relationship Name** blng__BillingGLRule__r **Relationship Type** Lookup **Refers To** blng__GLRule__c
blng__BillingGLTreatment__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The billing treatment associated with the billing General Ledger (GL) rule of the allocated credit note line. This field is a relationship field. **Relationship Name** blng__BillingGLTreatment__r **Relationship Type** Lookup **Refers To** blng__GLTreatment__c
blng__BillingRule__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The billing rule associated with the allocated credit note line. This field is a relationship field. **Relationship Name** blng__BillingRule__r **Relationship Type** Lookup

Field	Details
	Refers To blng__BillingRule__c
`blng__BillingTreatment__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The billing treatment associated with the allocated credit note line. This field is a relationship field. **Relationship Name** blng__BillingTreatment__r **Relationship Type** Lookup **Refers To** blng__BillingTreatment__c
`blng__CreditNoteLine__c`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The credit note line that's allocated. This field is a relationship field. **Relationship Name** blng__CreditNoteLine__r **Relationship Type** Master-detail **Refers To** blng__CreditNoteLine__c (the detail object)
`blng__ImpactAmount__c`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The allocated credit note line's financial impact against the customer's accounts receivable. When the credit note line is applied, the impact amount is the negative equivalent of the credit note line's amount. This field is a calculated field.

Field	Details
blng__InvoiceLine__c	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The invoice line to which the credit note line is allocated. This field is a relationship field. **Relationship Name** blng__InvoiceLine__r **Relationship Type** Master-detail **Refers To** blng__InvoiceLine__c
blng__LegalEntity__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The legal entity associated with the allocated credit note line. This field is a relationship field. **Relationship Name** blng__LegalEntity__r **Relationship Type** Lookup **Refers To** blng__LegalEntity__c
blng__Notes__c	**Type** textarea **Properties** Create, Nillable, Update **Description** Additional details about the credit note line's allocation.
blng__TaxGLRule__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description
	The tax rule for recording tax transactions of the allocated credit note line in an external general ledger.
	This field is a relationship field.
	Relationship Name
	blng__TaxGLRule__r
	Relationship Type
	Lookup
	Refers To
	blng__GLRule__c
blng__TaxGLTreatment__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The tax treatment for recording tax transactions of the allocated credit note line in an external general ledger.
	This field is a relationship field.
	Relationship Name
	blng__TaxGLTreatment__r
	Relationship Type
	Lookup
	Refers To
	blng__GLTreatment__c
blng__TaxRule__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The rule used to tax the allocated credit note line.
	This field is a relationship field.
	Relationship Name
	blng__TaxRule__r
	Relationship Type
	Lookup
	Refers To
	blng__TaxRule__c

Field	Details
blng__TaxTreatment__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The tax treatment assigned to the allocated credit note line. Tax rules and their treatments define how Salesforce Billing processes tax for your transactions. This field is a relationship field. **Relationship Name** blng__TaxTreatment__r **Relationship Type** Lookup **Refers To** blng__TaxTreatment__c
blng__Type__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Specifies the type of the credit note allocation. Possible values are: • Allocation • Unallocation
blng__Unallocated__c	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** This field indicates whether the credit note allocation of type Allocation has a corresponding credit note allocation of type Unallocation. The default value is false.

blng__CreditNoteLine__c

Represents product, service, adjustment, or tax line items that were included in a credit note.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the credit note line. The default value is USD.
Name	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the credit note line.
blng__AllocatedRevenueAmount__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The credit note's revenue amount that's allocated to invoice lines.
blng__Allocations__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The credit note line's amount that's allocated to invoice lines, excluding tax. This field is a calculated field.
blng__Balance__c	**Type** currency **Properties** Filter, Nillable, Sort

Field	Details

Description

Amount of the credit note line that's available for allocation or refund. The balance is calculated using the formula

```
blng__TotalAmount__c - blng__Allocations__c +
blng__Unallocations__c -  blng__Refunds__c - blng__Debits__c
```

.

This field is a calculated field.

---

blng__BaseCurrencyAmount__c

Type
currency

Properties
Create, Filter, Nillable, Sort, Update

Description
The credit note line's total amount in the base currency.

---

blng__BaseCurrencyFXDate__c

Type
date

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The date used to determine the foreign exchange rate when calculating the base currency amount.

---

blng__BaseCurrencyFXRate__c

Type
double

Properties
Create, Filter, Nillable, Sort, Update

Description
The foreign exchange rate that's used to convert the credit note line's total amount to the base currency amount.

---

blng__BaseCurrency__c

Type
string

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The base currency that's used for converting currency during data export.

---

blng__BillToAccount__c

Type
reference

Field	Details

Properties

Create, Filter, Group, Nillable, Sort, Update

Description

The account that receives information about the credit note line.

This field is a relationship field.

Relationship Name

blng__BillToAccount__r

Relationship Type

Lookup

Refers To

Account

`blng__BillToContact__c`

Type

reference

Properties

Create, Filter, Group, Nillable, Sort, Update

Description

The user who receives information about the credit note line. This value is inherited from the account's Bill to Contact field.

This field is a relationship field.

Relationship Name

blng__BillToContact__r

Relationship Type

Lookup

Refers To

Contact

`blng__BillingFinanceBook__c`

Type

reference

Properties

Create, Filter, Group, Nillable, Sort, Update

Description

The finance book associated with the related credit note.

This field is a relationship field.

Relationship Name

blng__BillingFinanceBook__r

Relationship Type

Lookup

Refers To

blng__FinanceBook__c

Field	Details
blng__BillingGLRule__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The General Ledger (GL) rule that's used to determine the GL accounts for the invoice line associated with the credit note line. This field is a relationship field. **Relationship Name** blng__BillingGLRule__r **Relationship Type** Lookup **Refers To** blng__GLRule__c
blng__BillingGLTreatment__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The billing treatment associated with the billing General Ledger (GL) rule of the credit note line. This field is a relationship field. **Relationship Name** blng__BillingGLTreatment__r **Relationship Type** Lookup **Refers To** blng__GLTreatment__c
blng__BillingRule__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The billing rule associated with the credit note line. This field is a relationship field. **Relationship Name** blng__BillingRule__r **Relationship Type** Lookup

Field	Details
	Refers To blng__BillingRule__c
`blng__BillingTreatment__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The billing treatment associated with the credit note line. This field is a relationship field. **Relationship Name** blng__BillingTreatment__r **Relationship Type** Lookup **Refers To** blng__BillingTreatment__c
`blng__CalculateTax__c`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** This field indicates whether tax must be calcualed for the credit note line. The default value is `false`.
`blng__CreditNoteLineDate__c`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date when the credit note line was posted.
`blng__CreditNote__c`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The credit note associated with the credit note line. This field is a relationship field. **Relationship Name** blng__CreditNote__r

Field	Details
	Relationship Type Master-detail **Refers To** blng__CreditNote__c
blng__Debits__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The allocated amount of the debit note that's associated with the credit note line. This field is a calculated field.
blng__EffectiveTaxDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date when the tax is applied to the credit note line.
blng__EndDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The end date of the credit note line.
blng__FinancePeriod__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The finance period associated with the credit note line. This field is a relationship field. **Relationship Name** blng__FinancePeriod__r **Relationship Type** Lookup **Refers To** blng__FinancePeriod__c

Field	Details

blng__ImpactAmount__c

Type
currency

Properties
Filter, Nillable, Sort

Description
The credit note line's financial impact against the customer's accounts receivable. When the credit note line is applied, the impact amount is the negative equivalent of the credit note line's amount.

This field is a calculated field.

---

blng__InvoiceLine__c

Type
reference

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The invoice that the credit note line was applied to.

This field is a relationship field.

Relationship Name
blng__InvoiceLine__r

Relationship Type
Lookup

Refers To
blng__InvoiceLine__c

---

blng__LegalEntityReference__c

Type
string

Properties
Create, Filter, Group, idLookup, Nillable, Sort, Update

Description
The legal entity associated with the credit note line.

---

blng__LegalEntity__c

Type
reference

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The legal entity associated with the credit note line.

This field is a relationship field.

Relationship Name
blng__LegalEntity__r

Field	Details
	Relationship Type Lookup
	Refers To blng__LegalEntity__c
blng__NetAllocations__c	**Type** currency
	Properties Filter, Nillable, Sort
	Description The difference between the allocated and unallocated amounts of the credit note line. This field is a calculated field.
blng__Notes__c	**Type** textarea
	Properties Create, Nillable, Update
	Description Additional details about the credit note line.
blng__OverrideInitialRevenueEndDate__c	**Type** date
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The date that overrides the revenue end date of the revenue schedule associated with the credit note line.
blng__OverrideInitialRevenueStartDate__c	**Type** date
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The date that overrides the revenue start date of the revenue schedule associated with the credit note line.
blng__ProductName__c	**Type** string
	Properties Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The name of the product associated with the credit note line.
blng__Product__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The product associated with the credit note line. This field is a relationship field. **Relationship Name** blng__Product__r **Relationship Type** Lookup **Refers To** Product2
blng__RefundsAllocations__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount of the credit note line that's allocated to refunds. This field is a calculated field.
blng__RefundsUnallocations__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount of the credit note line that's unallocated from refunds. This field is a calculated field.
blng__Refunds__c	**Type** currency **Properties** Filter, Nillable, Sort

Field	Details

Description

The amount of the credit note line that's been refunded. This amount is calculated by using the formula

```
blng__RefundsAllocations__c - blng__RefundsUnallocations__c
```

.

This field is a calculated field.

---

blng__RevenueExpectedAmount__c

Type
currency

Properties
Create, Filter, Nillable, Sort, Update

Description
The expected revenue impact of the credit note line, if its different from the credit note line amount. You can configure creating revenue schedules for the credit note line with revenue expected amount field as the amount of the revenue schedule.

---

blng__RevenueLiabilityAmount__c

Type
currency

Properties
Create, Filter, Nillable, Sort, Update

Description
The revenue liability associated with the credit note line. You can configure creating revenue schedules for the credit note line with revenue liability amount field as the amount of the revenue schedule.

---

blng__RevenueMostLikelyAmount__c

Type
currency

Properties
Create, Filter, Nillable, Sort, Update

Description
The most likely revenue impact of the credit note line for revenue calculations when the revenue amount is uncertain.

---

blng__RevenueScheduleStatus__c

Type
picklist

Properties
Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Description
Specifies the status of the revenue scehdule associated with the credit note line.

Possible values are:

- `Complete`

Field	Details
	• Error
	• Queued

Field	Details
blng__StartDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The start date of the credit note line.
blng__Status__c	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Specifies the status of the credit note line. Possible values are: • Cancelled • Draft • Posted The default value is Draft.
blng__Subtotal__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The amount of the credit note line, excluding tax, that's not refunded. The subtotal is calculated by deducting the refund amount from the amount.
blng__TaxAmount__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The amount of tax for the credit note line.
blng__TaxCity__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The city of the address that Salesforce Billing uses to calculate tax for the credit note line.
`blng__TaxCode__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The reference code used when tax for the credit note line is calculated by an external tax engine.
`blng__TaxCountry__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The country of the address that Salesforce Billing uses to calculate tax for the credit note line.
`blng__TaxEngine__c`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The tax engine that has calculated tax for the credit note line. Possible values are: • `Avalara AvaTax` • `Standard` The default value is `Standard`.
`blng__TaxErrorMessage__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The error message when the external tax engine fails to calculate tax for the credit note line.
`blng__TaxGLRule__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details

Description

The tax rule for recording tax transactions of the credit note line in an external general ledger.

This field is a relationship field.

Relationship Name

blng__TaxGLRule__r

Relationship Type

Lookup

Refers To

blng__GLRule__c

`blng__TaxGLTreatment__c`

Type

reference

Properties

Create, Filter, Group, Nillable, Sort, Update

Description

The tax treatment for recording tax transactions of the credit note line in an external general ledger.

This field is a relationship field.

Relationship Name

blng__TaxGLTreatment__r

Relationship Type

Lookup

Refers To

blng__GLTreatment__c

`blng__TaxPercentageApplied__c`

Type

percent

Properties

Create, Filter, Nillable, Sort, Update

Description

The percentage of tax that was applied to the credit note line.

`blng__TaxRule__c`

Type

reference

Properties

Create, Filter, Group, Nillable, Sort, Update

Description

The rule used to tax the credit note line.

This field is a relationship field.

Relationship Name

blng__TaxRule__r

Field	Details

| | **Relationship Type**
Lookup |
| | **Refers To**
blng__TaxRule__c |

blng__TaxState__c	**Type** string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The state of the address that Salesforce Billing uses to calculate tax for the credit note line.

blng__TaxStatus__c	**Type** picklist
	Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description Specifies the status of the tax calculation. Possible values are: - Completed - Copied - Error - Processing - Queued - Warning

blng__TaxStreet2__c	**Type** string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The street of the address that Salesforce Billing uses to calculate tax for the credit note line.

blng__TaxStreet__c	**Type** string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The street of the address that Salesforce Billing uses to calculate tax for the credit note line.

Field	Details
blng__TaxTreatment__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The tax treatment assigned to the credit note line. Tax rules and their treatments define how Salesforce Billing processes tax for your transactions. This field is a relationship field. **Relationship Name** blng__TaxTreatment__r **Relationship Type** Lookup **Refers To** blng__TaxTreatment__c
blng__TaxZipCode__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The postal code of the address that Salesforce Billing uses to calculate tax for the credit note line.
blng__TotalAmount__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The total amount of the credit note line that's available for allocation.
blng__Unallocations__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount that's been unallocated from the credit note line. This field is a calculated field.

blng__DebitNote__c

Represents information about the positive balance change that you can allocate to an invoice line or a positive credit note line.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the debit note. The default value is `USD`.
LastActivityDate	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The timestamp for when a user last associated a task or event action with the debit note.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp of when the current user had last accessed the debit note, or a record or list view related to the debit note.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort

Field	Details
	Description
	The timestamp when the current user last viewed the the debit note, or a record or list view related to the debit note. If this value is null, it's possible the user accessed this record or list view (LastReferencedDate) but didn't view it.

Field	Details
Name	**Type**
	string
	Properties
	Autonumber, Defaulted on create, Filter, idLookup, Sort
	Description
	The name of the debit note.

Field	Details
blng__Account__c	**Type**
	reference
	Properties
	Create, Filter, Group, Sort
	Description
	The customer account associated with the debit note.
	This field is a relationship field.
	Relationship Name
	blng__Account__r
	Relationship Type
	Master-detail
	Refers To
	Account

Field	Details
blng__Allocations__c	**Type**
	currency
	Properties
	Filter, Nillable, Sort
	Description
	The amount from the debit note line that's allocated.
	This field is a calculated field.

Field	Details
blng__AvailableForPayment__c	**Type**
	currency
	Properties
	Filter, Nillable, Sort
	Description
	The amount from the debit note that's available for payment.
	This field is a calculated field.

Field	Details
`blng__Balance__c`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Amount of the debit note that's available for application. This field is a calculated field.
`blng__BaseCurrencyAmount__c`	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The debit note's total amount in the base currency.
`blng__BaseCurrencyFXDate__c`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date used to determine the foreign exchange rate when calculating the base currency amount.
`blng__BaseCurrencyFXRate__c`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The foreign exchange rate that's used to convert the debit note's total amount to the base currency amount.
`blng__BaseCurrency__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The base currency that's used for converting currency during data export.
`blng__BillToAccount__c`	**Type** reference

Field	Details

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The account that receives information about the debit note.

This field is a relationship field.

Relationship Name
blng__BillToAccount__r

Relationship Type
Lookup

Refers To
Account

---

blng__BillToContact__c

Type
reference

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The user who receives information about the debit note. This value is inherited from the account's Bill to Contact field.

This field is a relationship field.

Relationship Name
blng__BillToContact__r

Relationship Type
Lookup

Refers To
Contact

---

blng__BillingFinanceBook__c

Type
reference

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The finance book associated with the debit note.

This field is a relationship field.

Relationship Name
blng__BillingFinanceBook__r

Relationship Type
Lookup

Refers To
blng__FinanceBook__c

Field	Details
blng__BillingFinancePeriod__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The finance period associated with the debit note. This field is a relationship field. **Relationship Name** blng__BillingFinancePeriod__r **Relationship Type** Lookup **Refers To** blng__FinancePeriod__c
blng__CreditNoteLineAllocations__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount allocated from all the related credit note lines. This field is a calculated field.
blng__CreditNoteLineUnallocations__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount that's unallocated from all the related credit note lines. This field is a calculated field.
blng__DebitNoteDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date when the debit note was posted.
blng__DebitNoteSourceAction__c	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details
	Description
	Specifies the action that created the debit note.
	Possible values are:
	• `Manual`
	The default value is `Manual`.
`blng__EffectiveTaxDate__c`	**Type**
	date
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The date when the tax is applied to the debit note.
`blng__ImpactAmount__c`	**Type**
	currency
	Properties
	Filter, Nillable, Sort
	Description
	The debit note's financial impact against the customer's accounts receivable. When the debit note is applied, the impact amount is equivalent to its total amount.
	This field is a calculated field.
`blng__InvoiceLineAllocations__c`	**Type**
	currency
	Properties
	Filter, Nillable, Sort
	Description
	The amount from the debit note that's allocated to an invoice line.
	This field is a calculated field.
`blng__InvoiceLineUnallocations__c`	**Type**
	currency
	Properties
	Filter, Nillable, Sort
	Description
	The amount from the debit note that's unallocated from an invoice line.
	This field is a calculated field.
`blng__NetCreditNoteAllocations__c`	**Type**
	currency

Field	Details
	Properties
	Filter, Nillable, Sort
	Description
	The total amount of the related credit note that's allocated to an invoice line. The net credit net allocations are calculated by deducting the related credit note line's unallocated amount from its allocated amount.
	This field is a calculated field.
blng__NetInvoiceAllocations__c	**Type**
	currency
	Properties
	Filter, Nillable, Sort
	Description
	The total amount of the debit note that's allocated to invoices. The net invoice allocations are calculated by deducting the related invoice line's unallocated amount from its allocated amount.
	This field is a calculated field.
blng__NetPaymentAllocations__c	**Type**
	currency
	Properties
	Filter, Nillable, Sort
	Description
	The total amount of the debit note that's allocated to payments. The net payment allocations are calculated by deducting the debit note's refund amount from its payment amount.
	This field is a calculated field.
blng__Notes__c	**Type**
	textarea
	Properties
	Create, Nillable, Update
	Description
	Additional details about the debit note.
blng__PaymentBalance__c	**Type**
	currency
	Properties
	Filter, Nillable, Sort
	Description
	The amount of the debit note after excluding payments and refunds.
	This field is a calculated field.

Field	Details

blng__Payments1__c

Type
currency

Properties
Filter, Nillable, Sort

Description
The payment made from the debit note's amount after deducting the refund amount.

This field is a calculated field.

blng__Payments__c

Type
currency

Properties
Filter, Nillable, Sort

Description
The payment made from the debit note's amount.

This field is a calculated field.

blng__Refunds__c

Type
currency

Properties
Filter, Nillable, Sort

Description
The refund that's added to the debit note's amount.

This field is a calculated field.

blng__Status__c

Type
picklist

Properties
Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Description
Specifies the status of the debit note.

Possible values are:

- Cancelled
- Draft
- Posted

The default value is Draft.

blng__Subtotal__c

Type
currency

Properties
Filter, Nillable, Sort

Field	Details
	Description
	The sum of subtotals of all the related debit note lines.
	This field is a calculated field.
blng__TaxAmount__c	**Type**
	currency
	Properties
	Filter, Nillable, Sort
	Description
	The amount of tax for the debit note.
	This field is a calculated field.
blng__TaxCity__c	**Type**
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The city of the address that Salesforce Billing uses to calculate tax for the debit note.
blng__TaxCountry__c	**Type**
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The country of the address that Salesforce Billing uses to calculate tax for the debit note.
blng__TaxErrorMessage__c	**Type**
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The error message when the external tax engine fails to calculate tax for the debit note.
blng__TaxState__c	**Type**
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The state of the address that Salesforce Billing uses to calculate tax for the debit note.

Field	Details
blng__TaxStreet2__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The street of the address that Salesforce Billing uses to calculate tax for the debit note.
blng__TaxStreet__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The street of the address that Salesforce Billing uses to calculate tax for the debit note.
blng__TaxZipCode__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The postal code of the address that Salesforce Billing uses to calculate tax for the debit note.
blng__TotalAmount__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total amount of the debit note. This field is a calculated field.
blng__TotalAvailableForAllocation__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total amount of the debit note that's available for allocation. This amount is calculated by deducting the debit note's total net allocations from the total amount. This field is a calculated field.
blng__TotalNetAllocations__c	**Type** currency

Field	Details
	Properties
	Filter, Nillable, Sort
	Description
	The amount that's been allocated by the credit note that's associated with the debit note. The total net allocations are calculated by summing up the net credit note and invoice allocations.
	This field is a calculated field.
blng__Unallocations__c	**Type**
	currency
	Properties
	Filter, Nillable, Sort
	Description
	The amount that's been unallocated by the debit note.
	This field is a calculated field.

blng__DebitNoteLine__c

Represents information about the amount that's allocated to increase the balance of an invoice line.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type**
	picklist
	Properties
	Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	The three-letter ISO 4217 currency code associated with the debit note line.
	The default value is USD.
Name	**Type**
	string

Field	Details
	Properties Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the debit note line.
blng__AllocatedRevenueAmount__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The debit note's revenue amount that's allocated to invoice lines.
blng__Allocations__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The debit note's amount that's allocated to invoice lines, excluding tax. This field is a calculated field.
blng__AvailableBalance__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** Amount of the debit note line that's available for allocation or refund. The balance is calculated using the formula ```\nblng__TotalAmount__c - blng__NetPaymentAllocations__c -\nblng__InvoiceLineNetAllocations__c -\nblng__CreditNoteLineNetAllocations__c\n``` . This field is a calculated field.
blng__AvailableForPayment__c	**Type** currency **Properties** Filter, Nillable, Sort

Field	Details

	Description
	The amount from the debit note line that's available for payment. This amount is calculated using the formula
	
	`blng__TotalPaymentAmount__c  -  blng__TotalRefundAmount__c`
	`-   blng__NetPaymentAllocations__c`
	
	.
	
	This field is a calculated field.

`blng__BaseCurrencyAmount__c`	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The debit note line's total amount in the base currency.
`blng__BaseCurrencyFXDate__c`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date used to determine the foreign exchange rate when calculating the base currency amount.
`blng__BaseCurrencyFXRate__c`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The foreign exchange rate that's used to convert the debit note line's total amount to the base currency amount.
`blng__BaseCurrency__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The base currency that's used for converting currency during data export.
`blng__BillToAccount__c`	**Type** reference

Field	Details

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The account that receives information about the debit note line.

This field is a relationship field.

Relationship Name
blng__BillToAccount__r

Relationship Type
Lookup

Refers To
Account

---

blng__BillToContact__c

Type
reference

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The user who receives information about the debit note line. This value is inherited from the account's Bill to Contact field.

This field is a relationship field.

Relationship Name
blng__BillToContact__r

Relationship Type
Lookup

Refers To
Contact

---

blng__BillingFinanceBook__c

Type
reference

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The finance book associated with the related debit note.

This field is a relationship field.

Relationship Name
blng__BillingFinanceBook__r

Relationship Type
Lookup

Refers To
blng__FinanceBook__c

Field	Details
blng__BillingGLRule__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The General Ledger (GL) rule that's used to determine the GL accounts for the invoice line associated with the debit note line. This field is a relationship field. **Relationship Name** blng__BillingGLRule__r **Relationship Type** Lookup **Refers To** blng__GLRule__c
blng__BillingGLTreatment__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The billing treatment associated with the billing General Ledger (GL) rule of the debit note line. This field is a relationship field. **Relationship Name** blng__BillingGLTreatment__r **Relationship Type** Lookup **Refers To** blng__GLTreatment__c
blng__BillingRule__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The billing rule associated with the debit note line. This field is a relationship field. **Relationship Name** blng__BillingRule__r **Relationship Type** Lookup

Field	Details
	Refers To blng__BillingRule__c
blng__BillingTreatment__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The billing treatment associated with the debit note line. This field is a relationship field. **Relationship Name** blng__BillingTreatment__r **Relationship Type** Lookup **Refers To** blng__BillingTreatment__c
blng__CalculateTax__c	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** This field indicates whether tax must be calculated for the debit note line. The default value is `false`.
blng__CreditNoteLineAllocations__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount allocated from all the related credit note lines. This field is a calculated field.
blng__CreditNoteLineNetAllocations__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total amount allocated from all the related credit note lines. This field is a calculated field.

Field	Details
blng__CreditNoteLineUnallocations__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount that's unallocated from all the related credit note lines. This field is a calculated field.
blng__DebitNoteLineDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date when the debit note line was posted.
blng__DebitNote__c	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The debit note associated with the debit note line. This field is a relationship field. **Relationship Name** blng__DebitNote__r **Relationship Type** Master-detail **Refers To** blng__DebitNote__c
blng__EffectiveTaxDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date when the tax is applied to the debit note line.
blng__EndDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details

Description

The end date of the debit note line.

---

blng__FinancePeriod__c

Type
reference

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The finance period associated with the debit note line.

This field is a relationship field.

Relationship Name
blng__FinancePeriod__r

Relationship Type
Lookup

Refers To
blng__FinancePeriod__c

---

blng__ImpactAmount__c

Type
currency

Properties
Filter, Nillable, Sort

Description
The debit note line's financial impact against the customer's accounts receivable. When the debit note line is applied, the impact amount is equivalent to the debit note line's total amount.

This field is a calculated field.

---

blng__InvoiceLineNetAllocations__c

Type
currency

Properties
Filter, Nillable, Sort

Description
The amount from the debit note line that's allocated to an invoice line. This amount is calculated by deducting the debit note line's unallocated amount from its allocated amount.

This field is a calculated field.

---

blng__LegalEntityReference__c

Type
string

Properties
Create, Filter, Group, idLookup, Nillable, Sort, Update

Field	Details
	Description
	The legal entity associated with the debit note line.

Field	Details
blng__LegalEntity__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The legal entity associated with the debit note line.
	This field is a relationship field.
	Relationship Name
	blng__LegalEntity__r
	Relationship Type
	Lookup
	Refers To
	blng__LegalEntity__c

Field	Details
blng__NetPaymentAllocations__c	**Type**
	currency
	Properties
	Filter, Nillable, Sort
	Description
	The total amount of the debit note line that's allocated to payments. The net payment allocations are calculated by deducting the debit note line's refund amount from its payment amount.
	This field is a calculated field.
	Formula

```
blng__Payments__c -  blng__Refunds__c
```

Field	Details
blng__Notes__c	**Type**
	textarea
	Properties
	Create, Nillable, Update
	Description
	Additional details about the debit note line.

Field	Details
blng__OverrideInitialRevenueEndDate__c	**Type**
	date
	Properties
	Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description
	The date that overrides the revenue end date of the revenue schedule associated with the debit note line.
blng__OverrideInitialRevenueStartDate__c	**Type**
	date
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The date that overrides the revenue start date of the revenue schedule associated with the debit note line.
blng__PaymentBalance__c	**Type**
	currency
	Properties
	Filter, Nillable, Sort
	Description
	The amount of the debit note line after excluding payments and refunds.
	This field is a calculated field.
blng__Payments1__c	**Type**
	currency
	Properties
	Filter, Nillable, Sort
	Description
	The payment made from the debit note line's amount after deducting the refund amount.
	This field is a calculated field.
blng__Payments__c	**Type**
	currency
	Properties
	Filter, Nillable, Sort
	Description
	The payment made from the debit note line's amount.
	This field is a calculated field.
blng__ProductName__c	**Type**
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The name of the product associated with the debit note line.
blng__Product__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The product associated with the debit note line. This field is a relationship field. **Relationship Name** blng__Product__r **Relationship Type** Lookup **Refers To** Product2
blng__Refunds__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount of the debit note line that's been refunded. This field is a calculated field.
blng__RevenueAgreementMatchingID__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** This field is used to group debit note lines with the same revenue agreement matching ID.
blng__RevenueExpectedAmount__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The expected revenue impact of the related credit note line.
blng__RevenueLiabilityAmount__c	**Type** currency

Field	Details

Properties

Create, Filter, Nillable, Sort, Update

Description

The revenue liability of the related credit note line.

---

blng__RevenueMostLikelyAmount__c

Type

currency

Properties

Create, Filter, Nillable, Sort, Update

Description

The most likely revenue impact of the related credit note line for revenue calculations when the revenue amount is uncertain.

---

blng__RevenueScheduleStatus__c

Type

picklist

Properties

Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Description

Specifies the status of the revenue schedule associated with the debit note line.

Possible values are:

- Complete
- Error
- Queued

---

blng__RevenueTransactionAmount__c

Type

currency

Properties

Create, Filter, Nillable, Sort, Update

Description

The amount of the revenue transaction that's associated with the debit note line.

---

blng__RevenueTransactionPrice__c

Type

currency

Properties

Create, Filter, Nillable, Sort, Update

Description

The price of the revenue transaction that's associated with the debit note line.

---

blng__StandaloneSellingAmount__c

Type

currency

Field	Details
	Properties Create, Filter, Nillable, Sort, Update **Description** The amount of the standalone order that's associated with the debit note line.
blng__StandaloneSellingPrice__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The price of the standalone order that's associated with the debit note line.
blng__StartDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The start date of the debit note line.
blng__Status__c	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Specifies the status of the debit note line. Possible values are: • Cancelled • Draft • Posted The default value is Draft.
blng__Subtotal__c	**Type** currency **Properties** Create, Filter, Sort, Update **Description** The amount of the debit note line, excluding tax, that's not refunded. The subtotal is calculated by deducting the refund amount from the debit note line's amount.
blng__TaxAmount__c	**Type** currency

Field	Details
	Properties Create, Filter, Nillable, Sort, Update **Description** The amount of tax for the debit note line.
blng__TaxCity__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The city of the address that Salesforce Billing uses to calculate tax for the debit note line.
blng__TaxCountry__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The country of the address that Salesforce Billing uses to calculate tax for the debit note line.
blng__TaxErrorMessage__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The error message when the external tax engine fails to calculate tax for the debit note line.
blng__TaxGLRule__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The tax rule for recording tax transactions of the debit note line in an external general ledger. This field is a relationship field. **Relationship Name** blng__TaxGLRule__r **Relationship Type** Lookup **Refers To** blng__GLRule__c

Field	Details
`blng__TaxGLTreatment__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The tax treatment for recording tax transactions of the debit note line in an external general ledger. This field is a relationship field. **Relationship Name** blng__TaxGLTreatment__r **Relationship Type** Lookup **Refers To** blng__GLTreatment__c
`blng__TaxPercentageApplied__c`	**Type** percent **Properties** Create, Filter, Nillable, Sort, Update **Description** The percentage of tax that was applied to the debit note line.
`blng__TaxRule__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The rule used to tax the debit note line. This field is a relationship field. **Relationship Name** blng__TaxRule__r **Relationship Type** Lookup **Refers To** blng__TaxRule__c
`blng__TaxState__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The state of the address that Salesforce Billing uses to calculate tax for the debit note line.
blng__TaxStatus__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Specifies the status of the tax calculation. Possible values are: • `Completed` • `Error` • `Processing` • `Queued` • `Warning`
blng__TaxStreet2__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The street of the address that Salesforce Billing uses to calculate tax for the debit note line.
blng__TaxStreet__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The street of the address that Salesforce Billing uses to calculate tax for the debit note line.
blng__TaxTreatment__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The tax treatment that's assigned to the debit note line. Tax rules and their treatments define how Salesforce Billing processes tax for your transactions. This field is a relationship field. **Relationship Name** blng__TaxTreatment__r

Field	Details
	Relationship Type Lookup
	Refers To blng__TaxTreatment__c
blng__TaxZipCode__c	**Type** string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The postal code of the address that Salesforce Billing uses to calculate tax for the debit note line.
blng__TotalAmount__c	**Type** currency
	Properties Create, Filter, Nillable, Sort, Update
	Description The total amount of the debit note line.
blng__TotalAvailableForAllocation__c	**Type** currency
	Properties Filter, Nillable, Sort
	Description The total amount of the debit note line that's available for allocation. This amount is calculated by using the formula `blng__TotalAmount__c - blng__TotalNetAllocations__c` . This field is a calculated field.
blng__TotalNetAllocations__c	**Type** currency
	Properties Filter, Nillable, Sort
	Description The amount allocated by the credit note that's associated with the debit note line. This field is a calculated field.
blng__TotalPaymentAmount__c	**Type** currency

Field	Details
	Properties Filter, Nillable, Sort **Description** The total amount that's been paid by using the debit note line. This field is a calculated field.
`blng__TotalRefundAmount__c`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total amount that's been refunded to the debit note line. This field is a calculated field.
`blng__Unallocations__c`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount that's been unallocated from the debit note line. This field is a calculated field.

blng__FinanceBook__c

Represents the storage of financial transactions for reporting and recordkeeping.

Supported Calls

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

Fields

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

Field	Details
	Description The three-letter ISO 4217 currency code associated with the finance book record when multiple currencies are enabled.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a finance book record indirectly, for example, through a list view or related record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a finance book record. If this value is null, it's possible that the user only accessed the finance book record or a related list view (LastReferencedDate), but not viewed the finance book record itself.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the finance book.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. The ID of the record owner. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
blng__Active__c	**Type** boolean

Field	Details
	Properties Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. Indicates whether the finance book is active (`true`) or not (`false`). To deactivate the finance book, clear the `Active` checkbox. This setting won't affect past or existing transactions or assignments. The default value is `true`.
`blng__FinancePeriodDuration__c`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. The duration of the finance period date range. Valid values are: • `Custom` • `Monthly` The default value is `Monthly`.
`blng__Notes__c`	**Type** textarea **Properties** Create, Nillable, Update **Description** Additional details or comments related to the finance book.
`blng__PeriodType__c`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. The type of finance book. Valid values are: • `Accounting` • `Revenue`

blng__FinancePeriod__c

Represents a reference object that stores lookups to all accounting or revenue transactions that occur within a specific period.

> ⊘ **Important:** Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the finance period record when multiple currencies are enabled.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a finance period record indirectly, for example, through a list view or related record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a finance period record. If this value is null, it's possible that the user only accessed the finance period record or a related list view (`LastReferencedDate`), but not viewed the finance period record itself.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the finance period.

Field	Details
`blng__Family__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The product family associated with the finance period record.
`blng__FinanceBook__c`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Required. The financial book associated with the billing. This field helps in categorizing and managing financial records according to different finance books. This field is a relationship field. **Relationship Name** blng__FinanceBook__r **Relationship Type** Master-detail **Refers To** blng__FinanceBook__c (the master object)
`blng__FinancePeriodFamily__c`	**Type** string **Properties** Create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The finance period family associated with the finance period.
`blng__LegalEntity__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The legal entity used to associate the finance period with other transactional records in Salesforce Billing. This field is a relationship field. **Relationship Name** blng__LegalEntity__r **Refers To** blng__LegalEntity__c

Field	Details
blng__NextOpenPeriod__c	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. The next open financial period for billing purposes. The default value is `false`.
blng__Notes__c	**Type** textarea **Properties** Create, Nillable, Update **Description** Additional details or comments related to the finance period.
blng__PeriodEndDate__c	**Type** date **Properties** Create, Filter, Group, Sort, Update **Description** Required. The end date of the financial period.
blng__PeriodStartDate__c	**Type** date **Properties** Create, Filter, Group, Sort, Update **Description** Required. The start date of the finance period.
blng__PeriodStatus__c	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. The status of a financial period to track the state of a financial period. Valid values are: • `Closed` • `Error` • `Open` • `Pending Close`

Field	Details
	• `Pending Open`
`blng__PeriodType__c`	**Type** string **Properties** Filter, Nillable, Sort **Description** The period type of the finance book. This field is a calculated field.
`blng__ReferenceId__c`	**Type** string **Properties** Create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The reference to source for which the finance period record is created.
`blng__UniqueId__c`	**Type** string **Properties** Create, Filter, Group, idLookup, Nillable, Sort, Update **Description** A unique identification number associated with the finance period record. This field is unique within your organization.

blng__GLAccount__c

Represents the accounting journal entries in a General Ledger (GL) account. You can define both a credit GL account and a debit GL account as endpoints for these entries. These GL accounts, along with all associated records, can later be exported to an external GL bookkeeping system.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the legacy entity when multiple currencies are enabled.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a GL account record indirectly, for example, through a list view or related record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a GL account record. If this value is null, it's possible that the user only accessed the GL account record or a related list view (`LastReferencedDate`), but not viewed the GL account record itself.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the GL account.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. The ID of the record owner. This field is a polymorphic relationship field.

Field	Details
	Relationship Name Owner
	Refers To Group, User
blng__Active__c	**Type** boolean
	Properties Create, Defaulted on create, Filter, Group, Sort, Update
	Description Required. Indicates whether the GL account is active (`true`) or not (`false`). To deactivate the GL account, clear the `Active` checkbox. This setting won't affect past or existing transactions or assignments. The default value is `true`.
blng__GLAccountDescription__c	**Type** textarea
	Properties Create, Nillable, Update
	Description The purpose and usage of a GL account.
blng__GLAccountNumber__c	**Type** string
	Properties Create, Filter, Group, Sort, Update
	Description Required. The number that's used to organize the details about the GL account.
blng__Notes__c	**Type** textarea
	Properties Create, Nillable, Update
	Description Additional details or comments related to the GL account.

blng__GLRule__c

Represents how transactions are recorded and categorized in the General Ledger (GL).

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the GL when multiple currencies are enabled.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a GL rule record indirectly, for example, through a list view or related record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a GL rule record. If this value is null, it's possible that the user only accessed the GL rule record or a related list view (`LastReferencedDate`), but not viewed the GL rule record itself.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the GL rule.
OwnerId	**Type** reference

Field	Details
	Properties Create, Defaulted on create, Filter, Group, Sort, Update
	Description Required. The ID of the record owner. This field is a polymorphic relationship field.
	Relationship Name Owner
	Refers To Group, User
blng__Active__c	**Type** boolean
	Properties Create, Defaulted on create, Filter, Group, Sort, Update
	Description Required. Indicates whether the GL rule is active (`true`) or not (`false`). To deactivate the GL rule, clear the `Active` checkbox. This setting won't affect past or existing transactions or assignments. The default value is `true`.
blng__Notes__c	**Type** textarea
	Properties Create, Nillable, Update
	Description Additional details or comments related to the GL rule.

blng__GLTreatment__c

Represents the lookups to a credit General Ledger (GL) account and a debit GL account. These lookups establish a relationship between your transaction and your GL accounts.

🛑 **Important:** Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Supported Calls

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

Salesforce Billing Custom Objects

blng__GLTreatment__c

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the General Ledger (GL) treatment when multiple currencies are enabled.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a GL treatment record indirectly, for example, through a list view or related record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a GL treatment record. If this value is null, it's possible that the user only accessed the GL treatment record or a related list view (`LastReferencedDate`), but not viewed the GL treatment record itself.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the GL treatment.
blng__Active__c	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. Indicates whether the GL treatment is active (`true`) or not (`false`). To deactivate the GL treatment, clear the `Active` checkbox. This setting won't affect past or existing transactions or assignments. The default value is `true`.

Field	Details
blng__CreditGLAccount__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The credit GL account transactions associated with the GL treatment.
	This field is a relationship field.
	Relationship Name
	blng__CreditGLAccount__r
	Refers To
	blng__GLAccount__c
blng__DebitGLAccount__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The debit GL account transactions associated with the GL treatment.
	This field is a relationship field.
	Relationship Name
	blng__DebitGLAccount__r
	Refers To
	blng__GLAccount__c
blng__GLLegalEntity__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The legal entity associated with the GL treatment.
	This field is a relationship field.
	Relationship Name
	blng__GLLegalEntity__r
	Refers To
	blng__LegalEntity__c
blng__GLRule__c	**Type**
	reference
	Properties
	Create, Filter, Group, Sort

Field	Details
	Description Required. Specifies how Salesforce Billing logs transactional data to the finance books. A GL rule includes a list of GL treatments. This field is a relationship field. **Relationship Name** blng__GLRule__r **Relationship Type** Master-detail **Refers To** blng__GLRule__c (the master object)
blng__Notes__c	**Type** textarea **Properties** Create, Nillable, Update **Description** Additional details or comments related to the GL treatment.
blng__UniqueId__c	**Type** string **Properties** Create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The unique identification number associated with the general ledger treatment record. This field is unique within your organization.

blng__Invoice__c

Represents a financial document describing the total amount a buyer must pay for provided goods or services.

🛇 Important: Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the invoice when multiple currencies are enabled.
Custom_Lock__c	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Specifies whether the invoice is locked or unlocked. Valid values are: • Locked • Unlocked The default value is Unlocked.
LastActivityDate	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The timestamp of when a user had last associated a task or event action with the invoice.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp of when the current user had last accessed the invoice, or a record or list view related to the invoice.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort

Field	Details

| | **Description**
The timestamp when a user last viewed this record, a record related to this record, or a list view. If this value is null, it's possible the user accessed this record or list view (`LastReferencedDate`) but didn't view it. |

Name	**Type** string
	Properties Autonumber, Defaulted on create, Filter, idLookup, Sort
	Description The name of the invoice.

blng__ARStatus__c	**Type** picklist
	Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description Shows whether the invoice has been credited, or canceled and rebilled. Valid values are: • `Cancel and Rebill` • `Credited`

blng__Account__c	**Type** reference
	Properties Create, Filter, Group, Sort
	Description The invoice's parent account that's inherited from the order product. This is a relationship field.
	Relationship Name blng__Account__r
	Relationship Type Master-detail
	Refers To Account (the master object)

| blng__Action__c | **Type**
picklist |
| | **Properties**
Create, Filter, Group, Nillable, Restricted picklist, Sort, Update |

Field	Details

Description

Specifies whether a user has credited the invoice by either selecting the **Cancel** and **Rebill** button or the **Credit** button.

Valid values are:

- Cancel and Rebill
- Credit

---

blng__Balance__c

Type
currency

Properties
Filter, Nillable, Sort

Description
The sum of balances of all the related invoice lines. This amount is equal to the invoice's total amount with tax, after deducting the payments and adjustments made. This is a calculated field.

---

blng__BaseCurrencyAmount__c

Type
currency

Properties
Create, Filter, Nillable, Sort, Update

Description
If you have currency conversion requirements, you can define your base currency amount here. When you export your data, you can use this value in your conversions.

---

blng__BaseCurrencyFXDate__c

Type
date

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
If you have currency conversion requirements, you can define your base currency foreign exchange (FX) date here. When you export your data, you can use this value in your conversions.

---

blng__BaseCurrencyFXRate__c

Type
double

Properties
Create, Filter, Nillable, Sort, Update

Description
If you have currency conversion requirements, you can define your base currency FX rate here. When you export your data, you can use this value in your conversions.

Field	Details
blng__BaseCurrency__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** If you have currency conversion requirements, you can define your base currency here.
blng__BillToContact__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The user who receives information about the invoice. This is a relationship field. **Relationship Name** blng__BillToContact__r **Relationship Type** Lookup **Refers To** Contact
blng__CorrectiveAction__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** When this field has a value of `Action Required`, the invoice is locked from receiving payments until the issue defined in the `Last Payment Run Processing Message` field is resolved. •
blng__Credits__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of credit notes allocated against the invoice and its invoice lines. Sum of all credit note amounts applied to this invoice. This is a calculated field.
blng__DaysOutstanding__c	**Type** double

Field	Details
	Properties Filter, Nillable, Sort **Description** The number of days past the invoice due date.
blng__DebitPayments__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of payments made to debit notes allocated against the invoice and its invoice lines. This is a calculated field.
blng__Debits__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of debit notes allocated against the invoice and its invoice lines. This is a calculated field.
blng__DeclinedPaymentRunCount__c	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** Salesforce Billing doesn't populate this field by default. You can add your own customization to it when you want to track the number of times a payment run that targets the invoice has been unable to pay it.
blng__DefaultPaymentType__c	**Type** string **Properties** Filter, Nillable, Sort **Description** Used in payment runs. A payment run picks up an invoice when the run and the invoice have matching Default Payment Type values. This is a calculated field.
blng__DueDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description
	The date by which customers must pay their invoice. This field is provided for users who want to add their own due date and collections features for AR and AP.
blng__ImpactAmount__c	**Type**
	currency
	Properties
	Filter, Nillable, Sort
	Description
	Inherits the value of the invoice's total amount. This field is provided for users who must monitor impact amounts separately as part of their billing and collections processes. This is a calculated field.
blng__InvoiceDate__c	**Type**
	date
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The date the invoice was created either by the bill now process or an invoice run.
blng__InvoiceLocked__c	**Type**
	boolean
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	Indicates whether the invoice is locked (true) or not (false). Before unlocking it, check whether a payment has already been created against this invoice. The default value is `false`.
blng__InvoicePostedDate__c	**Type**
	date
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The date the invoice was posted.
blng__InvoiceRunCreatedBy__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	Specifies the invoice run that created the invoice. This is a relationship field.

Field	Details
	Relationship Name blng__InvoiceRunCreatedBy__r
	Relationship Type Lookup
	Refers To blng__InvoiceRun__c
blng__InvoiceStatus__c	**Type** picklist
	Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description The state of the invoice. Valid values are: • Cancelled • Draft • Error • Initiated • Post In progress • Posted • Rebilled The default value is Draft.
blng__LastPaymentRunId__c	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description Specifies the most recent payment run that evaluated the invoice for payment. This is a relationship field.
	Relationship Name blng__LastPaymentRunId__r
	Relationship Type Lookup
	Refers To blng__PaymentRun__c
blng__LastPaymentRunProcessingMessage__c	**Type** string
	Properties Create, Filter, Group, Nillable, Sort, Update

Field	Details

Description

If a payment run that targeted the invoice encountered an issue, this field shows information on how to correct the issue and unlock the invoice for receiving payments.

---

blng__NegativeLineConversion__c

Type

currency

Properties

Filter, Nillable, Sort

Description

The sum of all `Negative Line Conversion` fields on the invoice's invoice lines. This is a calculated field.

---

blng__Notes__c

Type

textarea

Properties

Create, Nillable, Update

Description

Specifies any additional notes and information from a user.

---

blng__NumberOfInvoiceLines__c

Type

double

Properties

Filter, Nillable, Sort

Description

The total number of invoice lines. This is a calculated field.

---

blng__Order__c

Type

reference

Properties

Create, Filter, Group, Nillable, Sort, Update

Description

Populated only when Bill Now is used to create the invoice, or during an invoice run if the order product `Invoice Group` field has a value of Order. This is a relationship field.

Relationship Name

blng__Order__r

Relationship Type

Lookup

Refers To

Order

---

blng__PaymentBatch__c

Type

picklist

Field	Details
	Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description If a payment scheduler has a payment batch value, the payment run evaluates only invoices with matching payment batch values. If this field is null, payment runs evaluate the invoice only if the payment run's batch is also null.
blng__PaymentMethod__c	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description Payment methods store Automated Clearing House (ACH) or credit card details for a customer, and the payment gateway that accepts payments. This is a relationship field.
	Relationship Name blng__PaymentMethod__r
	Relationship Type Lookup
	Refers To blng__PaymentMethod__c
blng__PaymentProcessStatus__c	**Type** picklist
	Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description The status of the payment process. Valid values are: • Completed • Failed • Processing • Started
blng__PaymentRunId__c	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The ID of the payment run that evaluated and made payment against the invoice. This is a relationship field.
	Relationship Name blng__PaymentRunId__r

Field	Details
	Relationship Type Lookup
	Refers To blng__PaymentRun__c
blng__PaymentRunProcessingMessage__c	**Type** string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description This field doesn't have any functionality by default.
blng__PaymentStatus__c	**Type** string
	Properties Filter, Nillable, Sort
	Description A formula field that shows `Paid` when the invoice balance is zero (0). If payments are applied but there's still a balance, the formula returns `Partially Paid`. If there are no payments applied, then the field shows `Unpaid`. If only credit notes are allocated to an invoice and the balance is still positive, the payment status remains `Unpaid`. Only applying payments to the invoice can change the status to `Partially Paid`. If this is a requirement, create a custom field. This is a calculated field.
blng__PaymentsAgainstInvoiceLines__c	**Type** currency
	Properties Filter, Nillable, Sort
	Description Net amount of payments against the invoice line in the invoice. This is a calculated field.
blng__PaymentsAgainstInvoice__c	**Type** currency
	Properties Filter, Nillable, Sort
	Description The sum of payment amount allocated to the invoice header. This field is used only when not using line level allocations. This is a calculated field.
blng__Payments__c	**Type** currency

Field	Details
	Properties Filter, Nillable, Sort **Description** The number of payments made against this invoice. This is a calculated field.
blng__RefundsAgainstInvoiceLines__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** Cancellations of previous allocations to the invoice lines. This is a calculated field.
blng__RefundsAgainstInvoice__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** Cancellations of previous allocations to the invoice. This is a calculated field.
blng__StatusFlag__c	**Type** string **Properties** Filter, Nillable, Sort **Description** Specifies the status of the invoice, with different color codes representing each status. Valid values are: • `Orange`— Active with balance • `Green`— No balance due • `White`— Canceled This is a calculated field.
blng__Subtotal__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount to settle on this invoice, excluding tax. This is a calculated field.
blng__TargetDate__c	**Type** date

Field	Details
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The earliest date for invoicing.
`blng__TaxAmount__c`	**Type**
	currency
	Properties
	Filter, Nillable, Sort
	Description
	The sum of all taxes for this invoice. This is a calculated field.
`blng__TaxErrorMessage__c`	**Type**
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The notification about the issue related to taxes on the invoice.
`blng__TaxErrors__c`	**Type**
	double
	Properties
	Filter, Nillable, Sort
	Description
	Specifies a problem with the tax details associated with the invoice. This is a calculated field.
`blng__TaxStatus__c`	**Type**
	picklist
	Properties
	Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	The type of tax treatment this invoice qualifies for. Valid values are:
	• `Cancelled`
	• `Completed`
	• `Error`
	• `Posted`
	• `Queued`
`blng__TotalAmount__c`	**Type**
	currency

Field	Details
	Properties
	Filter, Nillable, Sort
	Description
	The total amount to be invoiced, including tax. This is a calculated field.
`blng__UniqueId__c`	**Type**
	string
	Properties
	Create, Filter, Group, idLookup, Nillable, Sort, Update
	Description
	The unique identification number associated with this invoice. This field has unique set to `true`.This field is unique within your organization.

blng__InvoiceLine__c

Represents the amount that a buyer must pay for a product, service, or fee. Invoice lines are created based on the amount of an order line.

🛑 **Important:** Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Supported Calls

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

Fields

Field	Details
`AVA_BLNG__Invoice_Message__c`	**Type**
	textarea
	Properties
	Create, Nillable, Update
	Description
	A summary of the invoice.
`CurrencyIsoCode`	**Type**
	picklist
	Properties
	Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details
	Description The three-letter ISO 4217 currency code associated with the invoice line when multiple currencies are enabled.
LastActivityDate	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The timestamp of when a user had last associated a task or event action with the invoice line.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the invoice line.
blng__ARStatus__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Specifies whether the invoice line has been credited, or canceled and rebilled. Valid values are: • Cancel and Rebill • Credited
blng__AllocatedRevenueAmount__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The total revenue impact of this invoice line. Possible scenarios of the allocated revenue amount: • Can differ from the subtotal and total amount (with tax). • Used when reallocating revenue among several transactions. • Can be configured to create revenue schedules for this invoice line using this field as the amount.

Field	Details
blng__Asset__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Inherited by default from the `Lifecycle-Managed Asset` field on the invoice line's parent order product. This field can be overridden with a different asset. This is a relationship field. **Relationship Name** blng__Asset__r **Relationship Type** Lookup **Refers To** Asset
blng__Balance__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total amount of the invoice that isn't settled, which equals total amount (with tax) − allocations + unallocations. This is a calculated field.
blng__BalancewithoutDebits__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total amount of the invoice that isn't settled, excluding debit note allocations. The balance without debits amount equals the total amount (with tax) − payments + unallocations against invoice lines − credits. This is a calculated field.
blng__BaseCurrencyAmount__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The credit note line's total amount, converted into the base currency.
blng__BaseCurrencyFXDate__c	**Type** date

Field	Details
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The date used to determine the foreign exchange rate when calculating the base currency amount.
blng__BaseCurrencyFXRate__c	**Type**
	double
	Properties
	Create, Filter, Nillable, Sort, Update
	Description
	The FX rate used to convert the credit note amount to the base currency amount.
blng__BaseCurrency__c	**Type**
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The base currency of this credit note line that's used for reporting purposes.
blng__BillingFinanceBook__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	This invoice line's transactions report to this finance book. This is a relationship field.
	Relationship Name
	blng__BillingFinanceBook__r
	Relationship Type
	Lookup
	Refers To
	blng__FinanceBook__c
blng__BillingFinancePeriod__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The invoice line's transactions report to a finance period within the finance book. This is a relationship field.

Field	Details

Relationship Name
blng__BillingFinancePeriod__r

Relationship Type
Lookup

Refers To
blng__FinancePeriod__c

---

blng__BillingFrequency__c

Type
picklist

Properties
Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Description
Indicates how frequently Salesforce Billing invoices the invoice line's originating order product. Valid values are:

- Annual
- Invoice Plan
- Monthly
- Quarterly
- Semiannual

---

blng__BillingGLRule__c

Type
reference

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
Defines the general ledger (GL) treatment for this invoice line. This is a relationship field.

Relationship Name
blng__BillingGLRule__r

Relationship Type
Lookup

Refers To
blng__GLRule__c

---

blng__BillingGLTreatment__c

Type
reference

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
Defines the debit and credit GL accounts that record this invoice line. This is a relationship field.

Field	Details
	Relationship Name blng__BillingGLTreatment__r
	Relationship Type Lookup
	Refers To blng__GLTreatment__c
blng__BillingRule__c	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The lookup to the billing rule used to generate this invoice line. This is a relationship field.
	Relationship Name blng__BillingRule__r
	Relationship Type Lookup
	Refers To blng__BillingRule__c
blng__BillingTransaction__c	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The billing transaction associated with the invoice line. This is a relationship field.
	Relationship Name blng__BillingTransaction__r
	Relationship Type Lookup
	Refers To blng__BillingTransaction__c
blng__BillingTreatment__c	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The lookup to the billing treatment used to generate this invoice line. This is a relationship field.

Field	Details
	Relationship Name blng__BillingTreatment__r **Relationship Type** Lookup **Refers To** blng__BillingTreatment__c
blng__CalculatedQuantity__c	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The number of billing periods charged on the invoice line.
blng__ChargeDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Specifies the transaction date of the invoice line.
blng__ChargeType__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Specifies if the invoice line is recurring, one-time, or usage (legacy). Possible values are: • One-Time • Recurring • Usage
blng__ConvertedCreditNoteLine__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** For converted negative invoice lines, the credit note line is created as part of the conversion. This is a relationship field. **Relationship Name** blng__ConvertedCreditNoteLine__r

Field	Details
	Relationship Type Lookup **Refers To** blng__CreditNoteLine__c
`blng__Credits__c`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of all credit note lines allocated to this invoice line. This is a calculated field.
`blng__DaysPastDue__c`	**Type** double **Properties** Filter, Nillable, Sort **Description** The number of days past the invoice line due date. This is a calculated field.
`blng__DebitPayments__c`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The net amount of all payments made to debit note allocations on this invoice line. The debit payments equals the debits (payment allocations) – debits (payment unallocations). This is a calculated field.
`blng__DebitsPaymentAllocations__c`	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The sum of all payments made against the debit note allocated to this invoice line.
`blng__DebitsPaymentUnallocations__c`	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The sum of all payments unallocated from debit notes allocated to this invoice line.

Field	Details
blng__Debits__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of all debit note lines allocated to this invoice line. This is a calculated field.
blng__DueDate__c	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The date by which customers must pay their invoice line. This is a calculated field.
blng__EndDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The end date of the billed period for this invoice line. This value can impact the revenue.
blng__GroupedQuantity__c	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** Salesforce Billing creates invoice lines for different groups of order products based on several order and order product fields.
blng__ImpactAmount__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The change to net receivables as a result of this credit note line. Positive credit note lines have a negative impact amount because they reduce the overall amount owed to the company. This is a calculated field.
blng__InvoiceLineState__c	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details

Description

Specifies if the invoice line is marked. Valid values are:

- `Merged`— The invoice line contains invoice sublines.
- `Regular`— The invoice line is standard and doesn't include sublines.

The default value is `Regular`.

---

blng__InvoiceLineStatus__c

Type
picklist

Properties
Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Description
The status of the invoice line and is inherited from the invoice's status. Valid values are:

- `Cancelled`
- `Draft`
- `Error`
- `Initiated`
- `Post In progress`
- `Posted`
- `Rebilled`

The default value is `Draft`.

---

blng__InvoiceRun__c

Type
reference

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The invoice run that created this invoice line. This is a relationship field.

Relationship Name
blng__InvoiceRun__r

Relationship Type
Lookup

Refers To
blng__InvoiceRun__c

---

blng__Invoice__c

Type
reference

Properties
Create, Filter, Group, Sort

Description
The parent invoice for this invoice line. This is a relationship field.

Field	Details
	Relationship Name blng__Invoice__r
	Relationship Type Master-detail
	Refers To blng__Invoice__c (the master object)
blng__LegalEntityReference__c	**Type** string
	Properties Create, Filter, Group, idLookup, Nillable, Sort, Update
	Description
blng__LegalEntity__c	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The lookup to this invoice line's legal entity. Legal entities control which treatments are associated with this invoice line. This is a relationship field.
	Relationship Name blng__LegalEntity__r
	Relationship Type Lookup
	Refers To blng__LegalEntity__c
blng__NegativeLineConversion__c	**Type** currency
	Properties Create, Defaulted on create, Filter, Nillable, Sort, Update
	Description The total amount of this invoice that's converted to a credit note line. Only negative invoice lines are converted.
blng__NetAllocationsAgainstInvoiceLines__c	**Type** currency
	Properties Filter, Nillable, Sort

Field	Details
	Description The net amount of all payments made against this invoice line, which equals (allocations against invoice lines) − (unallocations against invoice lines). This is a calculated field.
blng__Notes__c	**Type** textarea **Properties** Create, Nillable, Update **Description** Reference notes that users can add. Notes fields don't have associated system logic.
blng__OrderProduct__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The order product that generated the invoice line. This is a relationship field. **Relationship Name** blng__OrderProduct__r **Relationship Type** Lookup **Refers To** OrderItem
blng__OverrideInitialRevenueEndDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** This field enables users to override the revenue period's configured end date when creating revenue schedules.
blng__OverrideInitialRevenueStartDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** This field enables users to override the revenue period's configured start date when creating revenue schedules.
blng__PaymentsAgainstDebitNoteLines__c	**Type** currency

Field	Details
	Properties Filter, Nillable, Sort **Description** The sum of payments made to debit notes allocated against the invoice and its invoice lines. This is a calculated field.
blng__PaymentsAgainstInvoiceLines__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total number of payment allocations made against the invoice lines. This is a calculated field.
blng__Payments__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The net amount of payments against this invoice line and any of its allocated debit note lines, which equals the (amount allocated against invoice line) + (amount allocated against debit note lines) − (amount unallocated against invoice lines) − (amount unallocated against debit note lines). This is a calculated field.
blng__Product__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Product that created the order product being invoiced. This is a relationship field. **Relationship Name** blng__Product__r **Relationship Type** Lookup **Refers To** Product2
blng__Quantity__c	**Type** double **Properties** Create, Filter, Nillable, Sort, Update

Field	Details
	Description Quantity invoiced from the order product.
blng__RefundsAgainstDebitNoteLines__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** This field is a calculated field.
blng__RefundsAgainstInvoiceLines__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** This field is a calculated field.
blng__RevenueExpectedAmount__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** This user-defined field is the expected revenue impact of this credit note line, if different from the credit note line amount. The revenue expected amount can: • Differ from the subtotal and total amount (with tax). • Be configured to create revenue schedules for this credit note line using this field as the amount on the revenue schedule.
blng__RevenueLiabilityAmount__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** This user-defined field is the revenue liability associated with this credit note line. The revenue liability amount can: • Differ from the subtotal and total amount (with tax). • Be configured to create revenue schedules for this credit note line using this field as the amount on the revenue schedule.
blng__RevenueMostLikelyAmount__c	**Type** currency

Field	Details
	Properties Create, Filter, Nillable, Sort, Update
	Description This user-defined field is the most likely revenue impact of this credit note line for revenue calculations when the revenue amount is uncertain. This field can differ from the subtotal and total amount (with tax).
blng__RevenueScheduleStatus__c	**Type** picklist
	Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description Specifies the status of the revenue schedule creation for this invoice line. Valid values are: • `Complete` • `Error` • `Queued`
blng__StartDate__c	**Type** date
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The start date of the billed period for this invoice line. This value can impact the revenue. See Salesforce Help for details on revenue recognition.
blng__Subtotal__c	**Type** currency
	Properties Create, Filter, Nillable, Sort, Update
	Description The total amount to settle on this invoice line, excluding tax.
blng__TaxAmount__c	**Type** currency
	Properties Create, Filter, Nillable, Sort, Update
	Description The total tax owed on this invoice line.
blng__TaxCity__c	**Type** string

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** The city used during tax calculation. The invoice's tax address fields are used only if the invoice line's tax address fields are blank.
blng__TaxCountry__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The country used during tax calculation. The invoice's tax address fields are used only if the invoice line's tax address fields are blank.
blng__TaxCounty__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The county used during tax calculation. The invoice's tax address fields are used only if the invoice line's tax address fields are blank.
blng__TaxErrorMessage__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Descriptions of errors that occur during tax calculation.
blng__TaxError__c	**Type** double **Properties** Filter, Nillable, Sort **Description** Specifies a problem with the tax details associated with the invoice line. This is a calculated field.
blng__TaxGLRule__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description
	The lookup to the GL rule that defines the GL treatment of the tax for this invoice line. This is a relationship field.
	Relationship Name
	blng__TaxGLRule__r
	Relationship Type
	Lookup
	Refers To
	blng__GLRule__c
`blng__TaxGLTreatment__c`	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The lookup to the GL treatment that defines the debit and credit GL accounts used to record the tax for this invoice line. This is a relationship field.
	Relationship Name
	blng__TaxGLTreatment__r
	Relationship Type
	Lookup
	Refers To
	blng__GLTreatment__c
`blng__TaxLocationCode__c`	**Type**
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The location code used during tax calculation. The invoice's tax address fields are used only if the invoice line's tax address fields are blank.
`blng__TaxPercentageApplied__c`	**Type**
	percent
	Properties
	Create, Filter, Nillable, Sort, Update
	Description
	The total tax percentage applied to the invoice line and is null if the line is nontaxable.
`blng__TaxRule__c`	**Type**
	reference

Field	Details

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
Lookup to the tax rule that defines the tax engine and tax data, such as address and date, to use when calculating tax for this invoice line. This is a relationship field.

Relationship Name
blng__TaxRule__r

Relationship Type
Lookup

Refers To
blng__TaxRule__c

---

blng__TaxState__c

Type
string

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The state used during tax calculation. The invoice's tax address fields are used only if the invoice line's tax address fields are blank.

---

blng__TaxStatus__c

Type
picklist

Properties
Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Description
The status of the tax calculation for this invoice line. Valid values are:

- Completed
- Copied
- Error
- Processing
- Queued
- Warning

---

blng__TaxStreet2__c

Type
string

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The street address used during tax calculation. The invoice's tax address fields are used only if the invoice line's tax address fields are blank.

Field	Details
`blng__TaxStreet__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The street address used during tax calculation. The invoice's tax address fields are used only if the invoice line's tax address fields are blank.
`blng__TaxTreatment__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Lookup to the tax treatment used to generate this invoice line. This is a relationship field. **Relationship Name** blng__TaxTreatment__r **Relationship Type** Lookup **Refers To** blng__TaxTreatment__c
`blng__TaxZipCode__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The destination ZIP code or postal code used during tax calculation. The invoice's tax address fields are used only if the invoice line's tax address fields are blank.
`blng__TotalAmount__c`	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The subtotal of the invoice line plus tax.
`blng__UniqueId__c`	**Type** string **Properties** Create, Filter, Group, idLookup, Nillable, Sort, Update

Field	Details
	Description This field is system-generated and not updated or changed by users.
blng__UnitPrice__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The price of the order product.
blng__UsageSummary__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Lookup to the usage summary that was used to create the invoice line and is blank unless the usage summary is the source for the invoice line. This is a relationship field. **Relationship Name** blng__UsageSummary__r **Relationship Type** Lookup **Refers To** blng__UsageSummary__c

blng__InvoicePlan__c

Provides billing schedules with customizable billing frequencies. The invoice plan allows for non-standard billing intervals, skipping traditional monthly or quarterly cycles. For instance, the first payment occurs ten days after the sale, with the second payment two months later.

🛑 **Important:** Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with invoice plan when multiple currencies are enabled.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp of when the current user had last accessed the invoice plan, or a record or list view related to the invoice plan.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when a user last viewed this record, a record related to this record, or a list view. If this value is null, it's possible the user accessed this record or list view (`LastReferencedDate`) but didn't view it.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the invoice plan.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The user who owns the invoice plan record. This field is a polymorphic relationship field. **Relationship Name** Owner

Field	Details
	Relationship Type Lookup **Refers To** Group, User
`blng__Active__c`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the invoice plan is active (`true`) or not (`false`). The default value is `true`.
`blng__BillingScheduleCreation__c`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Billing schedules use predetermined dates to define when and how Salesforce Billing invoices an order product. An invoice plan can be created with `BillingScheduleCreation` set to `Order Product Activation` or `Manual`. When `BillingScheduleCreation` is set to `Order Product Activation`, billing schedules and billing transactions are generated automatically. Valid values are: • `Manual` • `Order Product Activation`

blng__InvoicePlanLine__c

Represents the custom behavior for a single invoice transaction. An invoice plan contains the invoice plan lines.

🛑 Important: Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with invoice plan line when multiple currencies are enabled.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the invoice plan line.
blng__CommencementDateOffsetUnits__c	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. The unit of time that defines the offset applied to the commencement date. Valid values are: • Days • Months • Years
blng__CommencementDateOffset__c	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The number by which the invoice start date is adjusted to have an invoice plan line start date.
blng__CommencementDate__c	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update

Field	Details
	Description Required. The start date of the invoice plan line cycle. The default value is `Order Product Activation`.
`blng__FlatAmount__c`	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** A fixed amount billed each installment.
`blng__InvoicePlan__c`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Required. The invoice plan related to the invoice plan line. This field is a relationship field. **Relationship Name** blng__InvoicePlan__r **Relationship Type** Master-detail **Refers To** blng__InvoicePlan__c (the master object)
`blng__Percentage__c`	**Type** percent **Properties** Create, Filter, Nillable, Sort, Update **Description** The percentage of total amount billed each installment.
`blng__ProcessingOrder__c`	**Type** double **Properties** Create, Filter, Sort, Update **Description** Required. The process involved in fulfilling an order.
`blng__ServicePeriodEnd__c`	**Type** picklist

Field	Details
	Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Specifies if the end date of the service is mentioned on the invoice plan line. Valid values are: • `Create without Date` • `Order Product End Date`
`blng__ServicePeriodStart__c`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Specifies if the start date of the service is mentioned on the invoice plan line. Valid values are: • `Create without Date` • `Order Product Activation` • `Order Product Creation`
`blng__Type__c`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. The type of amount displayed in the invoice plan line. Valid values are: • `Flat Amount` • `Percent` • `Remainder`

blng__InvoiceRun__c

During an invoice batch run, all billing schedules that meet the specified criteria are processed, resulting in the generation of invoices.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the invoice run when multiple currencies are enabled.
LastActivityDate	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The timestamp of when a user had last associated a task or event action with the invoice run.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the invoice run was last modified.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when a user last viewed this record, a record related to this record, or a list view. If this value is null, it's possible the user accessed this record or list view (`LastReferencedDate`) but didn't view it.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the invoice run.

Field	Details
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** A system-generated field that specifies the ID of the user who created the BillingBatchScheduler record. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User
blng__ApexJobId__c	**Type** string **Properties** Create, Filter, Group, idLookup, Nillable, Sort, Update **Description** A unique job ID number that specifies the Apex job queue line status of the invoice run.
blng__CleanUpStatus__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Clean-up invoice runs to correct any system errors, and reset order products and usage summaries. Valid values are: • Completed • Completed with errors • Started
blng__CompletedTime__c	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The timestamp when the invoice run finished processing.
blng__InvoiceDate__c	**Type** date

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** The invoice date stamp on the invoice run.
blng__InvoiceScheduler__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** A scheduled processing job triggered by the billing batch scheduler. This is a relationship field. **Relationship Name** blng__InvoiceScheduler__r **Relationship Type** Lookup **Refers To** blng__InvoiceScheduler__c
blng__Notes__c	**Type** textarea **Properties** Create, Nillable, Update **Description** Optional user-defined information about the scheduler.
blng__PercentageComplete__c	**Type** percent **Properties** Create, Filter, Nillable, Sort, Update **Description** The percentage of completion of the invoice run.
blng__ProgressBar__c	**Type** string **Properties** Filter, Nillable, Sort **Description** A horizontal progress bar displaying the completion status of the invoice run. This field is a calculated field.

Field	Details
blng__StartTime__c	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The timestamp of when the invoice run started processing.
blng__StatusIndicator__c	**Type** string **Properties** Filter, Nillable, Sort **Description** The color-coded visual status of the invoice run. This is a calculated field.
blng__Status__c	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The state of the invoice run. Valid values are: • Completed • Completed with errors • Started • Canceled • Failed • Stopped The default value is Started.
blng__TargetDate__c	**Type** date **Properties** Create, Filter, Group, Sort, Update **Description** Billing schedules having the next billing date before this date is picked up for invoicing.

blng__InvoiceScheduler__c

Represents whether and when an order product gets invoiced.

> ⛔ **Important:** Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the invoice scheduler when multiple currencies are enabled.
LastActivityDate	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The timestamp of when a user had last associated a task or event action with the invoice scheduler.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp of when the current user had last accessed the invoice scheduler, or a record or list view related to the invoice scheduler.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort The timestamp when a user last viewed this record, a record related to this record, or a list view. If this value is null, it's possible the user accessed this record or list view (`LastReferencedDate`) but didn't view it.

Field	Details
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the invoice scheduler.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the record owner. This is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User
blng__ApexJobId__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** A unique job ID number that specifies the Apex job queue line status of the invoice scheduler.
blng__AutomaticallyPostInvoices__c	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the invoices must be posted automatically (`true`) or not (`false`). By default, the invoices are generated in draft state.
blng__BillUsageCharges__c	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update

Field	Details
	Description Indicates whether the charge type of the order product is usage (`true`) or not (`false`). The default value is `true`.
`blng__EndOfMonth__c`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the monthly invoice processing must start on the last day of the month (`true`) or not (`false`). If this field is set to `true`, the scheduler start date also maps to the monthly invoice processing day. The default value is `false`.
`blng__IncludeAllCurrencies__c`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether all the available invoice currencies are included for processing (`true`) or not (`false`). The default value is `false`.
`blng__InvoiceBatches__c`	**Type** multipicklist **Properties** Create, Filter, Nillable, Restricted picklist, Update **Description** The billing batches included for invoice processing.
`blng__InvoiceDate__c`	**Type** date **Properties** Create, Defaulted on create, Filter, Group, Nillable, Sort, Update **Description** The date the invoice was created.
`blng__InvoiceDayOfMonth__c`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The start day of the invoice runs each month for monthly invoice runs.

Field	Details
blng__InvoiceDayOfWeek__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The start day of the invoice runs each week for weekly invoice runs.
blng__Notes__c	**Type** textarea **Properties** Create, Nillable, Update **Description** Reference notes that users can add. Notes fields don't have associated system logic.
blng__StartDateTime__c	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The start date and time of the invoice scheduler.
blng__TargetDate__c	**Type** date **Properties** Create, Defaulted on create, Filter, Group, Nillable, Sort, Update **Description** The next billing date. Active order products, usage summaries, or billing transactions with billing schedules having the next billing date before this date is picked up for invoicing.
blng__TargetDayOfMonth__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The day of the month associated with the next billing date. Active order products, usage summaries, or billing transactions with billing schedules having the next billing date before this date is picked up for invoicing.
blng__TargetDayOfWeek__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details
	Description The day of the week associated with the next billing date. Active order products, usage summaries, or billing transactions with billing schedules having the next billing date before this date is picked up for invoicing.
blng__Type__c	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The frequency of the invoice scheduler run. Valid values are: • `Daily` • `Monthly` • `Once` • `Weekly`

blng__LegalEntity__c

Represents the way an organization is structured. An organization can be a single legal entity or it can comprise more than one legal entity.

🛑 Important: Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the legacy entity when multiple currencies are enabled.

Field	Details
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed a record related to this record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed this record. If this value is null, it's possible that this record was referenced (`LastReferencedDate`) and not viewed.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the legal entity.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the record owner. This is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User
blng__Active__c	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update

Field	Details
	Description Indicates whether the legal entity is active (true) or not (false). To deactivate the legal entity, clear the `Active` checkbox. This setting won't affect past or existing transactions or assignments. The default value is `true`.
`blng__City__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The city of the company that this legal entity represents.
`blng__Country__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The country of the company that this legal entity represents.
`blng__County__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The county of the company that this legal entity represents.
`blng__LegalEntityNumber__c`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** A system generated unique number for a legal entity.
`blng__Notes__c`	**Type** textarea **Properties** Create, Nillable, Update **Description** Specifies any additional notes and information from a user.
`blng__State__c`	**Type** string

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The state of the company that this legal entity represents.
`blng__Street1__c`	**Type** string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The primary street of the company that this legal entity represents.
`blng__Street2__c`	**Type** string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description Any additional street details of the company that this legal entity represents.
`blng__ZipPostalCode__c`	**Type** string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The ZIP code or postal code of the company that this legal entity represents.

blng__Payment__c

Represents a single event when a shopper makes a payment. For credit cards, this event is a payment capture or payment sale, but it doesn't appear on the shopper's credit card statement.

🛑 Important: Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Three-letter ISO 4217 currency code associated with the payment record.
LastActivityDate	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The timestamp of when a user had last associated a task or event action with this record.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a payment record indirectly, for example, through a list view or related record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a payment record. If this value is null, it's possible that the user only accessed the payment record or a related list view (`LastReferencedDate`), but not viewed the payment record itself.
Name	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. The name of the payment record.
blng__Account__c	**Type** reference

Field	Details
	Properties Create, Filter, Group, Sort, Update **Description** Required. The account of the customer who made the payment. This field is a relationship field. **Relationship Name** blng__Account__r **Relationship Type** Master-detail **Refers To** Account (the master object)
blng__AllocatedtoDebitNoteLines__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The assigned payments to the debit note lines within your account. This field is a calculated field.
blng__AllocatedtoInvoiceLines__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The assigned payments to the invoice lines within an account. This field is a calculated field.
blng__AllocatedtoInvoices__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The assigned payments to the invoice within an account. This field is a calculated field.
blng__AllocationStatus__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details
	Description
	Indicates whether a payment has been allocated to invoices, invoice lines, and debit note lines. A status of `Fail` means the payment has not yet been allocated.
`blng__Allocations__c`	**Type**
	currency
	Properties
	Filter, Nillable, Sort
	Description
	The total sum of payments allocated to invoices, invoice lines, and debit note lines.
	This field is a calculated field.
`blng__Amount__c`	**Type**
	currency
	Properties
	Create, Filter, Sort, Update
	Description
	Required. The amount to be debited or captured.
`blng__Balance__c`	**Type**
	currency
	Properties
	Filter, Nillable, Sort
	Description
	The amount remaining after deducting the net applied amount from total amount.
	This field is a calculated field.
`blng__BankAccountName__c`	**Type**
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The bank account name of the customer who made the payment.
`blng__BankAccountNumber__c`	**Type**
	encryptedstring
	Properties
	Create, Nillable, Update
	Description
	The bank account number of the customer who made the payment.

Field	Details
blng__BankAccountType__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The bank account type of the customer who made the payment. Valid values are: • `Business Checking` • `Checking` • `Savings`
blng__BankName__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The bank name of the customer who made the payment.
blng__BankRoutingCode__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The valid routing number of the bank of the customer, also called as the American Bankers Association (ABA) code.
blng__BaseCurrencyAmount__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The amount of the payment in the base currency of the organization. It's used to standardize the amount across different currencies.
blng__BaseCurrencyFXDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date on which the Foreign Exchange (FX) rate is applied to convert an amount into the base currency.

Field	Details
blng__BaseCurrencyFXRate__c	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The FX rate used to convert an amount from a foreign currency to the base currency.
blng__BaseCurrency__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The primary currency in which the financial transactions are recorded and reported.
blng__BillingFinanceBook__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The financial ledger that tracks all transactions related to billing, including the allocation of payments to debit note lines. It ensures that all financial activities are accurately recorded and reported. This field is a relationship field. **Relationship Name** blng__BillingFinanceBook__r **Relationship Type** Lookup **Refers To** blng__FinanceBook__c
blng__BillingFinancePeriod__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** A specific time frame for financial reporting and accounting purposes. This field is a relationship field. **Relationship Name** blng__BillingFinancePeriod__r **Relationship Type** Lookup

Field	Details
	Refers To blng__FinancePeriod__c
blng__CardCodeResponse__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The response code from the card issuer. This code indicates the result of the authorization request for a credit card transaction.
blng__CardExpirationMonth__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The expiration month of a credit or debit card in the billing information. This information is sent from the payment gateway following a transaction request.
blng__CardExpirationYear__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The expiration year of a credit or debit card in the billing information. This information is sent from the payment gateway following a transaction request.
blng__CardType__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The credit or debit card type in the billing information. Valid values are: • American Express • Discover • Mastercard • Visa
blng__CheckDate__c	**Type** date

Field	Details
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The date on which a check payment was issued or received.
`blng__CheckNumber__c`	**Type**
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The check number issued in an external system for a refund.
`blng__FinancePeriod__c`	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	A finance period stores lookups to all the accounting or revenue transactions associated with the payment that occurred within a period of time. This field is a relationship field.
	Relationship Name
	blng__FinancePeriod__r
	Relationship Type
	Lookup
	Refers To
	blng__FinancePeriod__c
`blng__GLRule__c`	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The accounting treatment for financial transactions, including which general ledger accounts to debit and credit.
	This field is a relationship field.
	Relationship Name
	blng__GLRule__r
	Relationship Type
	Lookup
	Refers To
	blng__GLRule__c

Field	Details
`blng__GLTreatment__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The general ledger treatment defines how a financial transaction is recorded in the general ledger, specifying the accounts and the nature of the entries. This field is a relationship field. **Relationship Name** blng__GLTreatment__r **Relationship Type** Lookup **Refers To** blng__GLTreatment__c
`blng__GatewayID__c`	**Type** string **Properties** Create, Filter, Group, idLookup, Nillable, Sort, Update **Description** A unique transaction ID created by the payment gateway.
`blng__ImpactAmount__c`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The payment's financial impact against the customer's accounts receivable. This field is a calculated field.
`blng__Invoice__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The invoice to which the payment has been applied or unapplied. This field is a relationship field. **Relationship Name** blng__Invoice__r **Relationship Type** Lookup

Field	Details
	Refers To blng__Invoice__c
blng__LegalEntity__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The legal entity used to associate the payment with other transactional records in Salesforce Billing. This field is a relationship field. **Relationship Name** blng__LegalEntity__r **Relationship Type** Lookup **Refers To** blng__LegalEntity__c
blng__Notes__c	**Type** textarea **Properties** Create, Nillable, Update **Description** Additional information or comments related to the payment.
blng__PaidBy__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The customer who made a payment, the payment amount, and other details. This field is a relationship field. **Relationship Name** blng__PaidBy__r **Relationship Type** Lookup **Refers To** Contact
blng__PaymentAuthorization__c	**Type** reference

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** The authorization record for this payment. If there's a delayed capture (when the capture occurs after the authorization), all captures must be made against a previously successful authorization transaction. This field is a relationship field. **Relationship Name** blng__PaymentAuthorization__r **Relationship Type** Lookup **Refers To** blng__PaymentAuthorization__c
blng__PaymentDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date on which the payment was made.
blng__PaymentDescription__c	**Type** textarea **Properties** Create, Nillable, Update **Description** The additional details about the payment.
blng__PaymentGatewayToken__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** A unique string of numbers that's used to process transactions without exposing the actual payment information.
blng__PaymentGateway__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description
	ID of the payment gateway that processed the payment. If there's a delayed payment, the field is populated from the authorization record.
	This field is a relationship field.
	Relationship Name
	blng__PaymentGateway__r
	Relationship Type
	Lookup
	Refers To
	blng__PaymentGateway__c
`blng__PaymentMethod__c`	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The payment method associated with the payment.
	This field is a relationship field.
	Relationship Name
	blng__PaymentMethod__r
	Relationship Type
	Lookup
	Refers To
	blng__PaymentMethod__c
`blng__PaymentMode__c`	**Type**
	picklist
	Properties
	Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	The mode of payment. Select `Electronic` if the payments are processed through Salesforce Billing and `External` to manually enter a payment that was processed outside of Salesforce.
	Valid values are:
	• `Electronic`
	• `External`
`blng__PaymentType__c`	**Type**
	picklist
	Properties
	Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details
	Description The payment type for a payment. Valid values are: • ACH • Bank Transfer • Cash • Check • Credit Card
blng__RefundsAllocations__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total refund that has been allocated against this payment. This field is a calculated field.
blng__RefundsUnallocations__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total number of a previously applied refund that has since been unapplied from this payment. This field is a calculated field.
blng__Refunds__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** A refund that's applied to a payment. This field is a calculated field.
blng__RequestCreditCardNumber__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** A tokenized version of the credit card number sent to the customer bank within the transaction request. Salesforce Billing doesn't store credit card numbers or other sensitive customer payment information. Instead, it stores tokenized versions of that information and sends the tokens to the external payment gateway.

Field	Details
blng__RequestTypeOfPayment__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The type of payment action requested of the customer bank: authorization, charge, void, or refund.
blng__Status__c	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The status of the payment transaction. Valid values are: • `Cancelled`—The payment has been voided and can no longer be allocated. • `Draft`—The payment can be edited before posting and allocating it to a target. • `Processed`—The payment has been finalized and can be allocated against a target. • `Failed`—The authorization for the payment failed. The default value is `Draft`.
blng__Transaction__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The financial activities or operations that are recorded as part of a payment. This field is a relationship field. **Relationship Name** blng__Transaction__r **Relationship Type** Lookup **Refers To** blng__PaymentTransaction__c
blng__UnallocatedfromDebitNoteLines__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount from debit note lines that has not been allocated to any specific payment.

Field	Details
	This field is a calculated field.
blng__UnallocatedfromInvoiceLines__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount from invoice lines that has not been allocated to any specific payment. This field is a calculated field.
blng__UnallocatedfromInvoices__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount from invoice that has not been allocated to any specific payment. This field is a calculated field.
blng__Unallocations__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total unallocated amounts in a payment. This field is a calculated field.

blng__PaymentAllocationDebitNoteLine__c

Represents an individual line item on a debit note allocated to a payment, used to track and manage payment allocations against debit notes for maintaining accurate financial records.

🛇 Important: Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the payment allocation debit note line when multiple currencies are enabled.
Name	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. The name of the payment allocation debit note line.
blng__Amount__c	**Type** currency **Properties** Create, Filter, Sort, Update **Description** Required. The value allocated to a specific debit note line from a payment. This value represents the portion of the payment that's applied to adjust the balance indicated by the debit note line.
blng__BillingFinanceBook__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The financial ledger that tracks all transactions related to billing, including the allocation of payments to debit note lines. This value ensures that all financial activities are accurately recorded and reported. This field is a relationship field. **Relationship Name** blng__BillingFinanceBook__r **Refers To** blng__FinanceBook__c
blng__BillingFinancePeriod__c	**Type** reference

Field	Details

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The specific time frame during which the allocation of payments to debit note lines is recorded. This period helps in organizing financial data for reporting and analysis, such as monthly or quarterly financial statements. This field is a relationship field.

Relationship Name
blng__BillingFinancePeriod__r

Refers To
blng__FinancePeriod__c

`blng__DebitNoteAllocationId__c`

Type
string

Properties
Create, Filter, Group, idLookup, Nillable, Sort, Update

Description
A unique identifier for the allocation of a payment to a specific debit note line. This ID helps in tracking and managing the relationship between the payment and the debit note line it's applied to.

`blng__DebitNoteLine__c`

Type
reference

Properties
Create, Filter, Group, Sort

Description
Required. An individual entry on a debit note that specifies the amount and details of an adjustment. In the context of payment allocation, this entry represents the specific line item to which a portion of the payment is applied. This field is a relationship field.

Relationship Name
blng__DebitNoteLine__r

Relationship Type
Master-detail

Refers To
blng__DebitNoteLine__c (the detail object)

`blng__ImpactAmount__c`

Type
currency

Properties
Filter, Nillable, Sort

Field	Details
	Description The financial effect of allocating a payment to a debit note line. This value indicates how much the allocation reduces the outstanding balance of the debit note line. This field is a calculated field.
blng__InvoiceLine__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** An invoice line is an individual entry on an invoice detailing specific charges. This field ensures the payments are correctly applied to related debit note lines. This field is a relationship field. **Relationship Name** blng__InvoiceLine__r **Refers To** blng__InvoiceLine__c
blng__LegalEntity__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The entity responsible for issuing the debit note and receiving the payment. This field is a relationship field. **Relationship Name** blng__LegalEntity__r **Refers To** blng__LegalEntity__c
blng__Notes__c	**Type** textarea **Properties** Create, Nillable, Update **Description** Additional details or comments related to the allocation of a payment to a debit note line.
blng__Payment__c	**Type** reference **Properties** Create, Filter, Group, Sort

Field	Details
	Description Required. The funds allocated to adjust the balance specified by the debit note. This field is a relationship field. **Relationship Name** blng__Payment__r **Relationship Type** Master-detail **Refers To** blng__Payment__c (the master object)
`blng__Type__c`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The classification of the financial transaction. Valid values are: • `Allocation` • `Unallocation`
`blng__Unallocated__c`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. Funds or amounts from a payment that are yet to be assigned to a specific debit note line. These amounts are pending allocation to ensure accurate financial tracking and reconciliation. The default value is `false`.

blng__PaymentAllocationInvoice__c

Represents the allocation of payments to specific invoices. This object is used to track and manage how payments are applied to outstanding invoices, ensuring accurate financial records and reporting.

🛑 **Important:** Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the payment allocation invoice when multiple currencies are enabled.
Name	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. The name of the payment allocation invoice. A name is a unique value that helps in identifying the specific allocation record.
blng__Amount__c	**Type** currency **Properties** Create, Filter, Sort, Update **Description** Required. The total amount allocated to the invoice. The amount represents the value that has been applied to the invoice from the payment.
blng__BillingFinanceBook__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The financial book associated with the billing. This field helps in categorizing and managing financial records according to different finance books. This field is a relationship field. **Relationship Name** blng__BillingFinanceBook__r **Refers To** blng__FinanceBook__c
blng__BillingFinancePeriod__c	**Type** reference

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** The financial period during which the billing and payment allocation occur. This helps in tracking and reporting financial activities within a specific time frame. This field is a relationship field. **Relationship Name** blng__BillingFinancePeriod__r **Refers To** blng__FinancePeriod__c
blng__ImpactAmount__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount that impacts the invoice balance. The impact amount could be the portion of the payment that directly affects the outstanding balance of the invoice. This field is a calculated field.
blng__Invoice__c	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Required. The reference to the specific invoice that's being allocated a payment. This links the payment allocation to the corresponding invoice record. This field is a relationship field. **Relationship Name** blng__Invoice__r **Relationship Type** Master-detail **Refers To** blng__Invoice__c (the detail object)
blng__Notes__c	**Type** textarea **Properties** Create, Nillable, Update **Description** Additional details or comments related to the payment allocation invoice.

Field	Details
blng__Payment__c	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Required. The payment that's being allocated to the invoice. This field is a relationship field. **Relationship Name** blng__Payment__r **Relationship Type** Master-detail **Refers To** blng__Payment__c (the master object)
blng__Type__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The type of payment allocation. Valid values are: • Allocation • Unallocation
blng__Unallocated__c	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. The amount of the payment that remains unallocated. The default value is false.

blng__PaymentAllocationInvoiceLine__c

Represents the allocation of a payment to a specific invoice line. This object helps in tracking how payments are distributed across different invoice lines.

🛑 **Important:** Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the payment allocation invoice line when multiple currencies are enabled.
Name	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. The name of the payment allocation invoice line. A name is a unique value that helps in identifying the specific allocation record.
blng__Amount__c	**Type** currency **Properties** Create, Filter, Sort, Update **Description** Required. The amount of money allocated to the invoice line. This field specifies the portion of the payment applied to a specific invoice line.
blng__BillingFinanceBook__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The finance book associated with the billing. This field helps in categorizing and managing financial records according to different finance books. This field is a relationship field. **Relationship Name** blng__BillingFinanceBook__r

Field	Details
	Refers To blng__FinanceBook__c
blng__BillingFinancePeriod__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The financial period during which the billing occurs. This field is used for financial reporting and tracking purposes. This field is a relationship field. **Relationship Name** blng__BillingFinancePeriod__r **Refers To** blng__FinancePeriod__c
blng__ImpactAmount__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount that impacts the financial records due to the payment allocation. This field is used to understand the financial effect of the payment allocation. This field is a calculated field.
blng__InvoiceLine__c	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Required. The specific invoice line to which the payment is allocated. This field links the payment allocation to the corresponding invoice line item. This field is a relationship field. **Relationship Name** blng__InvoiceLine__r **Relationship Type** Master-detail **Refers To** blng__InvoiceLine__c (the master object)

Field	Details
blng__LegalEntity__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The legal entity associated with the payment allocation. This field helps in identifying the legal entity responsible for the transaction. This field is a relationship field. **Relationship Name** blng__LegalEntity__r **Refers To** blng__LegalEntity__c
blng__Notes__c	**Type** textarea **Properties** Create, Nillable, Update **Description** Additional notes or comments related to the payment allocation.
blng__Payment__c	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Required. The payment record associated with the allocation. This field is a relationship field. **Relationship Name** blng__Payment__r **Relationship Type** Master-detail **Refers To** blng__Payment__c (the detail object)
blng__Type__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The type of payment allocation. Valid values are:

Field	Details
	• Allocation • Unallocation
blng__Unallocated__c	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. The amount of the payment that remains unallocated. This field indicates how much of the payment is still available to be applied to other invoice lines. The default value is `false`.

blng__PaymentAuthorization__c

Represents the process of securing authorization from a customer's bank or credit card issuer to charge a specific amount to their account, verifying the availability of sufficient funds or credit.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the payment authorization when multiple currencies are enabled.
LastActivityDate	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The timestamp of when a user had last associated a task or event action with the payment authorization.

Field	Details
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a payment authorization record indirectly, for example, through a list view or related record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a payment authorization record. If this value is null, it's possible that the user only accessed the payment authorization record or a related list view (`LastReferencedDate`), but not viewed the payment authorization record itself.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the payment authorization. A name is a unique value that helps in identifying the specific allocation record.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. The unique identifier indicates the user or system responsible for managing or overseeing a specific payment authorization record. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
blng__Account__c	**Type** reference

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** The account of the customer who made the payment. This field is a relationship field. **Relationship Name** blng__Account__r **Refers To** Account
blng__Balance__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The total amount of the payment authorization.
blng__CurrentAmount__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The amount authorized to be debited or captured.
blng__EffectiveEndDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date when the payment authorization expires.
blng__EffectiveStartDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date when the payment authorization takes effect.
blng__GatewayReferenceDetails__c	**Type** textarea

Field	Details
	Properties Create, Nillable, Update **Description** Additional details about the payment authorization that can't be stored in other fields on the payment record.
blng__GatewayReferenceNumber__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The payment authorization transaction ID created by the payment gateway.
blng__NetPayments__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The net amount of payments processed, excluding any fees or adjustments. This field is used to track the actual amount received from a transaction after all deductions.
blng__Notes__c	**Type** textarea **Properties** Create, Nillable, Update **Description** Additional details or comments related to the payment allocation invoice.
blng__OriginalAmount__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The total amount authorized for the transaction.
blng__PaymentGateway__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The payment gateway that processed the payment authorization.

Field	Details
	This field is a relationship field.
	Relationship Name
	blng__PaymentGateway__r
	Refers To
	blng__PaymentGateway__c
blng__PaymentMethod__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The payment method used by the customer to make the payment.
	This field is a relationship field.
	Relationship Name
	blng__PaymentMethod__r
	Refers To
	blng__PaymentMethod__c
blng__Source__c	**Type**
	picklist
	Properties
	Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	The origin or source of the payment authorization. Valid values are:
	• Gateway
	• Manual
	The default value is Manual.
blng__Status__c	**Type**
	picklist
	Properties
	Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	The status of the payment transaction. Valid values are:
	• Active
	• Canceled
	• Processed
	The default value is Active.

blng__PaymentGateway__c

Represents the connection to an external payment gateway.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the payment gateway record when multiple currencies are enabled.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a payment gateway record indirectly, for example, through a list view or related record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a payment gateway record. If this value is null, it's possible that the user only accessed the payment gateway record or a related list view (LastReferencedDate), but not viewed the payment gateway record itself.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the payment gateway.

Field	Details
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. The ID of the record owner. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
adnblng__APILoginId__c	**Type** encryptedstring **Properties** Create, Nillable, Update **Description** A unique identifier to authenticate API requests. This field is used with a transaction key to ensure secure communication between the application and payment gateway.
adnblng__APITransactionKey__c	**Type** encryptedstring **Properties** Create, Nillable, Update **Description** The transaction key required for API authentication with the payment gateway.
adnblng__TestMode__c	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. Indicates whether the authorize.net merchant is operating in test mode (`true`) or not (`false`). The test mode is used to simulate transactions without processing actual payments. The default value is `false`.
blng__Active__c	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update

Field	Details

Description

Required. Indicates whether the payment gateway is active (`true`) or not (`false`). To deactivate the payment gateway, clear the `Active` checkbox. This won't affect past or existing transactions. The default value is `true`.

`blng__Default__c`

Type
boolean

Properties
Create, Defaulted on create, Filter, Group, Sort, Update

Description
Required. Indicates whether a particular payment gateway is set as the default gateway for processing payments (`true`) or not (`false`).

The default value is `false`.

`blng__GatewayType__c`

Type
picklist

Properties
Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Description
The name of the payment gateway provider.

Valid values are:

- `AuthorizeDotNet`
- `CyberSource`
- `PayFlowPro`
- `Payeezy`

`blng__Notes__c`

Type
textarea

Properties
Create, Nillable, Update

Description
Additional information or comments related to the payment gateway.

`csblng__MerchantId__c`

Type
encryptedstring

Properties
Create, Nillable, Update

Description
The ID of the merchant account used by the payment gateway. This merchant account links to a merchant account at the payment processor.

Field	Details
csblng__MerchantReference__c	**Type** encryptedstring **Properties** Create, Nillable, Update **Description** The merchant reference identifier for transactions processed through a payment gateway.
csblng__TestMode__c	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. Indicates whether the CyberSource merchant is operating in test mode (`true`) or not (`false`). The test mode is used to simulate transactions without processing actual payments. The default value is `false`.
csblng__TransactionSecurityKey__c	**Type** textarea **Properties** Create, Nillable, Update **Description** The security key used for authenticating transactions with the payment gateway.
pyzblng__APIKey__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The API key used to authenticate and authorize transactions with the payment gateway service.
pyzblng__APILoginId__c	**Type** encryptedstring **Properties** Create, Nillable, Update **Description** A unique identifier to authenticate the payeezy API requests. This field is used with a transaction key to ensure secure communication between the application and the payment gateway.

Field	Details
pyzblng__APISecret__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The API secret used along with the API key to authenticate and authorize transactions with the payment gateway service.
pyzblng__APITransactionKey__c	**Type** encryptedstring **Properties** Create, Nillable, Update **Description** Stores the payeezy transaction key required for API authentication with the payment gateway.
pyzblng__EnableUserIdentification__c	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. Indicates whether the user identification during payment processing is enabled (`true`) or not (`false`). The default value is `false`.
pyzblng__JSSecurityKey__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** A JavaScript security key used to enhance the security of client interactions with the payment gateway.
pyzblng__TestMode__c	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. Indicates whether the Payeezy merchant is operating in test mode (`true`) or not (`false`). Test mode is used to simulate transactions without processing actual payments. The default value is `false`.

Field	Details
pyzblng__Token__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Required. A token used for secure Payeezy transactions.
pyzblng__TransarmorToken__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** A token provided by the TransArmor service, which is used to securely handle and store payment card information.

blng__PaymentGatewayConfig__c

Represents the configuration for integrating Salesforce Billing with a payment gateway.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the invoice when multiple currencies are enabled.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Sort, Update

Field	Details
	Description Required. The name of the payment gateway configuration.
SetupOwnerId	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The ID of the owner of the payment gateway configuration. This field is a polymorphic relationship field.
	Relationship Name SetupOwner
	Refers To Organization, Profile, User
blng__ClassName__c	**Type** string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The Apex class name that handles the integration with a specific payment gateway.

blng__PaymentMapping__c

Represents the process of associating payment types, methods, or purposes with other information.

Supported Calls

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

Fields

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

Field	Details
	Description
	The three-letter ISO 4217 currency code associated with the payment mapping record when multiple currencies are enabled.
LastActivityDate	**Type**
	date
	Properties
	Filter, Group, Nillable, Sort
	Description
	The timestamp of when a user had last associated a task or event action with the payment mapping record.
LastReferencedDate	**Type**
	dateTime
	Properties
	Filter, Nillable, Sort
	Description
	The timestamp when the current user last accessed a payment mapping record indirectly, for example, through a list view or related record.
LastViewedDate	**Type**
	dateTime
	Properties
	Filter, Nillable, Sort
	Description
	The timestamp when the current user last viewed a payment mapping record. If this value is null, it's possible that the user only accessed the payment mapping record or a related list view (`LastReferencedDate`), but not viewed the payment mapping itself.
Name	**Type**
	string
	Properties
	Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update
	Description
	The name of the payment mapping record.
OwnerId	**Type**
	reference
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	Required. The ID of the record owner.

Field	Details
	This field is a polymorphic relationship field.
	Relationship Name Owner
	Refers To Group, User
blng__Amount__c	**Type** currency
	Properties Create, Filter, Nillable, Sort, Update
	Description The amount to be debited or captured.
blng__InvoiceLine__c	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The invoice line associated with the payment mapping record. This field is a relationship field.
	Relationship Name blng__InvoiceLine__r
	Refers To blng__InvoiceLine__c
blng__Invoice__c	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The invoice associated with the payment mapping record. This field is a relationship field.
	Relationship Name blng__Invoice__r
	Refers To blng__Invoice__c
blng__Notes__c	**Type** textarea
	Properties Create, Nillable, Update

Field	Details
	Description
	Additional information or comments related to the payment mapping record.
`blng__OrderProduct__c`	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The order product associated with the payment mapping record.
	This field is a relationship field.
	Relationship Name
	blng__OrderProduct__r
	Refers To
	OrderItem
`blng__Order__c`	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The order associated with the payment mapping record.
	This field is a relationship field.
	Relationship Name
	blng__Order__r
	Refers To
	Order
`blng__PaymentAuthorization__c`	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The authorization record for this payment mapping. If there's a delayed capture (when the capture occurs after the authorization), all captures must be made against a previously successful authorization transaction.
	This field is a relationship field.
	Relationship Name
	blng__PaymentAuthorization__r
	Refers To
	blng__PaymentAuthorization__c

Field	Details
blng__QuoteLine__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The quote line associated with the payment mapping record. This field is a relationship field. **Relationship Name** blng__QuoteLine__r **Refers To** SBQQ__QuoteLine__c
blng__Quote__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The quote associated with the payment mapping record. This field is a relationship field. **Relationship Name** blng__Quote__r **Refers To** SBQQ__Quote__c
blng__Status__c	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The status of the payment mapping. Valid values are: • Active • Canceled The default value is Active.

blng__PaymentMethod__c

Represents the storage of information about different payment methods used by customers.

224

> ⊘ **Important:** Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the payment authorization when multiple currencies are enabled.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a payment method record indirectly, for example, through a list view or related record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a payment method record. If this value is null, it's possible that the user only accessed the payment method record or a related list view (`LastReferencedDate`), but not viewed the payment method record itself.
Name	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the payment method.

225

Field	Details
`blng__Account__c`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Required. The account of the customer who made the payment. This field is a relationship field. **Relationship Name** blng__Account__r **Relationship Type** Master-detail **Refers To** Account (the master object)
`blng__Active__c`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. Indicates whether the payment method is active (`true`) or not (`false`). To deactivate the payment method, clear the `Active` checkbox. This won't affect past or existing transactions or assignments. The default value is `true`.
`blng__AutoPay__c`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. The payment method to create payments toward posted invoices on a customer account. The default value is `false`.
`blng__BankAccountName__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The name of the bank account associated with the payment method.
`blng__BankAccountNumber__c`	**Type** encryptedstring

Field	Details
	Properties Create, Nillable, Update **Description** The bank account number associated with the payment method.
blng__BankAccountType__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The type of bank account associated with the payment method. Valid values are: • `Business Checking` • `Checking` • `Savings`
blng__BankName__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The name of the bank associated with the payment method.
blng__BankRoutingCode__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The valid routing number of the bank, also known as the American Bankers Association (ABA) code.
blng__BillingAddress__c	**Type** textarea **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The billing address associated with the payment method that must match the address on file with the payment provider.
blng__BillingCity__c	**Type** string

227

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The city of the billing address associated with the payment method.
blng__BillingCompany__c	**Type** string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The company name associated with the billing address.
blng__BillingCountry__c	**Type** picklist
	Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description The country of the billing address associated with the payment method.
blng__BillingEmail__c	**Type** email
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The email address associated with the billing information.
blng__BillingFax__c	**Type** phone
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The fax details associated with the billing information.
blng__BillingFirstName__c	**Type** string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The first name provided in the billing information.
blng__BillingLastName__c	**Type** string

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The last name provided in the billing information.
blng__BillingPhone__c	**Type** phone
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The phone number provided in the billing information.
blng__BillingStateProvince__c	**Type** string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The state of the billing address associated with the payment method.
blng__BillingStreet__c	**Type** textarea
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The street of the billing address associated with the payment method.
blng__BillingZipPostal__c	**Type** string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The zip code of the billing address associated with the payment method.
blng__CVV__c	**Type** encryptedstring
	Properties Create, Nillable, Update
	Description The Card Verification Value (CVV) code linked to the credit or debit card in the billing information.

Field	Details
blng__CardBIN__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The six digit Bank Identification Number (BIN) associated with a credit or debit card in the billing information.
blng__CardExpirationMonth__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The expiration month of a credit or debit card in the billing information. This information is sent from the payment gateway following a transaction request.
blng__CardExpirationYear__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The expiration year of a credit or debit card in the billing information. This information is sent from the payment gateway following a transaction request.
blng__CardLastFour__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The last four digits of a credit or debit card number in the billing information.
blng__CardNumber__c	**Type** encryptedstring **Properties** Create, Nillable, Update **Description** The credit or debit card number in the billing information.
blng__CardType__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details
	Description
	The credit or debit card type in the billing information.
	Valid values are:
	• American Express
	• Discover
	• Mastercard
	• Visa
blng__GLRule__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The General Ledger (GL) rule associated with the payment method.
	This field is a relationship field.
	Relationship Name
	blng__GLRule__r
	Refers To
	blng__GLRule__c
blng__GLTreatment__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The GL treatment associated with the payment method.
	This field is a relationship field.
	Relationship Name
	blng__GLTreatment__r
	Refers To
	blng__GLTreatment__c
blng__GatewayResponse__c	**Type**
	textarea
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The response received from the payment gateway after processing a transaction.

Field	Details
`blng__LegalEntity__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The legal entity associated with a payment method. This field is a relationship field. **Relationship Name** blng__LegalEntity__r **Refers To** blng__LegalEntity__c
`blng__Nameoncard__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The name on the payment card.
`blng__NickName__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The nick name on the payment card.
`blng__Notes__c`	**Type** textarea **Properties** Create, Nillable, Update **Description** Additional information or comments related to the payment method.
`blng__PaymentGatewayToken__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** A token for secured payment gateway transactions.
`blng__PaymentGateway__c`	**Type** reference

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** The Salesforce Billing payment gateway record used to establish communication with the external payment gateway. This field is a relationship field. **Relationship Name** blng__PaymentGateway__r **Refers To** blng__PaymentGateway__c
blng__PaymentType__c	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The type of the payment method. Valid values are: • ACH • Credit Card
blng__StreetAddress2__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The street associated with the customer's identification address for the payment.
pyzblng__ChequeNumber__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The cheque number associated with a payment. This field is used for Payeezy gateway.
pyzblng__EncryptedCustomerId__c	**Type** encryptedstring **Properties** Create, Nillable, Update **Description** This field stores the encrypted customer data and is used for Payeezy gateway.

Field	Details
pyzblng__IdentificationAddress__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The address used to identify the customer making the payment.
pyzblng__IdentificationCity__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The city associated with the customer's identification address for the payment.
pyzblng__IdentificationCountry__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The country of the identification address of the customer making the payment.
pyzblng__IdentificationIdNumber__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The identification number to verify the customer making the payment.
pyzblng__IdentificationPhoneNumber__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The identification phone number of the customer making the payment.
pyzblng__IdentificationPhoneType__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The type of the identification phone number.

Field	Details
	Valid values are:

Valid values are:

- `Day`
- `Home`
- `Night`
- `Work`

---

pyzblng__IdentificationState__c

Type
picklist

Properties
Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Description
The state of the identification address of the customer making the payment.

---

pyzblng__IdentificationType__c

Type
picklist

Properties
Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Description
The process for verifying a customer's identity.

Valid values are:

- `Driver license`
- `Military ID`
- `Social Security Number`
- `Tax ID`

---

pyzblng__IdentificationZIP__c

Type
string

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The ZIP code associated with the customer's identification address for the payment.

---

blng__PaymentRun__c

Represents a batch processing job in Salesforce Billing. During a payment run, all payment schedules that meet the specified criteria are processed, resulting in the collection of payments.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the payment authorization when multiple currencies are enabled.
LastActivityDate	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The timestamp of when a user had last associated a task or event action with the account.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the payment run.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. The ID of the record owner. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup

Field	Details
	Refers To Group, User
blng__ApexJobIdForCleanUp__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** A unique identifier of an Apex job that's responsible for the cleanup tasks.
blng__ApexJobId__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** A unique identifier of an Apex job associated with the payment run.
blng__CleanUpStatus__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The status of the cleanup operations associated with the payment run. Valid values are: • `Completed` • `Started` • `Step 1 : Invoice update` • `Step 2 : Payment Transaction update`
blng__CompletedTime__c	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The time when the payment run was completed.
blng__FailedTransactions__c	**Type** double **Properties** Create, Defaulted on create, Filter, Nillable, Sort, Update

237

Field	Details
	Description
	The details of transactions that failed during the payment run process.
blng__InvoicesProcessed__c	**Type**
	double
	Properties
	Create, Defaulted on create, Filter, Nillable, Sort, Update
	Description
	The invoices that were successfully processed during the payment run.
blng__PaymentGateway__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The payment gateway used to process the transactions during the payment run.
	This field is a relationship field.
	Relationship Name
	blng__PaymentGateway__r
	Relationship Type
	Lookup
	Refers To
	blng__PaymentGateway__c
blng__PaymentScheduler__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The payment scheduler responsible for the payment run.
	This field is a relationship field.
	Relationship Name
	blng__PaymentScheduler__r
	Relationship Type
	Lookup
	Refers To
	blng__PaymentScheduler__c
blng__StartTime__c	**Type**
	dateTime

Field	Details
	Properties Create, Filter, Nillable, Sort, Update
	Description The start time of the payment run.
blng__Status__c	**Type** picklist
	Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description The status of the payment run. Valid values are: • Completed • Failed • Processing • Started
blng__SuccessfulTransactions__c	**Type** double
	Properties Create, Defaulted on create, Filter, Nillable, Sort, Update
	Description The number of transactions that were successfully processed during the payment run.
blng__TargetDate__c	**Type** date
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The date by which the payment run is intended to be completed.
blng__TotalPaymentProcessed__c	**Type** currency
	Properties Create, Defaulted on create, Filter, Nillable, Sort, Update
	Description The total number of payments that were successfully processed during the payment run.
blng__TransactionsGenerated__c	**Type** double

Field	Details
	Properties
	Filter, Nillable, Sort
	Description
	The number of transactions that were generated during the payment run.
	This field is a calculated field.

blng__PaymentScheduler__c

Represents automating payment runs by creating payment schedulers, which process the payment runs on a recurring basis.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type**
	picklist
	Properties
	Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	The three-letter ISO 4217 currency code associated with the payment scheduler when multiple currencies are enabled.
LastActivityDate	**Type**
	date
	Properties
	Filter, Group, Nillable, Sort
	Description
	The timestamp of when a user had last associated a task or event action with the account.
LastReferencedDate	**Type**
	dateTime
	Properties
	Filter, Nillable, Sort

Field	Details
	Description The timestamp when the current user last accessed a payment scheduler record indirectly, for example, through a list view or related record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a payment scheduler record. If this value is null, it's possible that the user only accessed the payment scheduler record or a related list view (`LastReferencedDate`), but not viewed the payment scheduler record itself.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** **Description** The name of the payment scheduler.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. The ID of the record owner. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User
blng__ApexJobId__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** A unique identifier of an Apex job associated with the payment scheduler.

Field	Details
blng__IncludeAllCurrencies__c	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. Indicates whether to include all currencies in the payment run (`true`) or not (`false`). The default value is `false`.
blng__Notes__c	**Type** textarea **Properties** Create, Nillable, Update **Description** Additional details or comments related to the payment scheduler.
blng__PaymentBatches__c	**Type** multipicklist **Properties** Create, Filter, Nillable, Restricted picklist, Update **Description** A collection of several payments combined for processing.
blng__PaymentGateway__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The Salesforce Billing payment gateway record used to establish communication with the external payment gateway. This field is a relationship field. **Relationship Name** blng__PaymentGateway__r **Relationship Type** Lookup **Refers To** blng__PaymentGateway__c
blng__PaymentType__c	**Type** multipicklist

Field	Details
	Properties Create, Filter, Nillable, Restricted picklist, Update **Description** The type of payment in the payment schedule. Valid values are: • ACH • Credit Card
blng__Startdateandtime__c	**Type** dateTime **Properties** Create, Defaulted on create, Filter, Nillable, Sort, Update **Description** The start date and time of the payment scheduler.
blng__TargetDate__c	**Type** date **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. The target date for the payment run. Billing schedules having the next billing date before this date is picked up for invoicing.
blng__TargetDayOfMonth__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** In a monthly schedule, the designated day of the month for the payment run.
blng__TargetDayofWeek__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** In a weekly schedule, the designated day of the week for the payment run.
blng__Type__c	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update

Field	Details
	Description Required. The type of payment schedule. Valid values are: • `Daily` • `Monthly` • `Once` • `Weekly`

blng__PaymentTerms__c

Represents the period customers have to pay for an invoiced order product.

Supported Calls

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

Fields

Field	Details
`CurrencyIsoCode`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the payment term when multiple currencies are enabled.
`Name`	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Sort, Update **Description** The name of the payment term.
`SetupOwnerId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The ID of the record owner. This is a polymorphic relationship field. **Relationship Name** SetupOwner **Relationship Type** Lookup **Refers To** Organization, Profile, User
blng__NumberOfDays__c	**Type** double **Properties** Create, Filter, Sort, Update **Description** The number of days by which the payment term is due.

blng__PaymentTransaction__c

Represents the payment transactions associated with a payment.

🛑 Important: Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the payment transaction record when multiple currencies are enabled.

Field	Details
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a payment transaction record indirectly, for example, through a list view or related record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a payment transaction record. If this value is null, it's possible that the user only accessed the payment transaction record or a related list view (`LastReferencedDate`), but not viewed the payment transaction record itself.
Name	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. The name of the payment transaction.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. The ID of the record owner. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User
blng__AVSResponseCode__c	**Type** string

Field	Details

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The numeric value contained in a gateway response alongside the response status, response, and response message. The code is used as an indexing value for different types of responses. You can generally ignore it and focus on the response and response message to get the details you need about the results of your transaction request.

blng__AVSZip__c

Type
string

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The numeric address and zip code used as an indexing value.

blng__Account__c

Type
reference

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The account containing the invoice that the payment transaction is set to pay.

This field is a relationship field.

Relationship Name
blng__Account__r

Relationship Type
Lookup

Refers To
Account

blng__Amount__c

Type
double

Properties
Create, Filter, Nillable, Sort, Update

Description
The amount that Salesforce Billing attempts to use against the balance of an invoice.

blng__Authorization__c

Type
string

Properties
Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The approval from a customer's bank or credit card issuer to charge a payment to their account.
blng__BankAccountName__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** A user-defined value for the bank account name. This field used to record information about externally recorded Automated Clearing House (ACH) payments.
blng__BankAccountNumber__c	**Type** encryptedstring **Properties** Create, Nillable, Update **Description** A user-defined value for the bank account number. This field used to record information about externally recorded ACH payments.
blng__BankAccountType__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** A user-defined value for the bank account type. This field used to record information about externally recorded ACH payments. Valid values are: • `Business Checking` • `Checking` • `Savings`
blng__BankName__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** A user-defined value for the bank name. This field used to record information about externally recorded ACH payments.
blng__BankRoutingCode__c	**Type** string

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** A user-defined value for the bank routing code. This field used to record information about externally recorded ACH payments.
blng__CardCodeResponse__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The response code from the card issuer. This code indicates the result of the authorization request for a credit card transaction.
blng__CardExpirationMonth__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The expiration month of a credit or debit card in the billing information. This information is sent from the payment gateway following a transaction request.
blng__CardExpirationYear__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The expiration year of a credit or debit card in the billing information. This information is sent from the payment gateway following a transaction request.
blng__CardType__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The credit or debit card type in the billing information. Valid values are: • American Express • Discover • Mastercard • Visa

249

Field	Details
blng__ExceptionMessage__c	**Type** textarea **Properties** Create, Nillable, Update **Description** The message displayed when a transaction is created without any associated payment against the invoice.
blng__GatewayDate__c	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date that the transaction request occurred in the payment gateway.
blng__GatewayID__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** A unique identification number associated with the transaction request that occurred in the payment gateway.
blng__GatewayRequest__c	**Type** textarea **Properties** Create, Nillable, Update **Description** The payment request to a payment gateway.
blng__GatewayResponse__c	**Type** textarea **Properties** Create, Nillable, Update **Description** The response from the payment gateway.
blng__GatewayStatus__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details
	Description
	The values that summarizes the payment gateway's response to the payment transaction request. Valid values are:

- `Decline`
- `Indeterminate`
- `PermanentFail`
- `RequiresReview`
- `Success`
- `SystemError`
- `ValidationError`

---

blng__IsPaymentProcessedSuccessfully__c

Type
boolean

Properties
Create, Defaulted on create, Filter, Group, Sort, Update

Description
Required. Indicates whether Salesforce Billing created a payment for a successful transaction (true) or not (false). If this field is deselected on a successful transaction, review why Salesforce Billing didn't create the payment, For example, the user who owns the payment run didn't have the permission to access payment records. After you resolve the issue, create a payment for the same amount as the payment transaction, and then allocate it to the transaction's invoice on your own.

The default value is `false`.

---

blng__Notes__c

Type
textarea

Properties
Create, Nillable, Update

Description
Additional information or comments related to the payment transaction.

---

blng__PaymentGateway__c

Type
reference

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The Salesforce Billing payment gateway record used to establish communication with the external payment gateway. This field is inherited from the payment method.

This field is a relationship field.

Relationship Name
blng__PaymentGateway__r

Field	Details
	Relationship Type Lookup
	Refers To blng__PaymentGateway__c
blng__PaymentMethod__c	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The payment method used to provide customer payment information to the payment gateway. This field is a relationship field.
	Relationship Name blng__PaymentMethod__r
	Relationship Type Lookup
	Refers To blng__PaymentMethod__c
blng__PaymentRun__c	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The payment run that created the payment transaction to record the result of external payment gateway communication. This field is a relationship field.
	Relationship Name blng__PaymentRun__r
	Relationship Type Lookup
	Refers To blng__PaymentRun__c
blng__ReasonCode__c	**Type** double
	Properties Create, Filter, Nillable, Sort, Update
	Description The reason for a particular payment transaction.

Field	Details
blng__RequestCreditCardNumber__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The credit card number associated with a payment transaction.
blng__RequestInvoice__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The invoice that Salesforce Billing is attempting to pay through the payment transaction. If the transaction is successful, Salesforce Billing creates a payment record and attempts to allocate it to the invoice. This field is a relationship field. **Relationship Name** blng__RequestInvoice__r **Relationship Type** Lookup **Refers To** blng__Invoice__c
blng__RequestTransactionType__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The type of payment action requested of the customer bank: authorization, charge, void, or refund.
blng__RequestTypeOfPayment__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The type of payment being requested.
blng__ResponseCode__c	**Type** string

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** The numeric value contained in a gateway response alongside the response status, response, and response message. The code is used as an indexing value for different types of responses. You can generally ignore it and focus on the response and response message to get the details you need about the results of your transaction request.
blng__ResponseMessage__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The message sent from the external payment gateway to show the result of the transaction request. You can review this message for additional context regarding the value of the transaction's gateway status.
blng__ResponseStatus__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The message sent from the external payment gateway to show the result of the transaction request. While this value can vary between gateways, it's often the same as the response message.
blng__Response__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** A detailed message contained in a gateway response alongside the response code, response, response status, and response message. Although specific responses vary between gateways, the response usually provides more information about the response message.
blng__SourceTransactionId__c	**Type** string **Properties** Create, Filter, Group, idLookup, Nillable, Sort, Update **Description** A unique identifier of the source transaction that's associated with the payment.

Field	Details

blng__Status__c	**Type**
	picklist
	
	Properties
	Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	
	Description
	The status of the payment transaction. The transaction's status is
	
	```\nIn Progress\n```
	
	while the external payment gateway communicates with the customer bank. Salesforce Billing changes it to `Completed` after the gateway returns a transaction response. Valid values are:
	
	• `Completed`
	• `In Progress`
	• `Processed`

blng__Transaction__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** All customer transactions in a payment transaction. This field is a relationship field. **Relationship Name** blng__Transaction__r **Relationship Type** Lookup **Refers To** blng__PaymentTransaction__c
blng__Type__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The type of action requested from the customer bank. Valid values are: • `Authorization` • `Capture` • `Charge` • `Refund` • `Void`

blng__Refund__c

Represents a refund made against a payment.

🛑 **Important:** Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the invoice when multiple currencies are enabled.
LastActivityDate	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The timestamp of when a user had last associated a task or event action with the refund.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a refund record indirectly, for example, through a list view or related record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort

256

Field	Details
	Description
	The timestamp when the current user last viewed a refund record. If this value is null, it's possible that the user only accessed the refund record or a related list view (`LastReferencedDate`), but not viewed the refund record itself.
`Name`	**Type**
	string
	Properties
	Autonumber, Defaulted on create, Filter, idLookup, Sort
	Description
	Required. The name of the refund.
`blng__Account__c`	**Type**
	reference
	Properties
	Create, Filter, Group, Sort
	Description
	Required. The customer account containing the payment that this refund targets.
	This field is a relationship field.
	Relationship Name
	blng__Account__r
	Relationship Type
	Master-detail
	Refers To
	Account (the master object)
`blng__Amount__c`	**Type**
	currency
	Properties
	Filter, Nillable, Sort
	Description
	The total amount of this refund.
	This field is a calculated field.
`blng__BaseCurrencyAmount__c`	**Type**
	currency
	Properties
	Create, Filter, Nillable, Sort, Update
	Description
	The amount of the refund in the base currency of the organization. It's used to standardize the amount across different currencies.

Field	Details
blng__BaseCurrencyFXDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date on which the Foreign Exchange (FX) rate is applied to convert an amount into the base currency.
blng__BaseCurrencyFXRate__c	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The foreign exchange rate used to convert an amount from a foreign currency to the base currency.
blng__BaseCurrency__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The primary currency in which the financial transactions are recorded and reported.
blng__BillingFinanceBook__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The financial ledger that tracks all transactions related to billing, including the allocation of payments to debit note lines. This value ensures that all financial activities are accurately recorded and reported. This field is a relationship field. **Relationship Name** blng__BillingFinanceBook__r **Refers To** blng__FinanceBook__c
blng__CheckNumber__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description
	The check number issued in an external system for a refund.
blng__GLRule__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The accounting treatment for financial transactions, including which general ledger accounts to debit and credit.
	This field is a relationship field.
	Relationship Name
	blng__GLRule__r
	Refers To
	blng__GLRule__c
blng__GLTreatment__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The general ledger treatment defines how a financial transaction is recorded in the general ledger, specifying the accounts and the nature of the entries.
	This field is a relationship field.
	Relationship Name
	blng__GLTreatment__r
	Refers To
	blng__GLTreatment__c
blng__ImpactAmount__c	**Type**
	currency
	Properties
	Filter, Nillable, Sort
	Description
	Specifies how this refund impacts a customer's accounts receivable.
	This field is a calculated field.
blng__LegalEntity__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description
	The legal entity used to associate the refund with other transactional records in Salesforce Billing.
	This field is a relationship field.
	Relationship Name
	blng__LegalEntity__r
	Refers To
	blng__LegalEntity__c
blng__Notes__c	**Type**
	textarea
	Properties
	Create, Nillable, Update
	Description
	Additional details or comments related to the refund.
blng__PaymentGatewayTransactionID__c	**Type**
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The transaction ID used to identify the refund in the payment gateway.
blng__RefundDate__c	**Type**
	date
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The date on which a refund is processed or issued to a customer.
blng__RefundDescription__c	**Type**
	textarea
	Properties
	Create, Nillable, Update
	Description
	A description or additional details about a refund.
blng__RefundLineCreationStatus__c	**Type**
	picklist
	Properties
	Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details

Description

The status of the refund line creation. A `Fail` status means that the refund line couldn't be created. The system configuration provides more details about the status.

`blng__RefundMode__c`

Type
picklist

Properties
Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update

Description
Required. The mode of refund. External mode refers to a refund made through an external system, while electronic mode represents a refund processed within Salesforce.

Valid values are:

- `Electronic`
- `External`

The default value is `External`.

`blng__RefundPaymentMethod__c`

Type
reference

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The payment method represents a credit card or banking details that can be used to process a payment.

This field is a relationship field.

Relationship Name
blng__RefundPaymentMethod__r

Refers To
blng__PaymentMethod__c

`blng__RefundType__c`

Type
picklist

Properties
Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Description
The type of refund being processed.

Valid values are:

- `ACH`
- `Bank Transfer`
- `Cash`
- `Check`

Field	Details
	• `Credit Card`
`blng__RefundedfromCreditNoteLines__c`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount refunded from credit note lines. This field is a calculated field.
`blng__RefundedfromPayments__c`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount refunded from payments. This field is a calculated field.
`blng__Status__c`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The state of this refund. Valid values are: • `Cancelled` • `Draft` • `Posted` The default value is `Draft`.

blng__RefundLineCreditNoteLine__c

Represents a relationship between a refund line and a credit note line.

🛑 **Important:** Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the refund line credit note line when multiple currencies are enabled.
Name	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. The name of the refund line credit note line.
blng__Amount__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The total amount applied to or unapplied from a refund line credit note line.
blng__BaseCurrencyAmount__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The amount of the refund line credit note line in the base currency of the organization. This value is used to standardize the amount across different currencies.
blng__BaseCurrencyFXDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The date on which the foreign exchange rate is applied to convert an amount into the base currency.
blng__BaseCurrencyFXRate__c	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The foreign exchange rate used to convert an amount from a foreign currency to the base currency.
blng__BaseCurrency__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The primary currency in which the financial transactions are recorded and reported.
blng__BillingFinanceBook__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** A set of financial records for reporting and accounting purposes. This field is a relationship field. **Relationship Name** blng__BillingFinanceBook__r **Refers To** blng__FinanceBook__c
blng__BillingFinancePeriod__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** A specific time frame for financial reporting and accounting purposes. This field is a relationship field. **Relationship Name** blng__BillingFinancePeriod__r

Field	Details

Refers To
blng__FinancePeriod__c

blng__CreditNoteLine__c

Type
reference

Properties
Create, Filter, Group, Sort

Description
Required. The credit note line item associated with the refund.

This field is a relationship field.

Relationship Name
blng__CreditNoteLine__r

Relationship Type
Master-detail

Refers To
blng__CreditNoteLine__c (the master object)

blng__GLRule__c

Type
reference

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The accounting treatment for financial transactions, including which general ledger accounts to debit and credit.

This field is a relationship field.

Relationship Name
blng__GLRule__r

Refers To
blng__GLRule__c

blng__GLTreatment__c

Type
reference

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The general ledger treatment defines how a financial transaction is recorded in the general ledger, specifying the accounts and the nature of the entries.

This field is a relationship field.

Relationship Name
blng__GLTreatment__r

Field	Details
	Refers To blng__GLTreatment__c
`blng__ImpactAmount__c`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Specifies how this refund line credit note line impacts a customer's accounts receivable. This field is a calculated field.
`blng__LegalEntity__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The legal entity used to associate the refund line credit note line with other transactional records in Salesforce Billing. This field is a relationship field. **Relationship Name** blng__LegalEntity__r **Refers To** blng__LegalEntity__c
`blng__Notes__c`	**Type** textarea **Properties** Create, Nillable, Update **Description** Additional details or comments related to the refund line credit note line.
`blng__Refund__c`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Required. The parent refund of the refund line. This field is a relationship field. **Relationship Name** blng__Refund__r

Field	Details
	Relationship Type Master-detail
	Refers To blng__Refund__c (the detail object)
blng__Status__c	**Type** picklist
	Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description The state of the refund line credit note line. Valid values are: • `Cancelled` • `Draft` • `Posted` The default value is `Draft`.
blng__Type__c	**Type** picklist
	Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description The type of refund line. Valid values are: • `Allocation` • `Unallocation`
blng__Unallocated__c	**Type** boolean
	Properties Create, Defaulted on create, Filter, Group, Sort, Update
	Description Required. The amount of the refund that remains unallocated. This field indicates how much of the refund is still available to be applied to other credit note lines. The default value is `false`.

blng__RefundLinePayment__c

Represents a portion of a refund that's applied to or unapplied from a payment. One refund line can represent the full amount of a refund.

🛑 **Important:** Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the refund line payment when multiple currencies are enabled.
LastActivityDate	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The timestamp of when a user had last associated a task or event action with the account.
Name	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. The name of the refund line payment.
blng__Amount__c	**Type** currency **Properties** Create, Filter, Sort, Update

Field	Details
	Description
	Required. The total amount applied to or unapplied from a payment by the refund line.
blng__BaseCurrencyAmount__c	**Type**
	currency
	Properties
	Create, Filter, Nillable, Sort, Update
	Description
	The amount of the refund line payment in the base currency of the organization. This value is used to standardize the amount across different currencies.
blng__BaseCurrencyFXDate__c	**Type**
	date
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The date on which the foreign exchange rate is applied to convert the amount into the base currency.
blng__BaseCurrencyFXRate__c	**Type**
	double
	Properties
	Create, Filter, Nillable, Sort, Update
	Description
	The foreign exchange rate used to convert an amount from a foreign currency to the base currency.
blng__BaseCurrency__c	**Type**
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The primary currency in which the financial transactions are recorded and reported.
blng__BillingFinanceBook__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	A set of financial records for reporting and accounting purposes.
	This field is a relationship field.

Field	Details
	Relationship Name blng__BillingFinanceBook__r
	Refers To blng__FinanceBook__c
blng__BillingFinancePeriod__c	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description A specific time frame for financial reporting and accounting purposes. This field is a relationship field.
	Relationship Name blng__BillingFinancePeriod__r
	Refers To blng__FinancePeriod__c
blng__GLRule__c	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The accounting treatment for financial transactions, including which general ledger accounts to debit and credit. This field is a relationship field.
	Relationship Name blng__GLRule__r
	Refers To blng__GLRule__c
blng__GLTreatment__c	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The general ledger treatment defines how a financial transaction is recorded in the general ledger, specifying the accounts and the nature of the entries. This field is a relationship field.
	Relationship Name blng__GLTreatment__r

Field	Details

Refers To
blng__GLTreatment__c

`blng__ImpactAmount__c`

Type
currency

Properties
Filter, Nillable, Sort

Description
Specifies how the payment refund line impacts a customer's accounts receivable.

This field is a calculated field.

`blng__LegalEntity__c`

Type
reference

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The legal entity used to associate the revenue line payment with other transactional records in Salesforce Billing.

This field is a relationship field.

Relationship Name
blng__LegalEntity__r

Refers To
blng__LegalEntity__c

`blng__Notes__c`

Type
textarea

Properties
Create, Nillable, Update

Description
Additional details or comments related to the refund line payment.

`blng__PaymentMode__c`

Type
picklist

Properties
Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Description
The method of payment used for a refund line payment.

Valid values are:

- `Electronic`
- `External`

Field	Details
blng__Payment__c	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Required. The payment record that the refund line targets. Refund applications and unapplications are made against the payment record. This field is a relationship field. **Relationship Name** blng__Payment__r **Relationship Type** Master-detail **Refers To** blng__Payment__c (the detail object)
blng__RefundLineDate__c	**Type** date **Properties** Create, Filter, Group, Sort, Update **Description** Required. The date associated with a refund line payment.
blng__Refund__c	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Required. The parent refund of the refund line payment. This field is a relationship field. **Relationship Name** blng__Refund__r **Relationship Type** Master-detail **Refers To** blng__Refund__c (the master object)
blng__Status__c	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The state of the refund line payment. Valid values are:

Field	Details
	• Cancelled
	• Draft
	• Posted
	The default value is `Draft`.
`blng__Type__c`	**Type**
	picklist
	Properties
	Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	The type of refund line.
	Valid values are:
	• Allocation
	• Unallocation
`blng__Unallocated__c`	**Type**
	boolean
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	Required. The amount of the refund that remains unallocated. This field indicates how much of the refund is still available to be applied to other payments. The default value is `false`.

blng__RevenueAdjustment__c

Represents the adjustment to increase, decrease, or transfer the available balance between revenue schedules when actual revenue differs from the initial revenue schedule.

Supported Calls

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

Fields

Field	Details
`CurrencyIsoCode`	**Type**
	picklist

Field	Details
	Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the revenue adjustment when multiple currencies are enabled.
LastActivityDate	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The timestamp of when a user had last associated a task or event action with the revenue adjustment.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a revenue adjustment record indirectly, for example, through a list view or related record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a revenue adjustment record. If this value is null, it's possible that the user only accessed the revenue adjustment record or a related list view (`LastReferencedDate`), but not viewed the revenue adjustment record itself.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the revenue adjustment.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update

Field	Details
	Description Required. The ID of the record owner. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
`blng__Account__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The parent account that's inherited from the order product. This field is a relationship field. **Relationship Name** blng__Account__r **Refers To** Account
`blng__AdjustmentAmount__c`	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The total amount contributed to either increase or decrease the available balance on a revenue schedule.
`blng__AdjustmentGroup__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** A collection of revenue adjustments.
`blng__Amount__c`	**Type** currency **Properties** Filter, Nillable, Sort **Description** All revenue for a revenue adjustment that's available to be adjusted or distributed.

Field	Details
	This field is a calculated field.

Field	Details
blng__FinanceBook__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The finance book that records transactions related to the revenue adjustment. This field is a relationship field. **Relationship Name** blng__FinanceBook__r **Refers To** blng__FinanceBook__c
blng__FinancePeriod__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The time period in which the revenue adjustment transactions are complete. This field is a relationship field. **Relationship Name** blng__FinancePeriod__r **Refers To** blng__FinancePeriod__c
blng__LegalEntity__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The legal entity used to associate the revenue adjustment with other transactional records in Salesforce Billing. This field is a relationship field. **Relationship Name** blng__LegalEntity__r **Refers To** blng__LegalEntity__c
blng__Notes__c	**Type** textarea

Field	Details

Properties

Create, Nillable, Update

Description

Additional details or comments related to the revenue adjustment.

---

blng__RelatedRevenueAdjustment__c

Type

reference

Properties

Create, Filter, Group, Nillable, Sort, Update

Description

The total changes created through revenue adjustments based on related revenue adjustments.

This field is a relationship field.

Relationship Name

blng__RelatedRevenueAdjustment__r

Refers To

blng__RevenueAdjustment__c

---

blng__RevenueAdjustmentDate__c

Type

date

Properties

Create, Filter, Group, Nillable, Sort, Update

Description

The date by which the changes to the revenue adjustment line must be incorporated.

---

blng__Status__c

Type

picklist

Properties

Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Description

The status of the revenue adjustment. Valid values are:

- `Draft`
- `Posted`

The default value is `Draft`.

---

blng__Type__c

Type

picklist

Properties

Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Description

The types of revenue changes supported by the revenue adjustment. Valid values are:

Field	Details
	• `Adjustment`
	• `Transfer`
	The default value is `Adjustment`.

blng__RevenueAdjustmentLine__c

Represents a type of revenue schedule balance change. You can have multiple adjustment lines that each target a different revenue schedule.

🛑 **Important:** Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Supported Calls

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

Fields

Field	Details
`CurrencyIsoCode`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the revenue adjustment line when multiple currencies are enabled.
`LastActivityDate`	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The timestamp of when a user had last associated a task or event action with the revenue adjustment line.
`Name`	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update

Field	Details
	Description The name of the revenue adjustment line.
blng__Account__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The parent account that's inherited from the order product. This field is a relationship field. **Relationship Name** blng__Account__r **Refers To** Account
blng__AdjustmentAmount__c	**Type** currency **Properties** Create, Filter, Sort, Update **Description** Required. The total amount added through revenue adjustments to the revenue adjustment line.
blng__FinanceBook__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The finance book that records transactions related to the revenue adjustment line. This field is a relationship field. **Relationship Name** blng__FinanceBook__r **Refers To** blng__FinanceBook__c
blng__FinancePeriod__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The time period in which the revenue adjustment line transactions are complete.

Field	Details
	This field is a relationship field.

Relationship Name
blng__FinancePeriod__r

Refers To
blng__FinancePeriod__c

Field	Details
blng__GLRule__c	

Type
reference

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The accounting treatment for financial transactions, including which general ledger accounts to debit and credit.

This field is a relationship field.

Relationship Name
blng__GLRule__r

Refers To
blng__GLRule__c

Field	Details
blng__GLTreatment__c	

Type
reference

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The general ledger treatment defines how a financial transaction is recorded in the general ledger, specifying the accounts and the nature of the entries.

This field is a relationship field.

Relationship Name
blng__GLTreatment__r

Refers To
blng__GLTreatment__c

Field	Details
blng__LegalEntity__c	

Type
reference

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The legal entity used to associate the revenue adjustment line with other transactional records in Salesforce Billing.

This field is a relationship field.

Field	Details
	Relationship Name blng__LegalEntity__r
	Refers To blng__LegalEntity__c
blng__Notes__c	**Type** textarea
	Properties Create, Nillable, Update
	Description Additional details or comments related to the revenue adjustment line.
blng__RevenueAdjustmentDate__c	**Type** date
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The date by which the changes to the revenue adjustment line must be incorporated.
blng__RevenueAdjustment__c	**Type** reference
	Properties Create, Filter, Group, Sort
	Description Required. The total changes created through revenue adjustments to this revenue adjustment line. This field is a relationship field.
	Relationship Name blng__RevenueAdjustment__r
	Relationship Type Master-detail
	Refers To blng__RevenueAdjustment__c (the master object)
blng__Status__c	**Type** picklist
	Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description Required. The status of the revenue adjustment line. Valid values are: • Draft

Field	Details

<table>
<tr><td></td><td>• <code>Posted</code>

The default value is <code>Draft</code>.</td></tr>
<tr><td><code>blng__TargetRevenueSchedule__c</code></td><td>Type
reference

Properties
Create, Filter, Group, Sort

Description
Required. The total revenue that is to be allocated.

This field is a relationship field.

Relationship Name
blng__TargetRevenueSchedule__r

Relationship Type
Master-detail

Refers To
blng__RevenueSchedule__c (the detail object)</td></tr>
<tr><td><code>blng__Type__c</code></td><td>Type
picklist

Properties
Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Description
The types of revenue changes supported by the revenue adjustment line. Valid values are:

• <code>Adjustment</code>
• <code>Transfer</code>

The default value is <code>Adjustment</code>.</td></tr>
</table>

blng__RevenueAgreement__c

Represents the objects with a revenue schedule related list. The revenue agreement is useful for grouping revenue schedules related to the same transaction, so that you can quickly evaluate revenue amounts for that transaction.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type**
	picklist
	Properties
	Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	The three-letter ISO 4217 currency code associated with the revenue agreement when multiple currencies are enabled.
Name	**Type**
	string
	Properties
	Autonumber, Defaulted on create, Filter, idLookup, Sort
	Description
	Required. The name of the revenue agreement.
OwnerId	**Type**
	reference
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	Required. The ID of the record owner.
	This field is a polymorphic relationship field.
	Relationship Name
	Owner
	Refers To
	Group, User
blng__Account__c	**Type**
	reference
	Properties
	Create, Filter, Group, Sort, Update
	Description
	Required. The parent account that's inherited from the order product.
	This field is a relationship field.
	Relationship Name
	blng__Account__r
	Refers To
	Account

Field	Details
blng__Contract__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The fields that help customize your renewals and amendments. This field is a relationship field. **Relationship Name** blng__Contract__r **Refers To** Contract
blng__Notes__c	**Type** textarea **Properties** Create, Nillable, Update **Description** Additional details or comments related to the revenue agreement.
blng__OrderProduct__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** A single revenue agreement is generated for an order product when the revenue schedule for that order product is created. Any amendment orders related to the original order product are also associated with the same revenue agreement. This field is a relationship field. **Relationship Name** blng__OrderProduct__r **Refers To** OrderItem
blng__Order__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The order that a sales rep contracted to create or amend this contract record. The order is populated when it's created and is updated only when a contract is amended. This field is a relationship field.

Field	Details
	Relationship Name
	blng__Order__r
	Refers To
	Order
blng__TotalAllocatedRevenueAmtBillings__c	**Type**
	currency
	Properties
	Create, Defaulted on create, Filter, Nillable, Sort, Update
	Description
	The total revenue impact from billing can vary from the subtotal and total amount (including tax) and is used when reallocating revenue across multiple transactions. Additionally, it can be configured to create revenue schedules based on this amount.
blng__TotalAllocatedRevenueAmtBookings__c	**Type**
	currency
	Properties
	Create, Defaulted on create, Filter, Nillable, Sort, Update
	Description
	The total revenue impact from bookings can vary from the subtotal and total amount (including tax) and is used when reallocating revenue across multiple transactions. Additionally, it can be configured to create revenue schedules based on this amount.
blng__TotalRevenueBillings__c	**Type**
	currency
	Properties
	Create, Defaulted on create, Filter, Nillable, Sort, Update
	Description
	The total revenue billing amount during a specific period.
blng__TotalRevenueBookings__c	**Type**
	currency
	Properties
	Create, Defaulted on create, Filter, Nillable, Sort, Update
	Description
	The total revenue booking amount during a specific period.
blng__TotalRevenueExpectedAmtBillings__c	**Type**
	currency
	Properties
	Create, Defaulted on create, Filter, Nillable, Sort, Update

Field	Details

Description

The revenue that finance operations users expect to come from billing.

---

blng__TotalRevenueExpectedAmtBookings__c   **Type**

currency

Properties

Create, Defaulted on create, Filter, Nillable, Sort, Update

Description

The revenue that finance operations users expect to come from bookings.

---

blng__TotalRevenueLiabilityAmtBillings__c   **Type**

currency

Properties

Create, Defaulted on create, Filter, Nillable, Sort, Update

Description

This user-defined field represents the revenue liability linked to the billing. The revenue liability amount may vary from the subtotal and total amount (including tax). Additionally, it can be set up to generate revenue schedules, using this field as the specified amount on the revenue schedule.

---

blng__TotalRevenueLiabilityAmtBookings__c   **Type**

currency

Properties

Create, Defaulted on create, Filter, Nillable, Sort, Update

Description

This user-defined field represents the revenue liability linked to the bookings. The revenue liability amount may vary from the subtotal and total amount (including tax). Additionally, it can be set up to generate revenue schedules, using this field as the specified amount on the revenue schedule.

---

blng__TotalRevenueMostLikelyAmtBillings__c   **Type**

currency

Properties

Create, Defaulted on create, Filter, Nillable, Sort, Update

Description

This user-defined field is the most likely revenue impact from billing for revenue calculations when the revenue amount is uncertain. This field can differ from the subtotal and total amount (with tax).

---

blng__TotalRevenueMostLikelyAmtBookings__c   **Type**

currency

Properties

Create, Defaulted on create, Filter, Nillable, Sort, Update

Field	Details
	Description This user-defined field is the most likely revenue impact from bookings for revenue calculations when the revenue amount is uncertain. This field can differ from the subtotal and total amount (with tax).

blng__RevenueDistributionMethod__c

Represents how the revenue of a company is spread over a specific period.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the revenue distribution method when multiple currencies are enabled.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a revenue distribution method record indirectly, for example, through a list view or related record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a revenue distribution method record. If this value is null, it's possible that the user only accessed the revenue distribution method

Field	Details
	record or a related list view (`LastReferencedDate`), but not viewed the revenue distribution method record rule itself.

Field	Details
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the revenue distribution method.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. The ID of the record owner. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
blng__Active__c	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. Indicates whether the legal entity is active (`true`) or not (`false`). To deactivate the legal entity, clear the **Active** checkbox. This setting won't affect past or existing transactions or assignments. The default value is `true`.
blng__DistributionMethod__c	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. The revenue amount is prorated based on the distribution method. Valid values are: • `Daily` • `Full Recognition` • `Monthly`

Field	Details
blng__FullRecognitionDate__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Specifies when revenue is recognized in full. Valid values are: • `Credit Note Date` • `Credit Note Line End Date` • `Credit Note Line Start Date` • `Debit Note Date` • `Debit Note Line End Date` • `Debit Note Line Start Date` • `Invoice Line End Date` • `Invoice Line Start Date` • `Manual` • `Order Product End Date` • `Order Product Start Date` • `Other`
blng__MonthlyRecognitionProration__c	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. If your revenue transaction covers a partial period (for example, 15 days in a monthly distribution method), the transaction amount is prorated based on this value. Currently, proration is only possible based on the number of days in the partial period.
blng__Notes__c	**Type** textarea **Properties** Create, Nillable, Update **Description** Additional details or comments related to the revenue distribution method.
blng__RevenueScheduleTermEndDate__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details
	Description
	The end date of the revenue schedule term. These dates can be sourced from the order product, invoice line, credit note, or debit note. If set to `Manual`, the user-defined code is referenced to determine the revenue distribution. Valid values are:

- `Credit Note Line End Date`
- `Debit Note Line End Date`
- `Invoice Line End Date`
- `Manual`
- `Order Product End Date`
- `Other`

Field	Details
`blng__RevenueScheduleTermStartDate__c`	**Type**
	picklist
	Properties
	Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	The start date of the revenue schedule term. These dates can be sourced from the order product, invoice line, credit note, or debit note. If set to `Manual`, the user-defined code is referenced to determine the revenue distribution. Valid values are:

- `Credit Note Line Start Date`
- `Debit Note Line Start Date`
- `Invoice Line Start Date`
- `Manual`
- `Order Product Start Date`
- `Other`

Field	Details
`blng__RevenueTransactionCreationProcess__c`	**Type**
	picklist
	Properties
	Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update
	Description
	Required. Specifies whether a revenue schedule and transaction are automatically created based on the value defined in the **Type** field.
	Valid values are:

- `Automatic`
- `Manual`

The default value is `Automatic`.

Field	Details
`blng__Type__c`	**Type**
	picklist

Field	Details
	Properties Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. The type of revenue transaction used in the revenue distribution method. Valid values are: • ``Credit Note`` • ``Debit Note`` • ``Invoice`` • ``Order`` • ``Other``

blng__RevenueRecognitionRule__c

Represents the creation of a revenue schedule for an invoiced order product or invoice line.

🛑 **Important:** Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Supported Calls

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

Fields

Field	Details
``CurrencyIsoCode``	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the revenue recognition rule when multiple currencies are enabled.
``LastReferencedDate``	**Type** dateTime **Properties** Filter, Nillable, Sort

Field	Details
	Description The timestamp when the current user last accessed a revenue recognition rule record indirectly, for example, through a list view or related record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a revenue recognition rule record. If this value is null, it's possible that the user only accessed the revenue recognition rule record or a related list view (`LastReferencedDate`), but not viewed the revenue recognition rule record itself.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the revenue recognition rule.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. The ID of the record owner. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
blng__Active__c	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. Indicates whether the revenue recognition rule is active (`true`) or not (`false`). To deactivate the revenue recognition rule, clear the **Active** checkbox. This setting won't affect past or existing transactions or assignments. The default value is `true`.

Field	Details
blng__CreateRevenueSchedule__c	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. The corresponding revenue schedule created for the revenue recognition rule. Valid values are: • `No` • `Yes`
blng__Notes__c	**Type** textarea **Properties** Create, Nillable, Update **Description** Additional details or comments related to the revenue recognition rule.

blng__RevenueRecognitionTreatment__c

Represents how and where the revenue of a product is recognized.

🛑 **Important:** Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the revenue recognition treatment when multiple currencies are enabled.

Field	Details
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a revenue recognition treatment record indirectly, for example, through a list view or related record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a revenue recognition treatment record. If this value is null, it's possible that the user only accessed the revenue recognition treatment record or a related list view (`LastReferencedDate`), but not viewed the revenue recognition treatment record itself.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the revenue recognition treatment.
blng__Active__c	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. Indicates whether the revenue recognition rule is active (`true`) or not (`false`). To deactivate the revenue recognition rule, clear the **Active** checkbox. This setting won't affect past or existing transactions or assignments. The default value is `true`.
blng__FlatAmount__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The fixed amount to be paid for the revenue recognition treatment.

Field	Details
blng__Notes__c	**Type** textarea **Properties** Create, Nillable, Update **Description** Additional details or comments related to the revenue recognition treatment.
blng__Percentage__c	**Type** percent **Properties** Create, Filter, Nillable, Sort, Update **Description** The total percentage applied to the revenue recognition treatment and is null if the line is nontaxable.
blng__ProcessingOrder__c	**Type** double **Properties** Create, Filter, Sort, Update **Description** Required. If you have multiple revenue recognition treatments targeting a single order product, you can give each of them a value here. The treatments are applied starting with the lowest number.
blng__RevenueAgreementAssociation__c	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The option for associating the revenue schedule to the revenue agreement. Valid values are: • Manual • New Agreement • Not Applicable • Revised Order Product The default value is Not Applicable.
blng__RevenueDistributionMethod__c	**Type** reference **Properties** Create, Filter, Group, Sort, Update

Field	Details
	Description
	Required. The revenue distribution method used for assigning review across periods.
	This field is a relationship field.
	Relationship Name
	blng__RevenueDistributionMethod__r
	Refers To
	blng__RevenueDistributionMethod__c
blng__RevenueFinanceBook__c	**Type**
	reference
	Properties
	Create, Filter, Group, Sort, Update
	Description
	Required. All transactions processed through this treatment record entries in this finance book record.
	This field is a relationship field.
	Relationship Name
	blng__RevenueFinanceBook__r
	Refers To
	blng__FinanceBook__c
blng__RevenueGLRule__c	**Type**
	reference
	Properties
	Create, Filter, Group, Sort, Update
	Description
	Required. All transactions processed through this treatment record journal entries based on this general ledger rule.
	This field is a relationship field.
	Relationship Name
	blng__RevenueGLRule__r
	Refers To
	blng__GLRule__c
blng__RevenueLegalEntity__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	This treatment applies to all order products that were considered under the parent revenue recognition rule and have a matching legal entity.

Field	Details
	This field is a relationship field.
	Relationship Name blng__RevenueLegalEntity__r
	Refers To blng__LegalEntity__c
blng__RevenueRecognitionRule__c	**Type** reference
	Properties Create, Filter, Group, Sort
	Description Required. The revenue recognition rule associated with the revenue recognition treatment. This field is a relationship field.
	Relationship Name blng__RevenueRecognitionRule__r
	Relationship Type Master-detail
	Refers To blng__RevenueRecognitionRule__c (the master object)
blng__RevenueScheduleAmount__c	**Type** picklist
	Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description The type of revenue recognition used in the revenue schedule. Valid values are: • Bookings Amount • Revenue Allocation Amount • Revenue Expected Amount • Revenue Liability Amount • Transaction Amount The default value is Transaction Amount.
blng__RevenueScheduleCreationAction__c	**Type** picklist
	Properties Create, Filter, Group, Restricted picklist, Sort, Update
	Description Required. The event that triggers the creation of the revenue schedule.

Field	Details
	Valid values are:

- Credit Note Posting
- Debit Note Posting
- Invoice Posting
- Order Activation
- Other

Field	Details
blng__Type__c	**Type** picklist
	Properties Create, Filter, Group, Restricted picklist, Sort, Update
	Description Required. The method for calculating the revenue recognition treatment amount. Valid values are:

- Flat Amount
- Percentage
- Remainder

Field	Details
blng__UniqueId__c	**Type** string
	Properties Create, Filter, Group, idLookup, Nillable, Sort, Update
	Description The unique identification number associated with this invoice. This field is unique within your organization.
blng__ValidateResult__c	**Type** string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The outcome of the validation checks performed during the revenue recognition treatment process.

blng__RevenueSchedule__c

Represents the fields that you can use to review the amount of revenue already distributed and remaining to be allocated. A revenue schedule is associated with a revenue transaction.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the revenue schedule when multiple currencies are enabled.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a revenue schedule record indirectly, for example, through a list view or related record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a revenue schedule record. If this value is null, it's possible that the user only accessed the revenue schedule record or a related list view (`LastReferencedDate`), but not viewed the revenue schedule record itself.
Name	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. The name of the revenue schedule.
OwnerId	**Type** reference

Field	Details
	Properties Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. The ID of the record owner. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
Quote__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The quote associated with the revenue schedule. This field is a relationship field. **Relationship Name** Quote__r **Refers To** SBQQ__Quote__c
blng__Account__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The account of the customer who made the transaction. This field is a relationship field. **Relationship Name** blng__Account__r **Refers To** Account
blng__Adjustments__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total of all adjustments made to the revenue schedule. This field is a calculated field.

Field	Details
blng__AllocatedRevenueAmount__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The amount of revenue allocated to or for a revenue schedule.
blng__Available__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total amount on the revenue schedule, including adjustments, that hasn't been distributed to any revenue transactions. This field is a calculated field.
blng__BaseCurrencyAmount__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The amount of the revenue schedule in the base currency of the organization. This value is used to standardize the amount across different currencies.
blng__BaseCurrencyFXDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date on which the Foreign Exchange (FX) rate is applied to convert the amount into the base currency.
blng__BaseCurrencyFXRate__c	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The FX rate used to convert an amount from a foreign currency to the base currency.
blng__BaseCurrency__c	**Type** string

Field	Details

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The primary currency in which the financial transactions are recorded and reported.

---

blng__BillingFinanceBook__c

Type
reference

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The financial ledger that tracks all transactions related to billing, including the allocation of payments to debit note lines. This value ensures that all financial activities are accurately recorded and reported.

This field is a relationship field.

Relationship Name
blng__BillingFinanceBook__r

Refers To
blng__FinanceBook__c

---

blng__BillingFinancePeriod__c

Type
reference

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The time frame during which the allocation of payments to revenue schedule is recorded. This period helps in organizing financial data for reporting and analysis, such as monthly or quarterly financial statements.

This field is a relationship field.

Relationship Name
blng__BillingFinancePeriod__r

Refers To
blng__FinancePeriod__c

---

blng__Contract__c

Type
reference

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The contract associated with the revenue schedule.

This field is a relationship field.

Field	Details
	Relationship Name blng__Contract__r
	Refers To Contract
blng__CreditNoteLine__c	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The credit note line associated with the revenue schedule. This field is a relationship field.
	Relationship Name blng__CreditNoteLine__r
	Refers To blng__CreditNoteLine__c
blng__DebitNoteLine__c	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The debit note line associated with the revenue schedule. This field is a relationship field.
	Relationship Name blng__DebitNoteLine__r
	Refers To blng__DebitNoteLine__c
blng__DeferredBalance__c	**Type** currency
	Properties Filter, Nillable, Sort
	Description
	Description The sum of unrecognized revenue in the current period. This field is a calculated field. This field is a calculated field.

Field	Details
blng__Deferred__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total amount of revenue that has been deferred and not yet recognized in the revenue schedule. This field is a calculated field.
blng__EstimatedRevenueTransactionCount__c	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The estimated number of revenue transactions that the schedule contains.
blng__FullRecognitionDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date by which the revenue schedule recognizes all revenue for its transactions. This field is inherited from the revenue distribution method's full recognition date field.
blng__InvoiceLine__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The invoice line associated with the revenue schedule. This field is a relationship field. **Relationship Name** blng__InvoiceLine__r **Refers To** blng__InvoiceLine__c
blng__LegalEntity__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details

Description

The legal entity associated with the revenue schedule.

This field is a relationship field.

Relationship Name

blng__LegalEntity__r

Refers To

blng__LegalEntity__c

---

blng__Notes__c

Type

textarea

Properties

Create, Nillable, Update

Description

Additional details or comments related to the revenue schedule.

---

blng__OrderProduct__c

Type

reference

Properties

Create, Filter, Group, Nillable, Sort, Update

Description

The order product associated with the revenue schedule.

This field is a relationship field.

Relationship Name

blng__OrderProduct__r

Refers To

OrderItem

---

blng__RecognizedRevenueLastClosedPeriod__c

Type

currency

Properties

Filter, Nillable, Sort

Description

The sum of all revenue that has been recognized through the current period.

This field is a calculated field.

---

blng__RevenueAgreementAllocationStatus__c

Type

string

Properties

Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description Specifies whether any revenue has been allocated to or from the revenue agreement related to a revenue schedule.
blng__RevenueAgreement__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The revenue agreement associated with the revenue schedule. This field is a relationship field. **Relationship Name** blng__RevenueAgreement__r **Refers To** blng__RevenueAgreement__c
blng__RevenueDistributionMethod_c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The method for calculating how revenue is spread across a period. This field is a relationship field. **Relationship Name** blng__RevenueDistributionMethod__r **Refers To** blng__RevenueDistributionMethod__c
blng__RevenueEndDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The earliest date for revenue recognized under the revenue schedule.
blng__RevenueExpectedAmount__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The total amount of revenue that's expected to be received in a revenue schedule.

Field	Details
blng__RevenueFinanceBook__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The finance book that records the transactions related to the revenue schedule. This field is a relationship field. **Relationship Name** blng__RevenueFinanceBook__r **Refers To** blng__FinanceBook__c
blng__RevenueLiabilityAmount__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The amount of revenue collected for which a product or service hasn't been rendered.
blng__RevenueMostLikelyAmount__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The amount of revenue that's most likely to be received.
blng__RevenueRecognitionGLRule__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The revenue recognition General Ledger (GL) rule associated with the revenue schedule. This field is a relationship field. **Relationship Name** blng__RevenueRecognitionGLRule__r **Refers To** blng__GLRule__c
blng__RevenueRecognitionGLTreatment__c	**Type** reference

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** The revenue recognition GL treatment associated with the revenue schedule. This field is a relationship field. **Relationship Name** blng__RevenueRecognitionGLTreatment__r **Refers To** blng__GLTreatment__c
blng__RevenueRecognitionRule__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The revenue recognition rule associated with the revenue schedule. This field is a relationship field. **Relationship Name** blng__RevenueRecognitionRule__r **Refers To** blng__RevenueRecognitionRule__c
blng__RevenueRecognitionTreatment__c	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** **Description** The revenue recognition treatment associated with the revenue schedule. This field is a relationship field. This field is a relationship field. **Relationship Name** blng__RevenueRecognitionTreatment__r **Refers To** blng__RevenueRecognitionTreatment__c
blng__RevenueStartDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The earliest date for revenue recognized under the revenue schedule.
blng__RevenueTransactionStatus__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The status of the process that runs to create revenue transactions for the revenue schedule. Valid values are: • `Complete` • `Error` • `Queued`
blng__TotalRevenueAmount__c	**Type** currency **Properties** Create, Filter, Sort, Update **Description** Required. The total amount available at the time the revenue schedule is created.
blng__Unrecognized__c	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total amount of all revenue allocated to a revenue schedule but not yet recognized. This field is a calculated field.

blng__RevenueTransaction__c

Represents the revenue transactions made against an order product.

🛑 **Important:** Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the revenue transaction when multiple currencies are enabled.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a revenue transaction record indirectly, for example, through a list view or related record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a revenue transaction record. If this value is null, it's possible that the user only accessed the revenue transaction record or a related list view (LastReferencedDate), but not viewed the revenue transaction record itself.
Name	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. The name of the revenue transaction.
blng__Account__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The account of the customer who made the transaction. This field is a relationship field.

Field	Details
	Relationship Name blng__Account__r **Refers To** Account
blng__BaseCurrencyAmount__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The amount of the revenue transaction in the base currency of the organization. This value is used to standardize the amount across different currencies.
blng__BaseCurrencyFXDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date on which the Foreign Exchange (FX) rate is applied to convert the amount into the base currency.
blng__BaseCurrencyFXRate__c	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The FX rate used to convert an amount from a foreign currency to the base currency.
blng__BaseCurrency__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The primary currency in which the financial transactions are recorded and reported.
blng__Contract__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The contract associated with the revenue transaction.

Field	Details
	This field is a relationship field.
	Relationship Name blng__Contract__r
	Refers To Contract
blng__CreditNoteLine__c	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The credit note line associated with the revenue transaction. This field is a relationship field.
	Relationship Name blng__CreditNoteLine__r
	Refers To blng__CreditNoteLine__c
blng__DebitNoteLine__c	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The debit note line associated with the revenue transaction. This field is a relationship field.
	Relationship Name blng__DebitNoteLine__r
	Refers To blng__DebitNoteLine__c
blng__DeferredBalance__c	**Type** currency
	Properties Create, Filter, Nillable, Sort, Update
	Description The total unrecognized revenue in the current period. This field is a calculated field.
blng__InvoiceLine__c	**Type** reference

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** The invoice line associated with the revenue transaction. This field is a relationship field. **Relationship Name** blng__InvoiceLine__r **Refers To** blng__InvoiceLine__c
blng__LegalEntity__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The legal entity associated with the revenue transaction. This field is a relationship field. **Relationship Name** blng__LegalEntity__r **Refers To** blng__LegalEntity__c
blng__Notes__c	**Type** textarea **Properties** Create, Nillable, Update **Description** Additional details or comments related to the revenue transaction.
blng__OrderProduct__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The order product associated with the revenue transaction. This field is a relationship field. **Relationship Name** blng__OrderProduct__r **Refers To** OrderItem

Field	Details
blng__OriginalFinancePeriod__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The original finance period defined in the revenue schedule. This field is a relationship field. **Relationship Name** blng__OriginalFinancePeriod__r **Refers To** blng__FinancePeriod__c
blng__PreviouslyRecognizedRevenue__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The sum of all prior transactions using the revenue schedule.
blng__RevenueAmount__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The amount of revenue associated with a transaction.
blng__RevenueDistributionMethod__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The distribution method selected in the revenue schedule. Specifies whether revenue is allocated over a predefined period, recognized all at once, or distributed based on a manually entered time frame.
blng__RevenueFinanceBook__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The finance book associated with the revenue transaction.

Field	Details
blng__RevenueFinancePeriodUniqueId__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** A unique ID of the revenue finance period.
blng__RevenueFinancePeriod__c	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Required. The revenue finance period associated with the revenue transaction. This field is a relationship field. **Relationship Name** blng__RevenueFinancePeriod__r **Relationship Type** Master-detail **Refers To** blng__FinancePeriod__c (the detail object)
blng__RevenueRecognitionGLRule__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The revenue recognition General Ledger (GL) rule associated with the revenue transaction. This field is a relationship field. **Relationship Name** blng__RevenueRecognitionGLRule__r **Refers To** blng__GLRule__c
blng__RevenueRecognitionGLTreatment__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** **Description** The revenue recognition GL treatment associated with the revenue transaction.

Field	Details
	This field is a relationship field.
	Relationship Name blng__RevenueRecognitionGLTreatment__r
	Refers To blng__GLTreatment__c
blng__RevenueRecognitionRule__c	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The revenue recognition rule linked to the revenue transaction, whichis used to calculate the transaction. This field is a relationship field.
	Relationship Name blng__RevenueRecognitionRule__r
	Refers To blng__RevenueRecognitionRule__c
blng__RevenueRecognitionTreatment__c	**Type** string
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The revenue recognition treatment linked to the revenue transaction, which is used to calculate the transaction.
blng__RevenueSchedule__c	**Type** reference
	Properties Create, Filter, Group, Sort
	Description Required. The revenue schedule associated with the revenue transaction. This field is a relationship field.
	Relationship Name blng__RevenueSchedule__r
	Relationship Type Master-detail
	Refers To blng__RevenueSchedule__c (the master object)

Field	Details
blng__Status__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The status of the revenue transaction. Valid values are: • `Canceled` • `Deferred` • `Recognized`
blng__SystemPeriodStartDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The start date of the system period for the revenue transaction.

blng__SubInvoiceLine__c

Represents a detailed line item within an invoice that breaks down the provided charges or services.

🛑 **Important:** Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Supported Calls

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

Fields

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

Field	Details
	Description The three-letter ISO 4217 currency code associated with the sub invoice line record when multiple currencies are enabled.
Name	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. The name of the sub invoice line.
blng__BillingTransaction__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The billing transaction associated with the sub invoice line. This field is a relationship field. **Relationship Name** blng__BillingTransaction__r **Refers To** blng__BillingTransaction__c
blng__ChargeDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The transaction date of the sub invoice line.
blng__EndDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The end date of the billed period for this sub invoice line. This value can impact the revenue.
blng__InvoiceLine__c	**Type** reference **Properties** Create, Filter, Group, Sort

Field	Details
	Description Required. The invoice line associated with the sub invoice line. This field is a relationship field. **Relationship Name** blng__InvoiceLine__r **Relationship Type** Master-detail **Refers To** blng__InvoiceLine__c (the master object)
blng__Notes__c	**Type** textarea **Properties** Create, Nillable, Update **Description** Additional details or comments related to the sub invoice line.
blng__OrderProduct__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The order product that generated the sub invoice line. This field is a relationship field. **Relationship Name** blng__OrderProduct__r **Refers To** OrderItem
blng__Quantity__c	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The quantity invoiced from the order product.
blng__StartDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description
	The start date of the billed period for this sub invoice line. This value can impact the revenue.
blng__Subtotal__c	**Type**
	currency
	Properties
	Create, Filter, Nillable, Sort, Update
	Description
	The total amount to settle on this sub invoice line, excluding tax.
blng__TaxAmount__c	**Type**
	currency
	Properties
	Create, Filter, Nillable, Sort, Update
	Description
	The total tax owed on this sub invoice line.
blng__UniqueId__c	**Type**
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	A unique identification number associated with the sub invoice line record. This field is unique within your organization.
blng__UnitPrice__c	**Type**
	currency
	Properties
	Create, Filter, Nillable, Sort, Update
	Description
	The price of the order product.

blng__TaxConfig__c

Represents the tax configuration associated with a billing schedule.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the tax configuration record when multiple currencies are enabled.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Sort, Update **Description** Required. The name of the tax configuration.
SetupOwnerId	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the record owner. This field is a polymorphic relationship field. **Relationship Name** SetupOwner **Refers To** Organization, Profile, User
blng__ClassName__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The Apex class that handles the tax calculations.

blng__TaxIntegration__c

Represents the interaction between Salesforce and external tax calculation services or systems.

Supported Calls

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

Fields

Field	Details
AVA_BLNG__AvaTaxAccountNumber__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The account number required to connect to AvaTax, a tax compliance software service for tax calculations.
AVA_BLNG__AvaTaxLicenseKey__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The license key for AvaTax.
AVA_BLNG__CustomerCode__c	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The customer code that's important for tax calculations or integrations with tax services. Valid values are: • Id • Name The default value is Id.
AVA_BLNG__DocCode__c	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The document code relevant to tax calculations or reporting. Valid values are: • Id

Field	Details
	• `Name` The default value is `Id`.
AVA_BLNG__Save_transactions_to_AvaTax__c	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. Indicates whether transactions are saved to AvaTax for tax calculation purposes (`true`) or not (`false`). The default value is `true`.
`AVA_BLNG__TestMode__c`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. Indicates whether the tax integration is operating in test mode (`true`) or not (`false`). The default value is `true`.
`AVA_BLNG__Validate_Address__c`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. Indicates whether to validate the addresses for tax purposes, in conjunction with a third-party tax service Avalara (`true`) or not (`false`). The default value is `true`.
`CurrencyIsoCode`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the tax integration record when multiple currencies are enabled.
`LastReferencedDate`	**Type** dateTime

Field	Details
	Properties
	Filter, Nillable, Sort
	Description
	The timestamp when the current user last accessed a tax integration record indirectly, for example, through a list view or related record.
LastViewedDate	**Type**
	dateTime
	Properties
	Filter, Nillable, Sort
	Description
	The timestamp when the current user last viewed a tax integration record. If this value is null, it's possible that the user only accessed the tax integration record or a related list view (`LastReferencedDate`), but not viewed the billing rule record itself.
Name	**Type**
	string
	Properties
	Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update
	Description
	The name of the tax integration.
OwnerId	**Type**
	reference
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	Required. The ID of the record owner.
	This field is a polymorphic relationship field.
	Relationship Name
	Owner
	Refers To
	Group, User
blng__Active__c	**Type**
	boolean
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	Required. Indicates whether the tax integration is active (`true`) or not (`false`). To deactivate the tax integration, clear the `Active` checkbox. This setting won't affect past or existing transactions or assignments. The default value is `true`.

Field	Details
blng__Default__c	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The default value is `false`.
blng__Notes__c	**Type** textarea **Properties** Create, Nillable, Update **Description** Additional details or comments related to the tax integration.
blng__TaxEngine__c	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. An instance of a tax engine provider and the merchant credentials for a specific instance. Valid values are: • `Avalara AvaTax` • `Standard` The default value is `Standard`.

blng__TaxRate__c

Represents a tax rate for a tax code and country.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the tax rate record when multiple currencies are enabled.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a tax rate record indirectly, for example, through a list view or related record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a tax rate record. If this value is null, it's possible that the user only accessed the tax rate record or a related list view (`LastReferencedDate`), but not viewed the tax rate record itself.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the tax rate.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. The ID of the record owner. This field is a polymorphic relationship field.

Field	Details
	Relationship Name Owner **Refers To** Group, User
`blng__AddressComparison__c`	**Type** string **Properties** Create, Filter, Group, idLookup, Nillable, Sort, Update **Description** Compares the address details provided in the tax rate record to ensure accurate tax calculations.
`blng__City__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The city of the company that this tax rate record represents.
`blng__Country__c`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The country of the company that this tax rate record represents.
`blng__LegalEntity__c`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The legal entity associated with the tax rate. This field is a relationship field. **Relationship Name** blng__LegalEntity__r **Refers To** blng__LegalEntity__c
`blng__Notes__c`	**Type** textarea

Field	Details

Properties
Create, Nillable, Update

Description
Additional details or comments related to the tax rate.

---

blng__PriorityAddressComparison__c

Type
string

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The priority of tax rates based on address comparisons.

---

blng__Priority__c

Type
picklist

Properties
Create, Filter, Group, Restricted picklist, Sort, Update

Description
Required. The tax rate to apply when multiple tax rates are available for a given transaction.

Valid values are:

- 0
- 1
- 2
- 3
- 4
- 5
- 6

---

blng__Rate__c

Type
percent

Properties
Create, Filter, Sort, Update

Description
Stores the rate of a tax.

---

blng__State__c

Type
string

Properties
Create, Filter, Group, Nillable, Sort, Update

Description
The state of the company that this tax rate record represents.

Field	Details
blng__TaxCode__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The product associated to a tax rate.
blng__ZipCode__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The zip code of the company that this tax rate record represents.

blng__TaxRule__c

Represents how Salesforce Billing processes tax for the transactions. All products that must be billed require a lookup to a tax rule.

Supported Calls

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

Fields

Field	Details
CurrencyIsoCode	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The three-letter ISO 4217 currency code associated with the tax rule record when multiple currencies are enabled.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort

Field	Details
	Description The timestamp when the current user last accessed a tax rule record indirectly, for example, through a list view or related record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a tax rule record. If this value is null, it's possible that the user only accessed the tax rule record or a related list view (`LastReferencedDate`), but not viewed the tax rule record itself.
Name	**Type** string **Properties** Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The name of the tax rule.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. The ID of the record owner. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
blng__Active__c	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. Indicates whether the tax rule is active (`true`) or not (`false`). To deactivate the tax rule, clear the `Active` checkbox. This setting won't affect past or existing transactions or assignments. The default value is `true`.

Field	Details
blng__Notes__c	**Type** textarea **Properties** Create, Nillable, Update **Description** Additional details or comments related to the tax rule.
blng__TaxableYesNo__c	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. Specifies whether the products associated with a tax rule are subject to taxation. Valid values are: • No • Yes The default value is Yes.

blng__TaxTreatment__c

Represents how Salesforce and external engines calculate taxes, and the tax engine to use for tax calculation.

🛑 **Important:** Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Supported Calls

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

Fields

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

Field	Details
	Description
	The three-letter ISO 4217 currency code associated with the tax treatment record when multiple currencies are enabled.
LastReferencedDate	**Type**
	dateTime
	Properties
	Filter, Nillable, Sort
	Description
	The timestamp when the current user last accessed a tax treatment record indirectly, for example, through a list view or related record.
LastViewedDate	**Type**
	dateTime
	Properties
	Filter, Nillable, Sort
	Description
	The timestamp when the current user last viewed a tax treatment record. If this value is null, it's possible that the user only accessed the tax treatment record or a related list view (`LastReferencedDate`), but not viewed the tax treatment record itself.
Name	**Type**
	string
	Properties
	Create, Defaulted on create, Filter, Group, idLookup, Nillable, Sort, Update
	Description
	The name of the tax treatment.
blng__Active__c	**Type**
	boolean
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	Required. Indicates whether the tax treatment is active (`true`) or not (`false`). To deactivate the legal entity, clear the `Active` checkbox. This setting won't affect past or existing transactions or assignments. The default value is `true`.
blng__Notes__c	**Type**
	textarea
	Properties
	Create, Nillable, Update

Field	Details
	Description
	Additional details or comments related to the tax treatment.
`blng__TaxCode__c`	**Type**
	string
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The product associated to a tax treatment.
`blng__TaxGLRule__c`	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The General Ledger (GL) rule associated with the tax treatment.
	This field is a relationship field.
	Relationship Name
	blng__TaxGLRule__r
	Refers To
	blng__GLRule__c
`blng__TaxIntegration__c`	**Type**
	reference
	Properties
	Create, Filter, Group, Sort
	Description
	Required. The tax integration record associated with tax treatment.
	This field is a relationship field.
	Relationship Name
	blng__TaxIntegration__r
	Relationship Type
	Master-detail
	Refers To
	blng__TaxIntegration__c (the master object)
`blng__TaxLegalEntity__c`	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The legal entity associated with the tax treatment. This field is a relationship field.
	Relationship Name blng__TaxLegalEntity__r
	Refers To blng__LegalEntity__c
`blng__TaxRule__c`	**Type** reference
	Properties Create, Filter, Group, Sort
	Description Required. The tax rule associated with the tax treatment. This field is a relationship field.
	Relationship Name blng__TaxRule__r
	Relationship Type Master-detail
	Refers To blng__TaxRule__c (the detail object)

blng__Usage__c

Represents the amount of service consumed over a period of time.

Supported Calls

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

Fields

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

Field	Details
	Description
	The three-letter ISO 4217 currency code associated with the usage record when multiple currencies are enabled.
LastActivityDate	**Type**
	date
	Properties
	Filter, Group, Nillable, Sort
	Description
	The timestamp of when a user had last associated a task or event action with the usage record.
LastReferencedDate	**Type**
	dateTime
	Properties
	Filter, Nillable, Sort
	Description
	The timestamp when the current user last accessed a usage record indirectly, for example, through a list view or related record.
LastViewedDate	**Type**
	dateTime
	Properties
	Filter, Nillable, Sort
	Description
	The timestamp when the current user last viewed a usage record. If this value is null, it's possible that the user only accessed the usage record or a related list view (`LastReferencedDate`), but not viewed the usage record itself.
Name	**Type**
	string
	Properties
	Autonumber, Defaulted on create, Filter, idLookup, Sort
	Description
	Required. The name of the usage record.
OwnerId	**Type**
	reference
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	Required. The ID of the record owner.

Field	Details
	This field is a polymorphic relationship field.
	Relationship Name
	Owner
	Refers To
	Group, User
blng__Account__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The account of the customer who consumed the service.
	This field is a relationship field.
	Relationship Name
	blng__Account__r
	Refers To
	Account
blng__BaseCurrencyAmount__c	**Type**
	currency
	Properties
	Create, Filter, Nillable, Sort, Update
	Description
	The amount of the payment in the base currency of the organization. It's used to standardize the amount across different currencies.
blng__BaseCurrencyFXDate__c	**Type**
	date
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The date on which the Foreign Exchange (FX) rate is applied to convert an amount into the base currency.
blng__BaseCurrencyFXRate__c	**Type**
	double
	Properties
	Create, Filter, Nillable, Sort, Update
	Description
	The FX rate used to convert an amount from a foreign currency to the base currency.

Field	Details
blng__BaseCurrency__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The primary currency in which the financial transactions are recorded and reported.
blng__EndDateTime__c	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** Required. The end date and time of a particular usage record.
blng__ErrorMessage__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** If Salesforce Billing can't calculate a value on this usage record, it shows a message with the reason for the error.
blng__MatchingAttribute__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** This field is filled in based on the consumption schedule. Usages are evaluated against usage summaries where the matching ID, matching attribute, and unit of measure fields align. When the charge type is usage, legacy usage pricing does not consider this field.
blng__MatchingId__c	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** Required. A usage and a usage summary are linked through a lookup relationship. For the usage to be included in the usage summary, both records must have the same matching ID.
blng__Notes__c	**Type** textarea

337

Field	Details
	Properties Create, Nillable, Update **Description** Additional information or comments related to the usage record.
blng__OrderProduct__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The order product associated with the usage record. This association helps in tracking the usage of a product that's ordered, which is essential for billing and invoicing purposes. This field is a relationship field. **Relationship Name** blng__OrderProduct__r **Refers To** OrderItem
blng__Order__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The order associated with the usage record. This field is used to manage and track billing-related information for orders. This field is a relationship field. **Relationship Name** blng__Order__r **Refers To** Order
blng__PreratedAmount__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** When users enter a prerated amount on a usage record, Salesforce Billing uses that value for the usage's total rather than the standard calculation of usage.
blng__PreratedQuantity__c	**Type** double

Field	Details
	Properties Create, Filter, Nillable, Sort, Update **Description** The total number of prerated times an end customer used this service during the usage record's start and end dates.
blng__ProcessingDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date when a particular usage record is processed.
blng__Quantity__c	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The total number of times an end customer used this service during the usage record's start and end dates.
blng__ReferenceId__c	**Type** string **Properties** Create, Filter, Group, idLookup, Nillable, Sort, Update **Description** Reference to source for which the usage record is created.
blng__SourceId__c	**Type** string **Properties** Create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The source associated with the usage record.
blng__StartDateTime__c	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** Required. The start date and time of a usage event.

Field	Details
blng__StatusCode__c	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** This field handles and sets all HTTP response codes.
blng__Status__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The status of the usage record. Valid values are: • Error • New • Processed • Queued for subtotal calculation • Warning - Unrated
blng__Subtotal__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The subtotal amount for a usage record.
blng__UnitOfMeasure__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Populated from the consumption schedule. Usages are rated against usage summaries where the matching ID, matching attribute, and unit of measure fields match. When the charge type is usage, legacy usage pricing ignores this field. Valid values are: • Hours • None

Field	Details
blng__UnitPrice__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** A usage summary's unit price is based on where the summary's total quantity falls within the price schedule. When you use volume price schedules, your usage records and usage summary all have the same unit price. But for tiered price schedules, your usages could have different unit prices than the usage summary based on how their quantity ranges fall in each price tier.
blng__UsageSummary__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The usage summary associated with the usage record. This field is a relationship field. **Relationship Name** blng__UsageSummary__r **Refers To** blng__UsageSummary__c

blng__UsageSummary__c

Represents the total quantity and value of related usages, and the date ranges for including usage records.

Supported Calls

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

Fields

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

Field	Details
	Description The three-letter ISO 4217 currency code associated with the usage summary record when multiple currencies are enabled.
LastActivityDate	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The timestamp of when a user had last associated a task or event action with the usage summary record.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a usage summary record indirectly, for example, through a list view or related record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a usage summary record. If this value is null, it's possible that the user only accessed the usage summary record or a related list view (LastReferencedDate), but not viewed the usage summary record itself.
Name	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. The name of the usage summary record.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. The ID of the record owner.

Field	Details
	This field is a polymorphic relationship field.
	Relationship Name Owner
	Refers To Group, User
blng__Account__c	**Type** reference
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The account of the customer who consumed the service. This field is a relationship field.
	Relationship Name blng__Account__r
	Refers To Account
blng__AppliedUnitPrice__c	**Type** currency
	Properties Create, Filter, Nillable, Sort, Update
	Description Reserved for internal use.
blng__BaseCurrencyAmount__c	**Type** currency
	Properties Create, Filter, Nillable, Sort, Update
	Description The amount of the payment in the base currency of the organization. It's used to standardize the amount across different currencies.
blng__BaseCurrencyFXDate__c	**Type** date
	Properties Create, Filter, Group, Nillable, Sort, Update
	Description The date on which the Foreign Exchange (FX) rate is applied to convert an amount into the base currency.

Field	Details
blng__BaseCurrencyFXRate__c	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The FX rate used to convert an amount from a foreign currency to the base currency.
blng__BaseCurrency__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The primary currency in which the financial transactions are recorded and reported.
blng__BilledQuantity__c	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The total quantity of this summary's usage that Salesforce Billing has billed. The billed quantity can contain usage from one or more usage records.
blng__BilledRevenue__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** Reserved for internal use.
blng__GroupedQuantity__c	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The total quantity of usage records that are grouped together for a specific order product.
blng__IncludedUnbilledUsage__c	**Type** double **Properties** Create, Filter, Nillable, Sort, Update

344

Field	Details
	Description
	Used only when the order product charge type is usage. Not used when a usage summary has an order product consumption schedule.
blng__IncludedUsageExhausted__c	**Type**
	boolean
	Properties
	Create, Defaulted on create, Filter, Group, Sort, Update
	Description
	Used only when the order product charge type is usage. Not used when a usage summary has an order product consumption schedule.
	The default value is `false`.
blng__InvoiceLine__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The invoice line that represents invoiced billings from this usage summary.
	This field is a relationship field.
	Relationship Name
	blng__InvoiceLine__r
	Refers To
	blng__InvoiceLine__c
blng__InvoiceRunProcessingStatus__c	**Type**
	picklist
	Properties
	Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	The order product status as evaluated by the invoice run.
	Valid values are:
	• `Completed`
	• `Error`
	• `In Progress`
	• `Pending Billing`
	• `Will Not Invoice`
blng__InvoiceRunStartDate__c	**Type**
	dateTime

Field	Details
	Properties Create, Filter, Nillable, Sort, Update **Description** The start date of the invoice run associated with the usage summary record.
blng__InvoiceRunStatusIndex__c	**Type** string **Properties** Create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The text string that mirrors the value of the invoice run processing status. Users must not edit this system field.
blng__InvoiceRun__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The invoice run associated with the usage summary record. This field is a relationship field. **Relationship Name** blng__InvoiceRun__r **Refers To** blng__InvoiceRun__c
blng__Invoice__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The invoice that contains an invoice line for this usage summary. This field is a relationship field. **Relationship Name** blng__Invoice__r **Refers To** blng__Invoice__c
blng__MatchingAttribute__c	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description
	Populated from the consumption schedule. Usages are rated against usage summaries where the matching ID, matching attribute, and unit of measure match. When charge type is usage, legacy usage pricing ignores this field.
blng__MatchingId__c	**Type**
	string
	Properties
	Create, Filter, Group, idLookup, Nillable, Sort, Update
	Description
	Salesforce Billing uploads usage only to usage summaries with matching IDs. Create a workflow rule that populates your usages and usage summaries based on certain criteria.
blng__NextBillingDate__c	**Type**
	date
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	The date that this usage summary is due for billing. The target date on an invoice run must be after this date for the usage summary to be invoiced. When you use bill now, the effective target date must be after this date.
blng__Notes__c	**Type**
	textarea
	Properties
	Create, Nillable, Update
	Description
	Additional information or comments related to the usage summary record.
blng__OrderItemConsumptionSchedule__c	**Type**
	reference
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	Lookup to the order item consumption schedule for pricing. Populated only for products with consumption schedules.
	This field is a relationship field.
	Relationship Name
	blng__OrderItemConsumptionSchedule__r
	Refers To
	SBQQ__OrderItemConsumptionSchedule__c

Field	Details
blng__OrderProduct__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The order product associated with the usage summary record. This association helps in tracking the usage of a product that's ordered, which is essential for billing and invoicing purposes. This field is a relationship field. **Relationship Name** blng__OrderProduct__r **Refers To** OrderItem
blng__Order__c	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The order associated with the usage summary record. This field is used to manage and track billing-related information for orders. This field is a relationship field. **Relationship Name** blng__Order__r **Refers To** Order
blng__OverrideStatus__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** When the status is set to `Canceled`, the usage summary does not load any additional usage, and the unique IDis cleared. Despite this, the usage summary can still be invoiced, as invoicing isn't blocked.
blng__OverrideSummaryEndDate__c	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description
	Salesforce Billing automatically calculates your summary's start and end dates based on your order product's next billing date and billing frequency. However, you can enter a date in this field to override the default summary end date.
blng__OverrideSummaryStartDate__c	**Type**
	date
	Properties
	Create, Filter, Group, Nillable, Sort, Update
	Description
	Salesforce Billing automatically calculates your summary's start and end dates based on your order product's next billing date and billing frequency. However, you can enter a date in this field to override the default summary start date.
blng__PreratedAmount__c	**Type**
	currency
	Properties
	Create, Filter, Nillable, Sort, Update
	Description
	The total amount value of prerated usage across all of this summary's usage records. When users enter a prerated amount on a usage record, Salesforce Billing uses that value for the usage's total rather than the standard calculation.
blng__PreratedQuantity__c	**Type**
	double
	Properties
	Create, Filter, Nillable, Sort, Update
	Description
	The total quantity value of prerated usage across all of this summary's usage records. When users enter a prerated amount on a usage record, Salesforce Billing uses that value for the usage's total rather than the standard calculation.
blng__Source__c	**Type**
	picklist
	Properties
	Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	The source of the usage summary.
	Valid values are:
	• External
	• Standard

Field	Details
blng__Status__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The status of the usage summary. Valid values are: • Cancelled • New • Processed • Queued For Group Calculation • Queued For Invoice • Queued For Subtotal Calculation
blng__Subtotal__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The subtotal amount for a usage summary record.
blng__SummaryEndDate__c	**Type** date **Properties** Create, Filter, Group, Sort, Update **Description** Required. Salesforce Billing includes usage in a usage summary if the records have matching IDs and the usage's processing date falls within the summary's start and end dates.
blng__SummaryStartDate__c	**Type** date **Properties** Create, Filter, Group, Sort, Update **Description** Required. Salesforce Billing includes usage in a usage summary if the records have matching IDs and the usage's processing date falls within the summary's start and end dates.
blng__TotalQuantity__c	**Type** double **Properties** Create, Filter, Nillable, Sort, Update

Field	Details
	Description The total quantity of the summary's usage that's consumed.
blng__TotalRevenue__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** Reserved for internal use.
blng__UnbilledQuantity__c	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The quantity of the summary's usage that isn't invoiced.
blng__UnbilledRevenue__c	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** Reserved for internal use.
blng__UniqueId__c	**Type** string **Properties** Create, Filter, Group, idLookup, Nillable, Sort, Update **Description** A unique identification number associated with the usage summary record. This field is unique within your organization.
blng__UnitOfMeasure__c	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Populated from the consumption schedule. Usages are rated against usage summaries where the matching ID, matching attribute, and unit of measure fields match. When charge type is usage, legacy usage pricing ignores this field. Valid values are:

Field	Details
	• Hours
	• None
blng__UnitPrice__c	**Type**
	currency
	Properties
	Create, Filter, Nillable, Sort, Update
	Description
	A usage summary's unit price is based on where the summary's total quantity falls within the price schedule. When you use volume price schedules, your usage records and usage summary have the same unit price. But for tiered price schedules, your usages could have different unit prices than the usage summary based on how their quantity ranges fall in each price tier.
blng__UsageCount__c	**Type**
	double
	Properties
	Create, Filter, Nillable, Sort, Update
	Description
	The number of usage records rated against this usage summary. The number doesn't change when a rated usage record is deleted.

CHAPTER 3   Salesforce Billing Developer Guides & Lightning Components

Want to get started developing with Salesforce Billing? Check out developer guides, plugins, and Lightning components.

Salesforce Billing Payment Gateway Developer Guide

Use Salesforce Billing gateway API to enable communication between Salesforce Billing and external payment gateways. Salesforce Billing can integrate with payment gateways that process credit card and ACH transactions. While we provide several default integration options, you can also use our payment gateway API to support custom integrations.

Get Started with Salesforce Billing Payment Gateway API
Use Salesforce Billing gateway API to enable communication between Salesforce Billing and external payment gateways.

Payment Gateway Class Reference
Review the classes and methods available when working with payment gateways in Salesforce Billing.

Get Started with Salesforce Billing Payment Gateway API

Use Salesforce Billing gateway API to enable communication between Salesforce Billing and external payment gateways.

Salesforce Billing Global Payment API
Salesforce Billing exposes payment transactions as global API. When you're working with our transaction API in your payment gateway, review important guidelines.

Integrating a Payment Gateway Package
Configure Salesforce Billing and your payment gateway package to communicate with an external payment gateway.

Salesforce Billing Global Payment API

Salesforce Billing exposes payment transactions as global API. When you're working with our transaction API in your payment gateway, review important guidelines.

Salesforce Billing acts as a pass-through layer for transactions. It routes the processing of input parameters to a payment gateway package and then returns the result of the transaction to the API caller.

🛑 **Important:** Salesforce Billing doesn't perform any implicit operations as a result of calling the transaction API. Our global methods provide only a layer to interact with a payment gateway to process a payment transaction. For example, calling the `chargeTransaction` global API on its own doesn't create a payment transaction object or a payment object.

All global methods in the `blng` namespace use the following input parameter structure.

`Input - Map<String, TransactionParameter> mapOfTransactionParameterById`

The map represents a unique string, such as an invoice number. The value is an instance of the `TransactionParameter` class, which contains the information needed to build a payment request.

All global methods in the `blng` namespace use the following output parameter structure.

`Output - List<TransactionResult>`

When a payment gateway sends a response following a transaction request, your payment gateway package evaluates the response and stores the information in the `TransactionResult` class.

Salesforce Billing global payment API includes the following methods.

API Method Name	Description
generateToken	Generates a token for a given input.

API Method Name	Description
authorizeTransaction	Authorizes a transaction.
captureTransaction	Captures an unauthorized transaction.
voidRefundTransaction	Voids a refund transaction.
chargeTransaction	Charges a transaction.
voidTransaction	Voids a payment transaction (charge or capture).
refundTransaction	Refunds a payment transaction for a payment that was made previously in the same gateway.
voidTokenTransaction	Voids a token.
getPaymentStatus	Identifies the status of a payment transaction.
getRefundStatus	Identifies the status of a refund transaction.
nonReferredRefund	Performs a nonreference refund.

Sending Transaction Information with TransactionParameter

The transaction parameter class represents a list of transaction field values that Salesforce Billing sends to the payment gateway package. You can set the class's attributes through Salesforce Billing global API, the Payment Center, or a payment run.

Mapping Gateway Responses to Payment Transactions with TransactionResult

The `TransactionResult` class contains a list of information that the payment gateway sends in response to a transaction request. For charge transactions, your payment gateway package evaluates this information and maps it to a Payment Transaction record in Salesforce Billing.

Sending Transaction Information with TransactionParameter

The transaction parameter class represents a list of transaction field values that Salesforce Billing sends to the payment gateway package. You can set the class's attributes through Salesforce Billing global API, the Payment Center, or a payment run.

For complete reference information on the `TransactionParameter` class's methods and parameters, review TransactionParameter Class.

This table shows how to set `TransactionParameter`'s attributes for transactions made through Salesforce Billing Global API, the Payment Center, or payment runs. Billing Global API attributes are optional unless otherwise indicated.

TransactionParameter Attribute	Source: Billing Global API	Source: Payment Center	Source: Payment Run
PaymentMethod	setPaymentMethod(paymentMethod)	Set by the Billing package.	Set by the Billing package.
Invoice	setInvoice(invoice) Required for sending Level 2 (L2) and Level 3 (L3) parameters when making a payment against an invoice.	Set by the Billing package for payments made against an invoice.	Set by the Billing package for all payments.

TransactionParameter Attribute	Source: Billing Global API	Source: Payment Center	Source: Payment Run
	Level 2 includes data such as tax details and customer codes. Level 3 includes additional details such as item descriptions and quantities. Level 2 and Level 3 payment processing is used especially for B2B and B2G transactions.		
InvoiceLine	`setInvoiceLine(line)` Sets a list of invoice lines. Required for sending L2 and L3 parameters when making a payment against one or more invoice lines.	Set by the Billing package for payments made against an invoice line.	Not set.
Account	`setAccount(accountInstance)`	Set by the Billing package.	Set by the Billing package.
RequestBody	`setRequestBody(requestBody)`	Set by the payment gateway package. Contains the request sent to the gateway.	Set by the payment gateway package. Contains the request sent to the gateway.
FirstName	`setFirstName(firstName)`	Set by the Billing package.	Set by the Billing packag.e
LastName	`setLastName(lastName)`	Set by the Billing package.	Set by the Billing package.
EmailId	`setEmailId(emailId)`	Set by the Billing package.	Set by the Billing package.
Street	`setStreet(street)`	Set by the Billing package.	Set by the Billing package.
City	`setCity(city)`	Set by the Billing package.	Set by the Billing package.
State	`setState(state)`	Set by the Billing package.	Set by the Billing package.
ZIP code	`setZipCode(zipCode)`	Set by the Billing package.	Set by the Billing package.
Country	`setCountry(country)`	Set by the Billing package.	Set by the Billing package.
Phone	`setPhone(phone)`	Set by the Billing package.	Set by the Billing package.
Amount	`setAmount(amount)`	Set by the Billing package.	Set by the Billing package.
Transaction	Not set	Not set	Set by the Billing package.
RequestingInvoiceId	`setRequestingInvoiceId(requestingInvoiceId)`	Set by the Billing package for L2 and L3 parameters.	Set by the Billing package for L2 and L3 parameters.
CardCodeResponse	`setCardCodeResponse` Required only for reference refund transactions made through Payeezy.	Set by the Billing package for reference refunds made through Payeezy.	Not set

TransactionParameter Attribute	Source: Billing Global API	Source: Payment Center	Source: Payment Run
MerchantId	`setMerchantId()`	Set by the Billing package for reference refunds made through Payeezy.	Not set
GatewayId	`setGateway(gateway)` Sets the transaction ID for the payment. Required only for reference refund transactions. Required to set by caller for Reference Refund transaction. It is transaction id for the payment.	Set by the Billing package as Gateway reference ID for payment. Used in reference refund transactions.	Not set
CurrencyId	`setCurrencyId(currencyId)` Set only for transactions made through Payeezy.	Set only for transactions made through Payeezy.	Set only for transactions made through Payeezy.
Gateway	`setGateway(gateway)` Required. Contains the ID of the Salesforce Billing payment gateway record used in the transaction.	Set by the Billing package.	Set by the Billing package.
ResponseValueByKey	Not set	Not set.	NA
Payment	`setPayment(paymentInstance)` Required to set when calling `refundTransaction`. Must query all fields from the payment instance except system fields.	Set by the Billing package when `RefundTransaction` is called. Must query all fields from the payment instance except system fields.	NA

👁 Example:

```
global class TransactionParameter
{

    global Map<string, string> mapOfResponseValueByKey = new Map<string, string>();

    // Transaction Result
    global TransactionResult transactionResult = new TransactionResult();

    global void setTransactionUseCase(TransactionUseCase transactionUseCase) {}

    global TransactionUseCase getTransactionUseCase() {}

    global void setTransactionFrequencyType(TransactionFrequency
```

```apex
transactionFrequencyType) {}

    global TransactionFrequency getTransactionFrequencyType() {}

    global void setIsCredentialsOnFile(Boolean isCredentialsOnFile) {}

    global Boolean getIsCredentialsOnFile() {}

    //Set RequestingInvoiceId
    global void setRequestingInvoiceId(Id requestingInvoiceId) {}

    //Get RequestingInvoiceId
    global Id getRequestingInvoiceId() {}

    //Set cardCodeResponse
    global void setCardCodeResponse(String cardCodeResponse) {}

    // Get cardCodeResponse
    global String getCardCodeResponse() {}

    // Set requestingCreditCardNumber
    global void setCardNumber(String requestingCreditCardNumber) {}

    // Get requestingCreditCardNumber
    global String getCardNumber() {}

    // Set Payment
    global void setPayment(Payment__c paymentInstance) {}

    // Get Payment
    global Payment__c getPayment() {}

    // Get merchantRefId
    global String getMerchantId() {}

    //Set MerchantId
    global void setMerchantId(String merchantId) {}

    //set Gateway ID
    global void setGateWayId(String gatewayId) {}

    // Get Gateway ID
    global String getGateWayId() {}

    // Get ResponseValueByKey
    global Map<string, string> getResponseValueByKey() {}

    // Set Gateway
    global void setGateWay(PaymentGateway__c gateway) {}

    // Get Gateway
    global PaymentGateway__c getGateWay() {}

    // Set PaymentMethod
```

```
global void setPaymentMethod(PaymentMethod__c paymentMethod) {}

// Get PaymentMethod
global PaymentMethod__c getPaymentMethod() {}

// Set Invoice
global void setInvoice(Invoice__c invoice) {}

// Get Invoice
global Invoice__c getInvoice() {}

// Set listOfInvoiceLine
global void setInvoiceLine(InvoiceLine__c line) {}

// Get listOfInvoiceLine
global List<InvoiceLine__c> getInvoiceLine() {}

// Set Transaction
global void setTransaction(PaymentTransaction__c transactionInstance) {}

// Get Transaction
global PaymentTransaction__c getTransaction() {}

// Set Account
global void setAccount(Account accountInstance) {}

// Get Account
global Account getAccount() {}

// Set RequestBody
global void setRequestBody(String requestBody) {}

// Get RequestBody
global String getRequestBody() {}

// Set FirstName
global void setFirstName(String firstName) {}

// Get FirstName
global String getFirstName() {}

// Set LastName
global void setLastName(String lastName) {}

// Get LastName
global String getLastName() {}

// Set EmailId
global void setEmailId(String emailId) {}

// Get EmailId
global String getEmailId() {}

// Set Street
```

```
        global void setStreet(String street) {}

        // Get Street
        global String getStreet() {}

        // Set City
        global void setCity(String city) {}

        // Get City
        global String getCity() {}

        // Set State
        global void setState(String state) {}

        // Get State
        global String getState() {}

        // Set ZipCode
        global void setZipCode(String zipCode) {}

        // Get ZipCode
        global String getZipCode() {}

        // Set Country
        global void setCountry(String country) {}

        // Get Country
        global String getCountry() {}

        // Set Phone
        global void setPhone(String phone) {}

        // Get Phone
        global String getPhone() {}

        // Set Currency Id
        global void setCurrencyId(String currencyId) {}

        // Get Currency Id
        global String getCurrencyId() {}

        // Set Amount
        global void setAmount(String amount) {}

        // Get Amount
        global String getAmount() {}

}
```

Mapping Gateway Responses to Payment Transactions with TransactionResult

The `TransactionResult` class contains a list of information that the payment gateway sends in response to a transaction request. For charge transactions, your payment gateway package evaluates this information and maps it to a Payment Transaction record in Salesforce Billing.

For complete reference information on the `TransactionResult` class's methods and parameters, review TransactionResult Class.

`TransactionResult` methods map their output to specific fields on the payment transaction object.

📝 **Note:** The payment gateway package doesn't create a payment transaction for tokenization and refund transactions.

Table 1: TransactionResult to Payment Transaction Field Mapping

Field Label	API Name	TransactionResult Method
Card Code Response	blng__CardCodeResponse__c	`setCardCodeResponse()`
Gateway Status	blng__GatewayStatus__c	`setGatewayStatus()`
Request-Type of Payment	blng__RequestTypeOfPayment__c	Internal
Response	blng__Response__c	Internal
Response Code	blng__ResponseCode__c	`setResponseCode()`
Response Gateway ID	blng__GatewayID__c	`setGatewayId()`
Response Message	blng__ResponseMessage__c	`setResponseMessage()`
Response Status	blng__ResponseStatus__c	Internal
Status	blng__Status__c	Internal
Type	blng__Type__c	Charge, Refund, Void

Integrating a Payment Gateway Package

Configure Salesforce Billing and your payment gateway package to communicate with an external payment gateway.

Configure Payment Gateways in Salesforce

Install a payment gateway package and configure your payment gateway entities to communicate with your payment gateway provider.

Salesforce Billing Payment Gateway Interfaces

A payment gateway package requires three interfaces to communicate with Salesforce Billing. Each interface contains methods that the user-defined PaymentGatewayAPI class implements to process different types of transaction actions.

Configuring Salesforce Billing to Access Your PaymentGatewayAPI Class

To integrate with a payment gateway package, Salesforce Billing must know which gateway package class contains the methods required to process a payment request and convert the results to data that Salesforce Billing can read. The Salesforce Billing custom setting Payment Gateway Config defines the class that Salesforce Billing calls from the gateway package. You must also add code that installs a record of the class in PaymentGatewayConfig.

Mapping Gateway Response Codes to Salesforce Billing Gateway Statuses

Payment gateways have many response codes for payment operations. Salesforce Billing categorizes these codes into one of seven Gateway Status field values on your payment transaction record. The gateway status lets you know the result of a gateway call and whether you must correct anything. When you configure a payment gateway package, create metadata that maps your chosen gateway's response codes to one of the statuses.

Capture Externally Authorized Payments

Authorize payments in an external system, then capture the payment in Salesforce Billing using our Capture Transaction API methods. You can then use Salesforce Billing to manage the payment lifecycle and settle invoices.

Merchant Credential Fields

Each payment gateway package installs merchant credential fields on the Salesforce Billing payment gateway object. You can use these fields to provide merchant information when making a transaction request to the payment gateway.

Required Gateway Classes

All payment gateway and payment method records require several fields to communicate with the payment gateway, regardless of the gateway type. Specific gateway types may also require additional fields.

Configure a Payment Gateway in Salesforce Billing

Configure Salesforce Billing to send data to your chosen payment gateway.

Configure Payment Gateways in Salesforce

Install a payment gateway package and configure your payment gateway entities to communicate with your payment gateway provider.

🛑 **Important:** If a custom payment gateway doesn't send the paymentGatewayToken, Salesforce Billing payment gateway API submits an extra tokenization call, which creates a new payment method.

1. Install your payment gateway package.

   Salesforce Billing provides payment gateway packages for AuthorizeDotNet, Cybersource, Payeezy, and PayFlowPro. You can also use packages for other gateways.

2. In Salesforce Billing, create a payment gateway record and set its Gateway Type field to your payment gateway. If your gateway name doesn't appear as a Gateway Type value, add it to the picklist.

3. Review the merchant credential fields on your payment gateway and enter the necessary merchant information.

   Your payment gateway record requires merchant credential fields. It sends the values of these fields as API to the payment gateway. Each payment gateway package installs different fields based on the needs of the payment gateway provider. For more information, review Merchant Credential Fields.

4. Check the payment gateway's Active field.

5. Check the payment gateway's Default field.

   The payment gateway's Default checkbox indicates that Salesforce Billing uses the gateway's merchant credentials if an ACH or credit card payment transaction doesn't provide stored credentials. Otherwise, the transaction fails. A payment gateway's Active and Default fields must be selected for the gateway entity to pass its information to the payment gateway provider.

Salesforce Billing Payment Gateway Interfaces

A payment gateway package requires three interfaces to communicate with Salesforce Billing. Each interface contains methods that the user-defined PaymentGatewayAPI class implements to process different types of transaction actions.

[Creating a PaymentGateway Interface](#)

The PaymentGateway interface contains several basic transaction methods that the `PaymentGatewayAPI` class implements to handle gateway requests. The methods allow PaymentGatewayAPI to create a credit card token, process a payment charge, void a payment, authorize a payment, and refund a payment.

[Creating a PaymentGateways Interface](#)

The `PaymentGateways` interface uses the `processPayments` method, which evaluates the payment transaction's type to perform a payment action.

[Creating a PaymentGatewayStatus Interface](#)

Salesforce Billing implements the PaymentGatewayStatus interface to populate the `TransactionResult`'s `GatewayStatus` enum with the value of the return code that the external gateway set in `transactionResult`.

Creating a PaymentGateway Interface

The PaymentGateway interface contains several basic transaction methods that the `PaymentGatewayAPI` class implements to handle gateway requests. The methods allow PaymentGatewayAPI to create a credit card token, process a payment charge, void a payment, authorize a payment, and refund a payment.

Create a credit card token

```
Map<String, TransactionResult> generateToken(Map<String, TransactionParameter>
mapOfTransactionParameterById,PaymentGatewayParameter paymentGatewayParameter);
```

Process a payment charge

```
Map<String, TransactionResult> chargeTransaction(Map<String, TransactionParameter>
mapOfTransactionParameterById,PaymentGatewayParameter paymentGatewayParameter);
```

Void a payment

```
Map<String, TransactionResult> voidTransaction(Map<String, TransactionParameter>
mapOfTransactionParameterById,PaymentGatewayParameter paymentGatewayParameter);
```

Authorize a payment

```
Map<String, TransactionResult> authorizeTransaction(Map<String, TransactionParameter>
mapOfTransactionParameterById,PaymentGatewayParameter paymentGatewayParameter);
```

Refund a payment

```
Map<String, TransactionResult> refundTransaction(Map<String, TransactionParameter>
mapOfTransactionParameterById,PaymentGatewayParameter paymentGatewayParameter);
```

So, if you want your gateway package to process a `chargeTransaction` sent from Salesforce Billing, the package must implement `chargeTransaction` within a `PaymentGatewayAPI` class. Let's see how this configuration looks in a sample CyberSource implementation.

👁 Example:

```
global class CyberSourceAPI implements blng.PaymentGateWay
{
    global static Map<String, blng.TransactionResult> generateToken(Map<String,
blng.TransactionParameter> mapOfTransactionParameterById)
    {
        return service().generateToken(mapOfTransactionParameterById);
    }

    global static Map<String, blng.TransactionResult> authorizeTransaction(Map<String,
 blng.TransactionParameter> mapOfTransactionParameterById)
    {
```

```
            return service().authorizeTransaction(mapOfTransactionParameterById);
        }

        global static Map<String, blng.TransactionResult> chargeTransaction(Map<String,
blng.TransactionParameter> mapOfTransactionParameterById)
        {
            return service().chargeTransaction(mapOfTransactionParameterById);
        }

        global static Map<String, blng.TransactionResult> voidTransaction(Map<String,
blng.TransactionParameter> mapOfTransactionParameterById)
        {
            return service().voidTransaction(mapOfTransactionParameterById);
        }

        global static Map<String, blng.TransactionResult> refundTransaction(Map<String,
blng.TransactionParameter> mapOfTransactionParameterById)
        {
            return service().refundTransaction(mapOfTransactionParameterById);
        }
}
```

Creating a PaymentGateways Interface

The `PaymentGateways` interface uses the `processPayments` method, which evaluates the payment transaction's type to perform a payment action.

The `PaymentGateways` interface calls the `processPayments` method, which evaluates the TransactionParameter object.

```
global interface PaymentGateways
{
    Map<String, TransactionResult> processPayments(Map<String, TransactionParameter>
mapOfTransactionParameterById,PaymentGatewayParameter paymentGatewayParameter);
}
```

The `processPayments` method can perform different actions based on the `TransactionType` value of the TransactionParameter's `paymentGatewayParameter`. When you configure your `PaymentGateways` interface, define all the `TransactionType` string variables that `processPayments` can accept. Then, add your logic for identifying the transaction type and calling the appropriate method. We've provided a sample here for a CyberSource payment gateway implementation.

👁 Example:

```
global class CyberSourceAPI implements blng.PaymentGateWay, blng.PaymentGateWays,
blng.PaymentGateWayStatus
{
    // ==========================================================================
    // CONSTANT
    // ==========================================================================

    // ==========================================================================
    // STATIC VARIABLES
    // ==========================================================================

    // Attribute to implement singleton pattern for CyberSource class
```

```
    private static CyberSource CyberSourceInstance;

    // ===========================================================================
    // VARIABLES
    // ===========================================================================
    //Billing sends following strings for the transactions
    private static final String GENERATE_TOKEN = 'generateToken';
    private static final String AUTHORIZE_TRANSACTION = 'authorizeTransaction';
    private static final String CHARGE_TRANSACTION = 'chargeTransaction';
    private static final String VOID_TRANSACTION = 'voidTransaction';
    private static final String REFUND_TRANSACTION = 'refundTransaction';
    private static final String CAPTURE_TRANSACTION = 'captureTransaction';
    private static final String VOID_REFUND_TRANSACTION = 'voidRefundTransaction';
    private static final String VOID_TOKEN_TRANSACTION = 'voidTokenTransaction';
    private static final String GET_PAYMENT_STATUS = 'getPaymentStatus';
    private static final String GET_REFUND_STATUS = 'getRefundStatus';
    private static final String NON_REFERRED_REFUND = 'nonReferredRefund';
    private static final String PROCESS_PAYMENTS = 'processPayments';
    // ===========================================================================
    // Methods
    // ===========================================================================

    global static Map<String, blng.TransactionResult> processPayments(Map<String,
blng.TransactionParameter> mapOfTransactionParameterById, blng.PaymentGatewayParameter
 paymentGatewayParameter)
    {
        if(paymentGatewayParameter.getTransactionType() == GENERATE_TOKEN)
        {
            return generateToken(mapOfTransactionParameterById);
        }
        else if(paymentGatewayParameter.getTransactionType() == AUTHORIZE_TRANSACTION)

        {
            return authorizeTransaction(mapOfTransactionParameterById);
        }
        else if(paymentGatewayParameter.getTransactionType() == CHARGE_TRANSACTION)
        {
            return chargeTransaction(mapOfTransactionParameterById);
        }
        else if(paymentGatewayParameter.getTransactionType() == VOID_TRANSACTION)
        {
            return voidTransaction(mapOfTransactionParameterById);
        }
        else if(paymentGatewayParameter.getTransactionType() == REFUND_TRANSACTION)
        {
            return refundTransaction(mapOfTransactionParameterById);
        }
        else if(paymentGatewayParameter.getTransactionType() == CAPTURE_TRANSACTION)
        {
            return captureTransaction(mapOfTransactionParameterById);
        }
        else if(paymentGatewayParameter.getTransactionType() == VOID_REFUND_TRANSACTION)

        {
```

```
                    return voidRefundTransaction(mapOfTransactionParameterById);
            }
        else if(paymentGatewayParameter.getTransactionType() == VOID_TOKEN_TRANSACTION)

            {
                return voidTokenTransaction(mapOfTransactionParameterById);
            }
            else if(paymentGatewayParameter.getTransactionType() == GET_PAYMENT_STATUS)
            {
                return getPaymentStatus(mapOfTransactionParameterById);
            }
            else if(paymentGatewayParameter.getTransactionType() == GET_REFUND_STATUS)
            {
                return getRefundStatus(mapOfTransactionParameterById);
            }
            else if(paymentGatewayParameter.getTransactionType() == NON_REFERRED_REFUND)
            {
                return nonReferredRefund(mapOfTransactionParameterById);
            }

            return NULL;
    }
}
```

Creating a PaymentGatewayStatus Interface

Salesforce Billing implements the PaymentGatewayStatus interface to populate the `TransactionResult`'s `GatewayStatus` enum with the value of the return code that the external gateway set in `transactionResult`.

```
global interface PaymentGateWayStatus
{
    void populateGatewayStatus(TransactionResult transactionResult);
}
```

So, if you want your gateway package to process a `chargeTransaction` sent from Salesforce Billing, the package must implement `chargeTransaction` within a `PaymentGatewayAPI` class. Let's see how this configuration looks in a sample CyberSource implementation.

```
global class CyberSourceAPI implements blng.PaymentGateWay, blng.PaymentGateWays,
blng.PaymentGateWayStatus
{
    /**
    Implement other methods as described above
    */
    private static CyberSource service()
    {
        if (NULL == CyberSourceInstance)
        {
            CyberSourceInstance = CyberSource.getInstance();
        }
        return CyberSourceInstance;
    }
    /**
    * @name populateGatewayStatus
```

```
    * @description Method that populates the GatewayStatus enum on the TransactionResult,
    * given return codes that are already set in the TransactionResults
    * @param transactionResult
    */
    global static void populateGatewayStatus(blng.TransactionResult transactionResult)
    {
        service().populateGatewayStatus(transactionResult);
    }
}
```

Configuring Salesforce Billing to Access Your PaymentGatewayAPI Class

To integrate with a payment gateway package, Salesforce Billing must know which gateway package class contains the methods required to process a payment request and convert the results to data that Salesforce Billing can read. The Salesforce Billing custom setting Payment Gateway Config defines the class that Salesforce Billing calls from the gateway package. You must also add code that installs a record of the class in PaymentGatewayConfig.

First, create your primary gateway package class. For ease of reference, name it after your payment gateway, followed by "API." The class must implement all three of our payment gateway interfaces. Here's an example for CyberSource.

```
global class CyberSourceAPI implements blng.PaymentGateWay, blng.PaymentGateWays,
blng.PaymentGateWayStatus
{}
```

Next, add code that lets you install a record of the class in PaymentGatewayConfig. For this example, we've titled our method `insertPaymentGatewayConfigCustomSettings`.

```
public void insertPaymentGatewayConfigCustomSettings()
{
    String gatewayName = 'CyberSource';
    List<blng__PaymentGatewayConfig__c> listofConfiguration = new
List<blng__PaymentGatewayConfig__c>();
    blng__PaymentGatewayConfig__c gateway = new blng__PaymentGatewayConfig__c(
            Name = gatewayName,
            blng__ClassName__c = CyberSource.class.getName().substringBefore(gatewayName)
 + gatewayName + 'API');
    listofConfiguration.add(gateway);

    // get the all custom settings to check whether system has already settings or not
    Map<String, blng__PaymentGatewayConfig__c> mapOfConfigurationValues =
blng__PaymentGatewayConfig__c.getAll();
    // if system does not have the gateway, then insert it
    if(!mapOfConfigurationValues.containskey(gatewayName)) {
        insert listofConfiguration;
    }
}
```

Mapping Gateway Response Codes to Salesforce Billing Gateway Statuses

Payment gateways have many response codes for payment operations. Salesforce Billing categorizes these codes into one of seven Gateway Status field values on your payment transaction record. The gateway status lets you know the result of a gateway call and whether you must correct anything. When you configure a payment gateway package, create metadata that maps your chosen gateway's response codes to one of the statuses.

First, let's look at the values for the payment transaction's Status field in Salesforce Billing. When your gateway integration sets Status to a value, Salesforce Billing attempts to perform the action listed in the Salesforce Billing Action column in the following table.

Response	Definition	Salesforce Billing Action
Success	The gateway call succeeded.	Attempt payment creation and allocation.
Decline	The gateway call failed, but it can still work if you try again. For example, the customer had insufficient funds or briefly lost their connection to the internet. This response is also known as a "soft decline."	Update the payment run's decline count. Create payment transaction with gateway status of Decline.
Validation Error	Customer payment data is incorrect, such a misspelling in the credit card address or incorrect CVV.	Create a payment transaction with a gateway status of Validation Error.
Permanent Fail	The gateway call failed and won't work even if you try again. If you receive this response, don't make further payments using the related payment method.	Create a payment transaction with a gateway status of Permanent Fail.
Requires Review	The customer bank requires additional information before completing the payment.	Create a payment transaction with gateway status of Requires Review.
Indeterminate	The gateway didn't respond to the call. This response usually happens when Salesforce Billing times out waiting for the response. This response doesn't increase the payment run's failure count.	Lock the invoice from further payment runs. Create a payment transaction with a gateway status of Indeterminate. Allow users to autovoid or manually create a payment.
System Fail	Salesforce Billing ended the payment request call before receiving a response. For example, Salesforce Billing lost credentials or lost access to its server. This response increases the payment run's failure count. Salesforce Billing ends payment calls if it doesn't receive a response from the gateway in two minutes.	Create a payment transaction with a gateway status of System Fail. Allow users to autovoid or manually create a payment.

Response codes and descriptions vary between payment gateway providers. Let's look at a few examples and see how they can map to a gateway status.

Gateway	Code	Gateway's Description	Salesforce Billing Equivalent
Payeezy	100	"Approved"	Success
CyberSource	151	"The request was received, but a server timeout occurred. This error doesn't include timeouts	Indeterminate

Gateway	Code	Gateway's Description	Salesforce Billing Equivalent
		between the client and the server." "To avoid duplicating the transaction, don't resend the request until you have reviewed the transaction status at the Business Center. See the documentation for your CyberSource client for more information."	
PayFlowPro	4	"Invalid amount format. Use the format '#####.##' Don't include currency symbols or commas."	System
AuthorizeDotNet	E00057	"The user doesn't have permissions to submit requests from a mobile device."	Fail

[Configure Response Code Metadata in Your Payment Gateway Package](#)

Payment gateway packages require a custom metadata type file to communicate with Salesforce Billing. The custom metadata type defines the custom object and fields that the gateway package uses to store request and response information sent from the payment gateway. Your custom metadata type file evaluates the value of your gateway package's custom gateway field. Then, the file must convert the gateway field to a value that Salesforce Billing sends to the payment transaction's Status field.

[Configuring Salesforce Billing to Read Your Mapping Metadata](#)

After you've set up your mapping metadata, configure your payment gateway adapter Apex class to read the gateway status and assign it to `TransactionResult`.

Configure Response Code Metadata in Your Payment Gateway Package

Payment gateway packages require a custom metadata type file to communicate with Salesforce Billing. The custom metadata type defines the custom object and fields that the gateway package uses to store request and response information sent from the payment gateway. Your custom metadata type file evaluates the value of your gateway package's custom gateway field. Then, the file must convert the gateway field to a value that Salesforce Billing sends to the payment transaction's Status field.

The fields used to store gateway request and response values vary between payment gateway packages. Let's look at a few examples from the custom metadata created for the AuthorizeDotNet, CyberSource, and Payeezy packages. In each, we provide a recommended metadata name.

Package	Metadata Name	API Names	Field Labels
AuthorizeDotNet	`adnblng_GatewayStatusMapping_mdt`	• Response from Gateway • Additional Gateway Response	• `adnblng_ResponseCode__c` • `adnblng_ResponseCode2__c` • `adnblng_GatewayStatus__c`

Package	Metadata Name	API Names	Field Labels
		• Corresponding Gateway Status	
CyberSource	csblng__GatewayStatusMapping__mdt	• Response from Gateway • Corresponding Gateway Status	• csblng_ResponseCode__c • csblng__GatewayStatus__c
Payeezy	pyzblng__GatewayStatusMapping__mdt	• Response from Gateway • Corresponding Gateway Status	• pyzblng_ResponseCode__c • pyzblng__GatewayStatus__c

👁 **Example:** Cybersource's response code 100 represents a successful communication with the payment gateway. Let's look at a metadata file that evaluates the Cybersource gateway package field `ResponseCode__c` for a value of *100* and then sets the payment transaction's status field to Success.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<CustomMetadata xmlns="http://soap.sforce.com/2006/04/metadata"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <label>100</label>
    <values>
        <field>ResponseCode__c</field>
        <value xsi:type="xsd:string">100</value>
    </values>
    <values>
        <field>GatewayStatus__c</field>
        <value xsi:type="xsd:string">Success</value>
    </values>
</CustomMetadata>
```

👁 **Example:** Some payment gateways return multiple response codes for one gateway transaction. Responses with multiple codes can occur when the gateway processes the transaction successfully but then the customer card provider responds with another value. For example, the AuthorizeDotNet code *I00001* indicates that the payment gateway successfully processed a request. However, the card provider can then send a response code of *3*, which indicates that the transaction was declined. The provider can also send a response code of *7*, which indicates that the credit card expiration date is invalid. Let's look at metadata samples for processing *I00001 – 3* and *I00001 – 7* respectively.

In this case, we assign *I00001 – 3* to a Salesforce Billing value of Permanent Fail.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<CustomMetadata xmlns="http://soap.sforce.com/2006/04/metadata"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<label>I00001-3</label>
     <values>
```

```xml
            <field>ResponseCode__c</field>
            <value xsi:type="xsd:string">I00001</value>
        </values>
        <values>
            <field>ResponseCode2__c</field>
            <value xsi:type="xsd:string">3</value>
        </values>
        <values>
            <field>GatewayStatus__c</field>
            <value xsi:type="xsd:string">PermanentFail</value>
        </values>
</CustomMetadata>
```

In this case, we assign `I00001 - 7` to a Salesforce Billing value of Decline.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<CustomMetadata xmlns="http://soap.sforce.com/2006/04/metadata"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<label>I00001-2</label>
        <values>
            <field>ResponseCode__c</field>
            <value xsi:type="xsd:string">I00001</value>
        </values>
        <values>
            <field>ResponseCode2__c</field>
            <value xsi:type="xsd:string">2</value>
        </values>
        <values>
            <field>GatewayStatus__c</field>
            <value xsi:type="xsd:string">Decline</value>
        </values>
</CustomMetadata>
```

Configuring Salesforce Billing to Read Your Mapping Metadata

After you've set up your mapping metadata, configure your payment gateway adapter Apex class to read the gateway status and assign it to `TransactionResult`.

First, define a `GatewayStatusType` enum.

```apex
global Enum GatewayStatusType { Success,
                                Decline,
                                ValidationError,
                                PermanentFail,
                                RequiresReview,
                                Indeterminate,
                                SystemError }

// Set gatewayStatus
global void setGatewayStatus(GatewayStatusType gatewayStatus)
{
    this.gatewayStatus = gatewayStatus;
}
```

```
// Get gatewayStatus
global GatewayStatusType getGatewayStatus()
{
    return gatewayStatus;
}
```

Next, add code that reads your GatewayStatusMapping metadata and sets the correct gateway status in your TransactionResult object. Here's one way you can configure it.

```
private static Map<string, blng.TransactionResult.GatewayStatusType>
mapGatewayStatusEnumTypesByStrings =
                new Map<string, blng.TransactionResult.GatewayStatusType>();
static {
    List<blng.TransactionResult.GatewayStatusType> enumValues =
blng.TransactionResult.GatewayStatusType.values();
    for (Integer i = 0; i < enumValues.size(); i++) {
        mapGatewayStatusEnumTypesByStrings.put(enumValues.get(i).name(), enumValues.get(i));

    }
}

// default Gateway Status - It should be always Indeterminate to avoid duplicate payments
private static final blng.TransactionResult.GatewayStatusType defaultGatewayStatus =
blng.TransactionResult.GatewayStatusType.Indeterminate;

/**
* Does a SOQL lookup on the mapper table and gets the gateway status mapped to the return
 code
* Returns the default enum if no match is found
* @param transactionResult
*/
public void populateGatewayStatus(blng.TransactionResult transactionResult) {
    blng.TransactionResult.GatewayStatusType gatewayStatus = defaultGatewayStatus;
    String returnCode = transactionResult.getResponseCode();
    if (returnCode != null) {
        List<GatewayStatusMapping__mdt> gatewayStatusMaps = [
                SELECT GatewayStatus__c
                FROM GatewayStatusMapping__mdt
                WHERE ResponseCode__c = :returnCode
                LIMIT 1
        ];
        if (!gatewayStatusMaps.isEmpty()) {
            gatewayStatus =
mapGatewayStatusEnumTypesByStrings.get(gatewayStatusMaps.get(0).GatewayStatus__c);
            if (gatewayStatus == null) {
                gatewayStatus = defaultGatewayStatus;
            }
        }
    }
    transactionResult.setGatewayStatus(gatewayStatus);
}
```

Capture Externally Authorized Payments

Authorize payments in an external system, then capture the payment in Salesforce Billing using our Capture Transaction API methods. You can then use Salesforce Billing to manage the payment lifecycle and settle invoices.

Delayed capture is useful when the payment is authorized at the time of sale and settled after an event occurs. For example, you can capture a payment after a subscription is provisioned or an item is delivered.

> ⛔ **Important:** Our Capture Payment API works out of the box with the Salesforce Billing Cybersource gateway integration package. Customers using other payment gateways must work with their payment gateway vendors to extend the gateway integration packages to the Capture Payment API.

EDITIONS

Available in: Salesforce Billing Winter '21 and later

Managing Payment Authorizations

Salesforce Billing doesn't process payment authorizations, but you can store information from an external authorization in a Salesforce Billing payment authorization record. To store external authorization information in Salesforce, add information from the external authorization to the following fields in the Salesforce Billing payment authorization record:

- `Account ID`
- `Original Amount`
- `Currency ISO Code` (for multi-currency orgs only)
- `Payment Gateway ID`
  - Use the ID of the Salesforce Billing payment gateway record that you use for storing payment capture information. The payment gateway record must have the same merchant credentials used to process the external payment authorization.
- `Gateway Reference Number`
  - The payment gateway supplies the reference number.
- `Status` this field must have the value `Processed`

Calling the Capture Transaction API

After you've created a payment authorization, you can pass it to the Capture Transaction API. To call the Capture Transaction API, add the following method to your `PaymentGateway` Apex class.

```
List<blng.OutputResult> result = blng.TransactionAPI.captureTransaction(lcip);
```

The CaptureTransaction API accepts information through the `CaptureInputParameter` class.

Table 2: CaptureInputParameter Parameters

Parameter	Description	Type	Required
`PaymentAuthorization`	ID of the payment authorization record from Salesforce Billing.	ID	Yes
`captureAmount`	Amount of payment to capture from the payment.	Decimal	Yes
`additionalParameters`	If your capture request requires more information, you can provide a map of additional	Map<String, String>	No

373

Parameter	Description	Type	Required
	key-value pairs to the gateway adapter		

The `CaptureOutputResult` class returns the results of your capture request. If you successfully capture the payment, Salesforce Billing creates a posted payment record related to your payment authorization. The CaptureTransaction API also sends a capture success message. If the capture fails, the CaptureTransaction API sends a capture failure message.

Table 3: CaptureOutputResult Parameters

Parameter	Description
isSuccess	Shows whether the payment capture request succeeded.
paymentID	The Salesforce Billing payment record that the CaptureAPI created as a result of successfully capturing a payment from the gateway.
paymentTransactionID	The Salesforce Billing payment transaction record containing information about the gateway interaction for the capture request.
errorMessage	If the payment capture was unsuccessful, contains information about the error.

Permissions

To perform a capture transaction, users require access to the following fields.

Payment

Write access on the following fields:

- Account
- Amount
- Status
- Payment Authorization
- Payment Gateway
- Payment Transaction
- Payment Gateway ID

Payment Transaction

Write access on the following fields:

- Account
- Amount
- Type
- Source Transaction ID
- Gateway Request
- Gateway Response
- Payment Gateway
- Gateway Status

- `Response`
- `Response Code`
- `Response Message`
- `Payment Gateway ID`
- `Gateway Date`
- `Status`

Referenced Classes

- `CaptureInputParameter` on page 380
- `InputParameter` on page 385
- `CaptureOutputResult` on page 382
- `OutputResult` on page 386

👁 Example:

Request

In this example, `CaptureInputParameter` receives the payment authorization record from Salesforce Billing, a payment capture amount of 4.23, and additional parameters. The parameters are stored in a list and sent to `TransactionAPI`, which performs the payment capture. `CaptureOutputResult` contains information about the capture.

```
String authId = 'a1wxxxxxxxxxxxxxxx';
Map<String, Schema.SObjectField> fieldMap =
blng__PaymentAuthorization__c.sObjectType.getDescribe().fields.getMap();
Set<String> fieldNames = fieldMap.keySet();
SObject myrecord = Database.query('select ' +
String.join((Iterable<String>)fieldNames, ',') + ' from blng__PaymentAuthorization__c
 Where Id = \''+authId+'\'');
Map<String,String> additionalParams = new Map<String,String>();
additionalParams.put('merchantReferenceCode','demoCapture');

blng.CaptureInputParameter cip = new blng.CaptureInputParameter();
cip.setPaymentAuthorization((blng__PaymentAuthorization__c)myrecord);
cip.setAmount(4.23);
cip.setAdditionalParameters(additionalParams);
List<blng.CaptureInputParameter> lcip = new List<blng.CaptureInputParameter>();
lcip.add(cip);
List<blng.CaptureOutputResult> result = blng.TransactionAPI.captureTransaction(lcip);
system.debug(result[0]);
```

Success Response

Following a successful payment capture, `CaptureOutputResult` returns a success response with the IDs of the payment transaction and payment created in Salesforce Billing.

```
[CaptureOutputResult.errorMessage=null, CaptureOutputResult.isSuccess=true,
paymentId=a25xxxxxxxxxxxxxxx, paymentTransactionId=a24xxxxxxxxxxxxxxx]
```

Failure Response

Following a failed payment capture, `CaptureOutputResult` returns a failure response with an error message and the ID of the payment transaction created in Salesforce Billing. The payment transaction contains information about the failed gateway communication.

```
[CaptureOutputResult.errorMessage=Invalid data. c:authRequestID,
CaptureOutputResult.isSuccess=false, paymentId=null,
paymentTransactionId=a24xxxxxxxxxxxxxx]
```

Merchant Credential Fields

Each payment gateway package installs merchant credential fields on the Salesforce Billing payment gateway object. You can use these fields to provide merchant information when making a transaction request to the payment gateway.

Table 4: AuthorizeDotNet Merchant Credential Fields

Attribute Name	API Name	Required	Type
APILoginId	adnblng__APILoginId__c	No	Text
APITransactionKey	adnblng__APITransactionKey__c	No	Text
TestMode	adnblng__TestMode__c	No	Checkbox

Table 5: CyberSource Merchant Credential Fields

Attribute Name	API Name	Required	Type
Merchant ID	csblng__MerchantId__c	No	Text
Merchant Reference	csblng__MerchantReference__c	No	Text
Test Mode	csblng__TestMode__c	No	Checkbox
Transaction Security Key	csblng__TransactionSecurityKey__c	No	Text

Table 6: Payeezy Merchant Credential Fields

Attribute Name	API Name	Required	Type
APIKey	pzyblng__APIKey__c	No	Text
APILoginId	pyzblng__APILoginId__c	No	Text
APISecret	pyzblng__APISecret__c	No	Text
APITransactionKey	pyzblng__APITransactionKey__c	No	Text
Enable User Identification	pyzblng__EnableUserIdentification__c	No	Checkbox
JSSecurityKey	pyzblng__JSSecurityKey__c	No	Text
TestMode	pyzblng__TestMode__c	No	Text
Token	pyzblng__Token__c	No	Text

Attribute Name	API Name	Required	Type
TransarmorToken	pyzblng__TransarmorToken__c	No	Text

Required Gateway Classes

All payment gateway and payment method records require several fields to communicate with the payment gateway, regardless of the gateway type. Specific gateway types may also require additional fields.

Payment Method Fields

Label	API Name
Payment Method Name	Name
Account	blng__Account__c
Active	blng__Active__c
Auto Pay	blng__AutoPay__c
Bank Account Name	blng__BankAccountName__c
Bank Account Number	blng__BankAccountNumber__c
Bank Account Type	blng__BankAccountType__c
Bank Name	blng__BankName__c
Bank Routing Code	blng__BankRoutingCode__c
Card BIN	blng__CardBIN__c
Card CVV	blng__CVV__c
Card Expiration Month	blng__CardExpirationMonth__c
Card Expiration Year	blng__CardExpirationYear__c
Card Last 4	blng__CardLastFour__c
Card Number	blng__CardNumber__c
Card Type	blng__CardType__c
City	blng__ChequeNumber__c
Company	blng__BillingCity__c
Country	blng__BillingCompany__c
Email	blng__BillingEmail__c
Fax	blng__BillingFax__c
First Name	blng__BillingFirstName__c

Gateway Response	blng__GatewayResponse__c
Last Name	blng__BillingLastName__c
Name on Card	blng__Nameoncard__c
Nick Name	blng__NickName__c
Payment Gateway	blng__PaymentGateway__c
Payment Gateway Token	blng__PaymentGatewayToken__c
Payment Type	blng__PaymentType__c
Phone Number	blng__BillingPhone__c
Postal Code	blng__BillingZipPostal__c
State	blng__BillingStateProvince__c
Street Address 1	blng__BillingStreet__c
Street Address 2	blng__StreetAddress2__c

Payment Gateway Fields

Gateway integrations require the following payment gateway record fields regardless of gateway type. If you have a field required by a specific gateway such as Authorize.Net, make sure to add that field to your payment gateway record as well.

Label	API Name
Payment Gateway Name	Name
Active	blng__Active__c
Default	blng__Default__c
Gateway Type	blng__GatewayType__c

Configure a Payment Gateway in Salesforce Billing

Configure Salesforce Billing to send data to your chosen payment gateway.

1. Set up remote site settings.

   a. From Setup, enter `Remote Site Settings`, and then select **Remote Site Settings**.

   b. Click **New Remote Site**.

   c. Add the domain production URL and domain sandbox URL for your gateway.

      We provide a list of URLs for our default-supported gateways, but you can add others based on your custom gateway's requirements.

EDITIONS

Available in: Salesforce Billing 7.0 and later

Gateway	Production URL	Sandbox URL
Authorize.net	https://api.authorize.net	https://apitest.authorize.net

378

Gateway	Production URL	Sandbox URL
CyberSource	https://ics2wsa.ic3.com	https://icswstesta.ic3.com
Payeezy	https://api.payeezy.com	https://api-cert.payeezy.com

2. Set up custom settings.

   a. From Setup, enter `Custom Settings`, and then select **Custom Settings**.

   b. Click **Payment Gateway Config**, then click **Manage**.

   c. Click **New**.

   d. Enter a name for your payment gateway.

      Match this value to the name on your payment gateway record's Gateway Type field.

   e. Enter your gateway class name.

      This value takes the form of `PackagePrefix.APIClassName`. We provide a list of class names for our default gateways, but you can add others based on your custom gateway's requirements.

Name	Gateway Class Name
AuthorizeDotNet	AuthorizeDotNetAPI
CyberSource	CyberSourceAPI
Payeezy	PayeezyAPI

3. Save your changes.

Payment Gateway Class Reference

Review the classes and methods available when working with payment gateways in Salesforce Billing.

CaptureInputParameter Class

Receives the payment authorization record, payment capture amount, and an optional list of additional parameters from Salesforce Billing.

CaptureOutputResult Class

Stores the results of the payment capture request.

Payment Gateway Core Classes

Your payment gateway references these classes to receive transaction information from Salesforce Billing. They can't be modified.

Custom Gateway Classes

Salesforce Billing provides three classes that users can customize to integrate with their chosen payment gateway.

CaptureInputParameter Class

Receives the payment authorization record, payment capture amount, and an optional list of additional parameters from Salesforce Billing.

Namespace

blng

Example

In this example, t `CaptureInputParameter` receives the payment authorization record from Salesforce Billing, a payment capture amount of 4.23, and additional parameters. The parameters are stored in a list and sent to the `TransactionAPI` class, which performs the payment capture. The `CaptureOutputResult` class contains the result of the capture.

```
blng.CaptureInputParameter cip = new blng.CaptureInputParameter();
cip.setPaymentAuthorization((blng__PaymentAuthorization__c)myrecord);
cip.setAmount(4.23);
cip.setAdditionalParameters(additionalParams);
List<blng.CaptureInputParameter> lcip = new List<blng.CaptureInputParameter>();
lcip.add(cip);
List<blng.CaptureOutputResult> result = blng.TransactionAPI.captureTransaction(lcip);
system.debug(result[0]);
```

CaptureInputParameter Methods

CaptureInputParameter Methods

The following are methods for `CaptureInputParameter`.

setPaymentAuthorization(paymentAuthorization)
Sets the ID of the payment authorization.

getPaymentAuthorization()
Returns the payment authorization record from Salesforce Billing.

setAmount(amount)
Sets the amount of the payment capture.

getAmount()
Returns the amount of the payment capture to be performed.

`setPaymentAuthorization(paymentAuthorization)`

Sets the ID of the payment authorization.

Signature

```
global static void setPaymentAuthorization(blng__PaymentAuthorization__c
paymentAuthorization)
```

Parameters

paymentAuthorization

    Type: `blng__PaymentAuthorization__c`

    The payment authorization record from Salesforce Billing.

Return Value

Type: void

getPaymentAuthorization()

Returns the payment authorization record from Salesforce Billing.

Signature

```
global static blng__PaymentAuthorization__c getPaymentAuthorization()
```

Return Value

Type: blng__PaymentAuthorization__c

setAmount(amount)

Sets the amount of the payment capture.

Signature

```
global static void setAmount(Decimal amount)
```

Parameters

amount

    Type: Decimal

    The amount of the payment capture. Can be none, part, or all of the payment.

Return Value

Type: void

getAmount()

Returns the amount of the payment capture to be performed.

Signature

```
global static Decimal getAmount()
```

Return Value

Type: Decimal

CaptureOutputResult Class

Stores the results of the payment capture request.

Namespace

blng

Example

Example:

Request

In this example, `CaptureInputParameter` receives the payment authorization record from Salesforce Billing, a payment capture amount of 4.23, and additional parameters. The parameters are stored in a list and sent to `TransactionAPI`, which performs the payment capture. `CaptureOutputResult` contains information about the capture.

```
String authId = 'a1wxxxxxxxxxxxxxxx';
Map<String, Schema.SObjectField> fieldMap =
blng__PaymentAuthorization__c.sObjectType.getDescribe().fields.getMap();
Set<String> fieldNames = fieldMap.keySet();
SObject myrecord = Database.query('select ' +
String.join((Iterable<String>)fieldNames, ',') + ' from blng__PaymentAuthorization__c
 Where Id = \''+authId+'\'');
Map<String,String> additionalParams = new Map<String,String>();
additionalParams.put('merchantReferenceCode','demoCapture');

blng.CaptureInputParameter cip = new blng.CaptureInputParameter();
cip.setPaymentAuthorization((blng__PaymentAuthorization__c)myrecord);
cip.setAmount(4.23);
cip.setAdditionalParameters(additionalParams);
List<blng.CaptureInputParameter> lcip = new List<blng.CaptureInputParameter>();
lcip.add(cip);
List<blng.CaptureOutputResult> result = blng.TransactionAPI.captureTransaction(lcip);
system.debug(result[0]);
```

Success Response

Following a successful payment capture, `CaptureOutputResult` returns a success response with the IDs of the payment transaction and payment created in Salesforce Billing.

```
[CaptureOutputResult.errorMessage=null, CaptureOutputResult.isSuccess=true,
paymentId=a25xxxxxxxxxxxxxxx, paymentTransactionId=a24xxxxxxxxxxxxxxx]
```

Failure Response

Following a failed payment capture, `CaptureOutputResult` returns a failure response with an error message and the ID of the payment transaction created in Salesforce Billing. The payment transaction contains information about the failed gateway communication.

```
[CaptureOutputResult.errorMessage=Invalid data. c:authRequestID,
CaptureOutputResult.isSuccess=false, paymentId=null,
paymentTransactionId=a24xxxxxxxxxxxxxxx]
```

CaptureOutputResult Methods

CaptureOutputResult Methods

The following are methods for `CaptureOutputResult`.

setPaymentId(paymentId)
Sets the ID of the Salesforce Billing payment record created after a successful capture transaction.

getPaymentId()
Returns the ID of the Salesforce Billing payment record that was created after a successful capture transaction.

setPaymentTransactionId(paymentTransactionId)
Sets the ID of the Salesforce Billing payment transaction record that was created after a successful capture transaction.

getPaymentTransactionId()
Returns the ID of the Salesforce Billing payment transaction record created after a successful capture transaction.

setPaymentId(paymentId)

Sets the ID of the Salesforce Billing payment record created after a successful capture transaction.

Signature

```
global static void setPaymentId(Id paymentId)
```

Parameters

paymentId
Type: Id

ID of the payment record created in Salesforce Billing. The payment record stores information about the payment.

Return Value

Type: void

getPaymentId()

Returns the ID of the Salesforce Billing payment record that was created after a successful capture transaction.

Signature

```
global static Id getPaymentId()
```

Return Value

Type: Id

setPaymentTransactionId(paymentTransactionId)

Sets the ID of the Salesforce Billing payment transaction record that was created after a successful capture transaction.

383

Signature

```
public static void setPaymentTransactionId(Id paymentTransactionId)
```

Parameters

paymentTransactionId

Type: Id

ID of the payment transaction record created in Salesforce Billing. This record stores the results of the communication with the payment gateway.

Return Value

Type: void

getPaymentTransactionId()

Returns the ID of the Salesforce Billing payment transaction record created after a successful capture transaction.

Signature

```
global static Id getPaymentTransactionId()
```

Return Value

Type: Id

Payment Gateway Core Classes

Your payment gateway references these classes to receive transaction information from Salesforce Billing. They can't be modified.

The following are the `blng` namespace classes required for payment gateway integration.

InputParameter Class

Captures and retrieves additional parameters to be sent during the capture payment transaction.

OutputResult Class

Stores and shows the results of the capture payment request made to the payment gateway.

PaymentGateway Interface

Interface for standard payment transaction operations such as voids and refunds.

PaymentGateways Interface

Interface with common payment gateway operation across all gateways.

PaymentGatewayParameter Class

Global parameter class for the payment gateway.

PaymentGatewayStatus Interface

Interface to populate the TransactionResult value object from the gateway response.

TransactionAPI Class

The TransactionAPI class includes global methods for several payment features.

InputParameter Class

Captures and retrieves additional parameters to be sent during the capture payment transaction.

Namespace

blng

InputParameter Methods

The following are methods for `InputParameter`.

setAdditionalParameters(additionalParams)

Creates a map of any additional parameters that the user sent with the payment capture request. This method and its parameters are
optional.

Signature

```
global static void setAdditionalParameters(Map<String,String> additionalParams)
```

Parameters

additionalParams
    Type: Map<String,String>

Return Value

Type: void

getAdditionalParameters()

Returns additional parameters sent during the capture payment request.

Signature

```
global static Map<string, string> getAdditionalParameters()
```

Return Value

Type: Map<String, String>

OutputResult Class

Stores and shows the results of the capture payment request made to the payment gateway.

Namespace

blng

> [OutputResult Methods]

OutputResult Methods

The following are methods for `OutputResult`.

> [setIsSuccess(isSuccess)]
> Sets the results of the capture request gateway communication as either `truetrue` (successful) or `falsefalse` (failed).
>
> [isSuccess(isSuccess)]
> Shows whether the capture payment request was successful in the payment gateway.
>
> [setErrorMessage(errorMessage)]
> Sets the error message following a failed payment capture request made to the payment gateway.
>
> [getErrorMessage()]
> Returns the error message that the payment gateway sent following a failed capture payment request.

setIsSuccess(isSuccess)

Sets the results of the capture request gateway communication as either `truetrue` (successful) or `falsefalse` (failed).

Signature

```
global static void setIsSuccess(boolean isSuccess)
```

Parameters

isSuccess
    Type: boolean

Return Value

Type: Void

`isSuccess(isSuccess)`

Shows whether the capture payment request was successful in the payment gateway.

Signature

```
global static Boolean isSuccess
```

Return Value

Type: Boolean

`setErrorMessage(errorMessage)`

Sets the error message following a failed payment capture request made to the payment gateway.

Signature

```
global static void setErrorMessage(String errorMessage)
```

Parameters

errorMessage
    Type: String

    Error message containing information about the failed gateway communication. For information about the error message, see the documentation for your gateway.

Return Value

Type: void

`getErrorMessage()`

Returns the error message that the payment gateway sent following a failed capture payment request.

Signature

```
global static String getErrorMessage()
```

Return Value

Type: String

PaymentGateway Interface

Interface for standard payment transaction operations such as voids and refunds.

Namespace

blng

Usage

A gateway implementor uses logic such as `paymentGatewayParameter.getTransactionType() == "generateToken"` and calls underlying payment transaction functions. Similar logic applies for voids, refunds, and other actions.

PaymentGateway Methods

PaymentGateway Example Implementation

PaymentGateway Methods

The following are methods for `PaymentGateway`.

generateToken(mapOfTransactionParameterById)

Generates a token for a payment method.

chargeTransaction(mapOfTransactionParameterById)

Authorize and capture a credit card payment and ACH payment.

voidTransaction(mapOfTransactionParameterById)

The Void transaction type can cancel either an original transaction or a transaction that hasn't been settled.

authorizeTransaction(mapOfTransactionParameterById)

Authorizes a credit card payment. To perform a charge transaction, the user must follow the authorization with a capture transaction.

refundTransaction(mapOfTransactionParameterById)

Refund a customer for a transaction that wsa successfully settled through the payment gateway.

generateToken(mapOfTransactionParameterById)

Generates a token for a payment method.

Signature

```
global Map<String, TransactionResult> generateToken(Map<String, TransactionParameter> mapOfTransactionParameterById)
```

Parameters

mapOfTransactionParameterById
    Type: Map<String, TransactionParameter>

    Map of parameter is based on the payment gateway.

Return Value

Type: Map<String, TransactionResult>

List of Transaction Results : List<TransactionResult>

chargeTransaction(mapOfTransactionParameterById)

Authorize and capture a credit card payment and ACH payment.

Signature

```
global Map<String, TransactionResult> chargeTransaction(Map<String, TransactionParameter>
mapOfTransactionParameterById)
```

Parameters

mapOfTransactionParameterById

Type: Map<String, TransactionParameter>

Map of parameter is based on the payment gateway.

Return Value

Type: Map<String, TransactionResult>

List of Transaction Results : List<TransactionResult>

voidTransaction(mapOfTransactionParameterById)

The Void transaction type can cancel either an original transaction or a transaction that hasn't been settled.

Signature

```
global Map<String, TransactionResult> voidTransaction(Map<String, TransactionParameter>
mapOfTransactionParameterById)
```

Parameters

mapOfTransactionParameterById

Type: Map<String, TransactionParameter>

Parameter map is based on the payment gateway.

Return Value

Type: Map<String, TransactionResult>

List of Transaction Results : List<TransactionResult>

authorizeTransaction(mapOfTransactionParameterById)

Authorizes a credit card payment. To perform a charge transaction, the user must follow the authorization with a capture transaction.

Signature

```
global Map<String, TransactionResult> authorizeTransaction(Map<String,
TransactionParameter> mapOfTransactionParameterById)
```

Parameters

mapOfTransactionParameterById

Type: Map<String, TransactionParameter>

Map of parameter is based on the payment gateway.

Return Value

Type: Map<String, TransactionResult>

List of Transaction Results : List<TransactionResult>

refundTransaction(mapOfTransactionParameterById)

Refund a customer for a transaction that wsa successfully settled through the payment gateway.

Signature

```
public Map<String, TransactionResult> refundTransaction(Map<String, TransactionParameter>
mapOfTransactionParameterById)
```

Parameters

mapOfTransactionParameterById
    Type: Map<String, TransactionParameter>

    Map of parameter is based on the payment gateway.

Return Value

Type: Map<String, TransactionResult>

List of Transaction Results : List<TransactionResult>

PaymentGateway Example Implementation

The YourGatewayAPI class implements the PaymentGateway, PaymentGateways, and PaymentGatewayStatus interfaces.

PaymentGateways Interface

Interface with common payment gateway operation across all gateways.

Namespace

blng

Usage

Test.

    PaymentGateways Methods

    PaymentGateways Example Implementation

PaymentGateways Methods

The following are methods for `PaymentGateways`.

processPayments(mapOfTransactionParameterById, paymentGatewayParameter)

Generates a token.

processPayments(mapOfTransactionParameterById, paymentGatewayParameter)

Generates a token.

Signature

```
global Map<String, TransactionResult> processPayments(Map<String, TransactionParameter>
mapOfTransactionParameterById, PaymentGatewayParameter paymentGatewayParameter)
```

Parameters

mapOfTransactionParameterById
    Type: Map<String, TransactionParameter>

paymentGatewayParameter
    Type: PaymentGatewayParameter

Return Value

Type: Map<String, TransactionResult>

PaymentGateways Example Implementation

The YourGatewayAPI class implements the PaymentGateway, PaymentGateways, and PaymentGatewayStatus interfaces.

PaymentGatewayParameter Class

Global parameter class for the payment gateway.

Namespace

blng

PaymentGatewayParameter Methods

PaymentGatewayParameter Methods

The following are methods for `PaymentGatewayParameter`.

setTransactionType(transactionType)
Sets TransactionType.

getTransactionType()
Gets TransactionType.

setTransactionType(transactionType)

Sets TransactionType.

Signature

```
global void setTransactionType(String transactionType)
```

Parameters

transactionType
    Type: String

Return Value

Type: Void

getTransactionType()

Gets TransactionType.

Signature

```
global String getTransactionType()
```

Return Value

Type: String

PaymentGatewayStatus Interface

Interface to populate the TransactionResult value object from the gateway response.

Namespace

blng

Usage

This method takes in the transaction result object and defines its attributes.

PaymentGatewayStatus Methods

PaymentGatewayStatus Example Implementation

PaymentGatewayStatus Methods

The following are methods for `PaymentGatewayStatus`.

PopulateGatewayStatus(transactionResult)
Populates the GatewayStatus enum on the TransactionResult, given return codes that are already set in the TransactionResults.

PopulateGatewayStatus(transactionResult)

Populates the GatewayStatus enum on the TransactionResult, given return codes that are already set in the TransactionResults.

Signature

```
global void PopulateGatewayStatus(TransactionResult transationResult)
```

Parameters

transationResult
    Type: TransactionResult

Return Value

Type: Void

PaymentGatewayStatus Example Implementation

The YourGatewayAPI class implements the PaymentGateway, PaymentGateways, and PaymentGatewayStatus interfaces.

TransactionAPI Class

The TransactionAPI class includes global methods for several payment features.

Namespace

blng

> TransactionAPI Methods

TransactionAPI Methods

The following are methods for `TransactionAPI`.

> generateToken(mapOfTransactionParameterById)
> Generates a token for a payment method.
>
> authorizeTransaction(mapOfTransactionParameterById)
> Authorizes a credit card payment. To perform a charge transaction, the user must follow the authorization with a capture transaction.
>
> captureTransaction(mapOfTransactionParameterById)
> Captures a payment for an authorize transaction.
>
> captureTransaction(captureInputParameters)
> Captures a payment for an authorization transaction using input parameters.
>
> chargeTransaction(mapOfTransactionParameterById)
> Authorize and capture a credit card payment or ACH payment.
>
> getPaymentStatus(mapOfTransactionParameterById)
> Gets the status of a payment transaction.
>
> voidTransaction(mapOfTransactionParameterById)
> Cancels an original transaction that hasn't been settled.
>
> refundTransaction(mapOfTransactionParameterById)
> Refunds a customer for a transaction that has been succesfully settled through the payment gateway.

nonReferredRefund(mapOfTransactionParameterById)

Performs a non-referred refund of a credit card payment or an ACH payment.

voidRefundTransaction(mapOfTransactionParameterById)

Voids an unsettled refund transaction from a credit card payment or ACH payment.

getRefundStatus(mapOfTransactionParameterById)

Gets the status of a refund transaction.

voidTokenTransaction(mapOfTransactionParameterById)

Voids a credit card token.

refundPayment(listofRefundParameters)

Refunds a payment.

resetInvoiceCorrectiveAction(invoices)

Unlocks an invoice.

validateRefundTransactionId(transactionId)

Validates whether a payment transaction can be refunded.

authorizeTransaction(invoiceID)

Authorizes a payment for an invoice.

chargeTransaction(invoiceID)

Authorizes a payment for an invoice.

generateToken(mapOfTransactionParameterById)

Generates a token for a payment method.

Signature

```
global static List<TransactionResult> generateToken(Map<String, TransactionParameter>
mapOfTransactionParameterById)
```

Parameters

mapOfTransactionParameterById
    Type: Map<String, TransactionParameter>

    Parameter map is based on the payment gateway.

Return Value

Type: List<TransactionResult>

List of Transaction Results : List<TransactionResult>

authorizeTransaction(mapOfTransactionParameterById)

Authorizes a credit card payment. To perform a charge transaction, the user must follow the authorization with a capture transaction.

Signature

```
public static List<TransactionResult> authorizeTransaction(Map<String,
TransactionParameter> mapOfTransactionParameterById)
```

Parameters

mapOfTransactionParameterById

    Type: Map<String, TransactionParameter>

    Parameter map is based on the payment gateway.

Return Value

Type: List<TransactionResult>

List of Transaction Results : List<TransactionResult>

`captureTransaction(mapOfTransactionParameterById)`

Captures a payment for an authorize transaction.

Signature

```
global static List<TransactionResult> captureTransaction(Map<String,
TransactionParameter> mapOfTransactionParameterById)
```

Parameters

mapOfTransactionParameterById

    Type: Map<String, TransactionParameter>

    Parameter map is based on the payment gateway.

Return Value

Type: List<TransactionResult>

List of Transaction Results : List<TransactionResult>

`captureTransaction(captureInputParameters)`

Captures a payment for an authorization transaction using input parameters.

Signature

```
global static List<CaptureOutputResult> captureTransaction(List<CaptureInputParameter>
captureInputParameters)
```

```
blng.TransactionAPI, captureTransaction, [List<CaptureInputParameter>],
List<CaptureOutputResult>
```

Parameters

captureInputParameters
  Type: `List<CaptureInputParameter>`

  A list of input parameters passed from the `CaptureInputParameter` class.

Return Value

Type: List<CaptureOutputResult on page 382>

chargeTransaction(mapOfTransactionParameterById)

Authorize and capture a credit card payment or ACH payment.

Signature

`public static List<TransactionResult> chargeTransaction(Map<String, TransactionParameter> mapOfTransactionParameterById)`

Parameters

mapOfTransactionParameterById
  Type: Map<String, TransactionParameter>

  Parameter map is based on the payment gateway.

Return Value

Type: List<TransactionResult>

List of Transaction Results : List<TransactionResult>

getPaymentStatus(mapOfTransactionParameterById)

Gets the status of a payment transaction.

Signature

`global static List<TransactionResult> getPaymentStatus(Map<String, TransactionParameter> mapOfTransactionParameterById)`

Parameters

mapOfTransactionParameterById
  Type: Map<String, TransactionParameter>

  Parameter map is based on the payment gateway.

Return Value

Type: List<TransactionResult>

List of Transaction Results : List<TransactionResult>

`voidTransaction(mapOfTransactionParameterById)`

Cancels an original transaction that hasn't been settled.

Signature

`global static` List<TransactionResult> voidTransaction(Map<`String`, TransactionParameter> mapOfTransactionParameterById)

Parameters

mapOfTransactionParameterById
    Type: Map<String, TransactionParameter>

    Parameter map is based on the payment gateway.

Return Value

Type: List<TransactionResult>

List of Transaction Results : List<TransactionResult>

`refundTransaction(mapOfTransactionParameterById)`

Refunds a customer for a transaction that has been succesfully settled through the payment gateway.

Signature

`global static` List<TransactionResult> refundTransaction(Map<`String`, TransactionParameter> mapOfTransactionParameterById)

Parameters

mapOfTransactionParameterById
    Type: Map<String, TransactionParameter>

    Parameter map is based on the payment gateway.

Return Value

Type: List<TransactionResult>

List of Transaction Results : List<TransactionResult>

`nonReferredRefund(mapOfTransactionParameterById)`

Performs a non-referred refund of a credit card payment or an ACH payment.

Signature

`global static` List<TransactionResult> nonReferredRefund(Map<`String`, TransactionParameter> mapOfTransactionParameterById)

Parameters

mapOfTransactionParameterById
    Type: Map<String, TransactionParameter>

    Parameter map is based on the payment gateway.

Return Value

Type: List<TransactionResult>

List of Transaction Results : List<TransactionResult>

`voidRefundTransaction(mapOfTransactionParameterById)`

Voids an unsettled refund transaction from a credit card payment or ACH payment.

Signature

```
global static List<TransactionResult> voidRefundTransaction(Map<String,
TransactionParameter> mapOfTransactionParameterById)
```

Parameters

mapOfTransactionParameterById
    Type: Map<String, TransactionParameter>

    Parameter map is based on the payment gateway.

Return Value

Type: List<TransactionResult>

List of Transaction Results : List<TransactionResult>

`getRefundStatus(mapOfTransactionParameterById)`

Gets the status of a refund transaction.

Signature

```
global static List<TransactionResult> getRefundStatus(Map<String, TransactionParameter>
mapOfTransactionParameterById)
```

Parameters

mapOfTransactionParameterById
    Type: Map<String, TransactionParameter>

    Parameter map is based on the payment gateway.

Return Value

Type: List<TransactionResult>

List of Transaction Results : List<TransactionResult>

voidTokenTransaction(mapOfTransactionParameterById)

Voids a credit card token.

Signature

```
global static List<TransactionResult> voidTokenTransaction(Map<String,
TransactionParameter> mapOfTransactionParameterById)
```

Parameters

mapOfTransactionParameterById
    Type: Map<String, TransactionParameter>

    Parameter map is based on the payment gateway.

Return Value

Type: List<TransactionResult>

List of Transaction Results : List<TransactionResult>

refundPayment(listofRefundParameters)

Refunds a payment.

Signature

```
global static List<TransactionResult> refundPayment(List<RefundParameter>
listofRefundParameters)
```

Parameters

listofRefundParameters
    Type: List<RefundParameter>

    Parameter map is based on the payment gateway.

Return Value

Type: List<TransactionResult>

List of Transaction Results : List<TransactionResult>

resetInvoiceCorrectiveAction(invoices)

Unlocks an invoice.

Signature

```
global static Boolean resetInvoiceCorrectiveAction(List<Invoice__c> invoices)
```

Parameters

invoices

Type: List<Invoice__c>

List of invoices to unlock due to payment issues.

Return Value

Type: Boolean

Returns True if invoice update is succesful, otherwise returns False.

validateRefundTransactionId(transactionId)

Validates whether a payment transaction can be refunded.

Signature

```
global static String validateRefundTransactionId(Id transactionId)
```

Parameters

transactionId

Type: Id

Single transactionId passed to validate whether a refund is allowed.

Return Value

Type: String

Returns String values such as the following.

- PAYMENT_TRANSACTION_NO_GATEWAY_ID_MESSAGE - Cannot refund payment transactions without gateway IDs. The invoice has been unlocked.
- INVALID_PAYMENT_TRANSACTION_FOR_REFUND_MESSAGE - Cannot refund payment transaction. This transaction is not currently eligible for refunding.
- ALLOW_REFUND_TRANSACTION - True
- REFUND_INVALID_PAYMENT_TRANSACTION_MESSAGE_ALOHA- Cannot refund invalid transactions or transactions without invoices.

authorizeTransaction(invoiceID)

Authorizes a payment for an invoice.

Signature

```
global static TransactionResult authorizeTransaction(Id invoiceID)
```

Parameters

invoiceID

Type: Id

Return Value

Type: TransactionResult

chargeTransaction(invoiceID)

Authorizes a payment for an invoice.

Signature

```
global static TransactionResult chargeTransaction(Id invoiceID)
```

Parameters

invoiceID
    Type: Id

Return Value

Type: TransactionResult

TransactionParameter Class

Acts as a container to store calculation-specific information.

Namespace

blng

TransactionParameter Methods

TransactionParameter Methods

The following are methods for `TransactionParameter`.

setRequestingInvoiceId(requestingInvoiceId)
Sets RequestingInvoiceId.

getRequestingInvoiceId()
Gets Requesting InvoiceId.

setCardCodeResponse(cardCodeResponse)
Sets cardCodeResponse.

getCardCodeResponse()
Gets cardCodeResponse.

setCardNumber(requestingCreditCardNumber)
Sets requestingCreditCardNumber.

getCardNumber()
Gets requestingCreditCardNumber.

setPayment(paymentInstance)

Sets payment details like merchantRefId, RequestCreditCardNumber__c, CardCodeResponse__c, GatewayID__c. Also sets payment method details for PaymentType__c, CardExpirationYear__c, CardExpirationMonth__c, BankName__c, BankAccountType__c, BankAccountNumber__c, BankAccountName__c, CardType__c, BankRoutingCode__c if they aren't already set.

getPayment()

Gets Payment.

getMerchantId()

Gets merchantRefId.

setMerchantId()

Sets MerchantId.

getGatewayId()

Gets Gateway ID.

getResponseValueByKey()

Gets ResponseValueByKey

setGateway(gateway)

Sets Gateway ID.

setPaymentMethod(paymentMethod)

Sets PaymentMethod.

getPaymentMethod()

Gets PaymentMethod.

setInvoice(invoice)

Sets invoice.

getInvoice()

Gets Invoice.

setInputParameter(inputParameter)

Sets the input parameters for a payment capture transaction.

getInputParameter()

Returns the input parameters sent from the `inputParameter` class.

setInvoiceLine(line)

Sets listOfInvoiceLine.

getInvoiceLine()

Gets listOfInvoiceLine.

setTransaction(transactionInstance)

Sets transaction instance and details like merchantRefId, RequestCreditCardNumber, ResponseCode, and GatewayID.

getTransaction()

Gets Transaction.

setAccount(accountInstance)

Sets Account.

getAccount()

Gets Account.

setRequestBody(requestBody)

Sets RequestBody.

getRequestBody()

Gets RequestBody.

setFirstName(firstName)

Sets FirstName.

getFirstName()

Gets FirstName.

setLastName(lastName)

Sets LastName.

getLastName()

Gets LastName.

setEmailId(emailId)

Sets EmailId.

getEmailId()

Gets EmailId.

setStreet(street)

Sets street.

getStreet()

Gets Street.

setCity(city)

Sets City.

getCity()

Gets City.

setState(state)

Sets state.

getState()

Gets State.

setZipCode(zipCode)

Sets zip code.

getZipCode()

Gets ZipCode.

setCountry(country)

Sets Country.

getCountry()

Gets Country.

setPhone(phone)

Sets phone.

getPhone()

Gets Phone.

setCurrencyId(currencyId)

Sets Currency Id.

getCurrencyId()

Gets Currency Id.

setAmount(amount)

Sets amount.

getAmount()

Gets Amount.

setRequestingInvoiceId(requestingInvoiceId)

Sets RequestingInvoiceId.

Signature

```
global void setRequestingInvoiceId(Id requestingInvoiceId)
```

Parameters

requestingInvoiceId
    Type: Id

Return Value

Type: void

getRequestingInvoiceId()

Gets Requesting InvoiceId.

Signature

```
global Id getRequestingInvoiceId()
```

Return Value

Type: Id

setCardCodeResponse(cardCodeResponse)

Sets cardCodeResponse.

Signature

```
global void setCardCodeResponse(String cardCodeResponse)
```

Parameters

cardCodeResponse
    Type: String

Return Value

Type: void

getCardCodeResponse()

Gets cardCodeResponse.

Signature

global String getCardCodeResponse()

Return Value

Type: String

setCardNumber(requestingCreditCardNumber)

Sets requestingCreditCardNumber.

Signature

global void setCardNumber(String requestingCreditCardNumber)

Parameters

requestingCreditCardNumber
    Type: String

Return Value

Type: void

getCardNumber()

Gets requestingCreditCardNumber.

Signature

global String getCardNumber()

Return Value

Type: String

setPayment(paymentInstance)

Sets payment details like merchantRefId, RequestCreditCardNumber__c, CardCodeResponse__c, GatewayID__c. Also sets payment method details for PaymentType__c, CardExpirationYear__c, CardExpirationMonth__c, BankName__c, BankAccountType__c, BankAccountNumber__c, BankAccountName__c, CardType__c, BankRoutingCode__c if they aren't already set.

Signature

```
global void setPayment(Payment__c paymentInstance)
```

Parameters

paymentInstance
Type: Payment__c

Return Value

Type: void

getPayment()

Gets Payment.

Signature

```
global Payment__c getPayment()
```

Return Value

Type: Payment__c

getMerchantId()

Gets merchantRefId.

Signature

```
global String getMerchantId()
```

Return Value

Type: String

setMerchantId()

Sets MerchantId.

Signature

```
global String setMerchantId()
```

Return Value

Type: String

getGatewayId()

Gets Gateway ID.

Signature

```
global String getGatewayId()
```

Return Value

Type: String

getResponseValueByKey()

Gets ResponseValueByKey

Signature

```
global Map<String, String> getResponseValueByKey()
```

Return Value

Type: Map<String, String>

setGateway(gateway)

Sets Gateway ID.

Signature

```
global void setGateway(PaymentGateway__c gateway)
```

Parameters

gateway
    Type: PaymentGateway__c

Return Value

Type: void

setPaymentMethod(paymentMethod)

Sets PaymentMethod.

Signature

```
global void setPaymentMethod(PaymentMethod__c paymentMethod)
```

Parameters

paymentMethod
　　Type: PaymentMethod__c

Return Value

Type: void

getPaymentMethod()

Gets PaymentMethod.

Signature

global PaymentMethod__c getPaymentMethod()

Return Value

Type: PaymentMethod__c

setInvoice(invoice)

Sets invoice.

Signature

global void setInvoice(Invoice__c invoice)

Parameters

invoice
　　Type: Invoice__c

Return Value

Type: void

getInvoice()

Gets Invoice.

Signature

global Invoice__c getInvoice()

Return Value

Type: Invoice__c

setInputParameter(inputParameter)

Sets the input parameters for a payment capture transaction.

Signature

```
global void setInputParameter(InputParameter inputParameter)
```

Parameters

inputParameter
    Type: InputParameter

Return Value

Type: void

getInputParameter()

Returns the input parameters sent from the `inputParameter` class.

Signature

```
global InputParameter getInputParameter()
```

Return Value

Type: InputParameter

setInvoiceLine(line)

Sets listOfInvoiceLine.

Signature

```
global void setInvoiceLine(InvoiceLine__c line)
```

Parameters

line
    Type: InvoiceLine__c

Return Value

Type: void

getInvoiceLine()

Gets listOfInvoiceLine.

Signature

```
global List<InvoiceLine__c> getInvoiceLine()
```

Return Value

Type: List<InvoiceLine__c>

setTransaction(transactionInstance)

Sets transaction instance and details like merchantRefId, RequestCreditCardNumber, ResponseCode, and GatewayID.

Signature

```
global void setTransaction(PaymentTransaction__c transactionInstance)
```

Parameters

transactionInstance
   Type: PaymentTransaction__c

Return Value

Type: void

getTransaction()

Gets Transaction.

Signature

```
global PaymentTransaction__c getTransaction()
```

Return Value

Type: PaymentTransaction__c

setAccount(accountInstance)

Sets Account.

Signature

```
global void setAccount(Account accountInstance)
```

Parameters

accountInstance
   Type: Account

Return Value

Type: void

getAccount()

Gets Account.

Signature

global Account getAccount()

Return Value

Type: Account

setRequestBody(requestBody)

Sets RequestBody.

Signature

global void setRequestBody(String requestBody)

Parameters

requestBody
    Type: String

Return Value

Type: void

getRequestBody()

Gets RequestBody.

Signature

global String getRequestBody()

Return Value

Type: String

setFirstName(firstName)

Sets FirstName.

Signature

global void setFirstName(String firstName)

Parameters

firstName
    Type: String

Return Value

Type: void

getFirstName()

Gets FirstName.

Signature

global String getFirstName()

Return Value

Type: String

setLastName(lastName)

Sets LastName.

Signature

global void setLastName(String lastName)

Parameters

lastName
    Type: String

Return Value

Type: void

getLastName()

Gets LastName.

Signature

global String getLastName()

Return Value

Type: String

setEmailId(emailId)

Sets EmailId.

Signature

```
global void setEmailId(String emailId)
```

Parameters

emailId
    Type: String

Return Value

Type: void

getEmailId()

Gets EmailId.

Signature

```
global String getEmailId()
```

Return Value

Type: String

setStreet(street)

Sets street.

Signature

```
global void setStreet(String street)
```

Parameters

street
    Type: String

Return Value

Type: void

getStreet()

Gets Street.

Signature

```
global String getStreet()
```

Return Value

Type: String

setCity(city)

Sets City.

Signature

```
global void setCity(String city)
```

Parameters

city
    Type: String

Return Value

Type: void

getCity()

Gets City.

Signature

```
global String getCity()
```

Return Value

Type: String

setState(state)

Sets state.

Signature

```
global void setState(String state)
```

Parameters

state
    Type: String

Return Value

Type: void

getState()

Gets State.

Signature

`global String getState()`

Return Value

Type: String

setZipCode(zipCode)

Sets zip code.

Signature

`global void setZipCode(String zipCode)`

Parameters

zipCode
    Type: String

Return Value

Type: void

getZipCode()

Gets ZipCode.

Signature

`global String getZipCode()`

Return Value

Type: String

setCountry(country)

Sets Country.

Signature

`global void setCountry(String country)`

Parameters

country
    Type: String

Return Value

Type: void

getCountry()

Gets Country.

Signature

```
global String getCountry()
```

Return Value

Type: String

setPhone(phone)

Sets phone.

Signature

```
global void setPhone(String phone)
```

Parameters

phone
    Type: String

Return Value

Type: void

getPhone()

Gets Phone.

Signature

```
global String getPhone()
```

Return Value

Type: String

`setCurrencyId(currencyId)`

Sets Currency Id.

Signature

```
global void setCurrencyId(String currencyId)
```

Parameters

currencyId
    Type: String

Return Value

Type: void

`getCurrencyId()`

Gets Currency Id.

Signature

```
global String getCurrencyId()
```

Return Value

Type: String

`setAmount(amount)`

Sets amount.

Signature

```
global void setAmount(String amount)
```

Parameters

amount
    Type: String

Return Value

Type: void

`getAmount()`

Gets Amount.

Signature

```
global String getAmount()
```

Return Value

Type: String

TransactionResult Class

Holds the result of the transaction.

Namespace

blng

TransactionResult Methods

TransactionResult Methods

The following are methods for `TransactionResult`.

setResponseCodeMessage(responseCodeMessage)
Sets ResponseCodeMessage.

getResponseCodeMessage()
Gets ResponseCodeMessage.

setCardCodeResponse(cardCodeResponse)
Sets CardCodeResponse.

getCardCodeResponse()
Gets CardCodeResponse.

setCustomerProfileToken(customerProfileToken)
Sets customerProfileToken.

getCustomerProfileToken()
Gets CustomerProfileToken.

setPaymentToken(paymentToken)
Sets PaymentToken.

getPaymentToken()
Gets PaymentToken.

setResponseStatus(responseStatus)
Sets ResponseStatus.

getResponseStatus()
Gets ResponseStatus.

setGatewayRequest(gatewayRequest)
Sets the capture payment request sent to the payment gateway.

getGatewayRequest(gatewayRequest)

Returns the string containing the payment capture request sent to the gateway.

setGatewayResponse(gatewayResponse)

Returns the gateway's response following a capture payment transaction.

getGatewayResponse(gatewayResponse)

Returns the string containing the gateway's response from a capture payment transaction.

setResponseMessage(responseMessage)

Sets ResponseMessage.

getResponseMessage()

Gets ResponseMessage.

setResponseCode(responseCode)

Sets ResponseCode.

getResponseCode()

Gets ResponseCodeMessage.

setResponseToValidate(responseToValidate)

Sets ResponseToValidate.

getResponseToValidate()

Gets Response To Validate.

setGatewayId(gatewayId)

Sets Gateway ID.

getGatewayId()

Gets Gateway ID.

setType(type)

Sets Type.

getType()

Gets Type.

setRequestTypeOfPayment(requestTypeOfPayment)

Sets RequestTypeOfPayment.

getRequestTypeofPayment()

Gets RequestTypeOfPayment.

setRequestTransactionType(requestTransactionType)

Sets RequestTransactionType.

getRequestTransactionType()

Gets RequestTransactionType.

setIsSuccess(isSuccess)

Sets IsSuccess

getIsSuccess()

Gets IsSuccess.

setError(error)

Sets Error.

getErrors()

Gets Errors.

setResponseValueByKey(getResponseValueByKey)

Sets ResponseValueByKey

getResponseValueByKey()

Gets ResponseValueByKey.

setId(Id)

Sets ID.

getId()

Gets ID.

setMessage(message)

Sets Message.

getMessage()

Gets Message.

setEntity(entity)

Sets Message.

getEntity()

Gets Message.

setGatewayStatus(gatewayStatus)

Sets gatewayStatus.

getGatewayStatus()

Gets GatewayStatus.

setResponseCodeMessage(responseCodeMessage)

Sets ResponseCodeMessage.

Signature

```
global void setResponseCodeMessage(String responseCodeMessage)
```

Parameters

responseCodeMessage
Type: String

Return Value

Type: void

getResponseCodeMessage()

Gets ResponseCodeMessage.

Signature

```
global String getResponseCodeMessage()
```

Return Value

Type: String

setCardCodeResponse(cardCodeResponse)

Sets CardCodeResponse.

Signature

```
global void setCardCodeResponse(String cardCodeResponse)
```

Parameters

cardCodeResponse
    Type: String

Return Value

Type: void

getCardCodeResponse()

Gets CardCodeResponse.

Signature

```
global String getCardCodeResponse()
```

Return Value

Type: String

setCustomerProfileToken(customerProfileToken)

Sets customerProfileToken.

Signature

```
global void setCustomerProfileToken(String customerProfileToken)
```

Parameters

customerProfileToken
    Type: String

Return Value

Type: void

getCustomerProfileToken()

Gets CustomerProfileToken.

Signature

```
global String getCustomerProfileToken()
```

Return Value

Type: String

setPaymentToken(paymentToken)

Sets PaymentToken.

Signature

```
public void setPaymentToken(String paymentToken)
```

Parameters

paymentToken
    Type: String

Return Value

Type: void

getPaymentToken()

Gets PaymentToken.

Signature

```
global String getPaymentToken()
```

Return Value

Type: String

setResponseStatus(responseStatus)

Sets ResponseStatus.

Signature

```
global void setResponseStatus(String responseStatus)
```

Parameters

responseStatus
    Type: String

Return Value

Type: void

getResponseStatus()

Gets ResponseStatus.

Signature

`global String getResponseStatus()`

Return Value

Type: String

setGatewayRequest(gatewayRequest)

Sets the capture payment request sent to the payment gateway.

Signature

`global void setGatewayRequest(String gatewayRequest)`

Parameters

gatewayRequest
    Type: String

Return Value

Type: Void

getGatewayRequest(gatewayRequest)

Returns the string containing the payment capture request sent to the gateway.

Signature

`global String GatewayRequest()`

Return Value

Type: String

setGatewayResponse(gatewayResponse)

Returns the gateway's response following a capture payment transaction.

Signature

```
global void setGatewayResponse(String gatewayResponse)
```

Parameters

gatewayResponse
    Type: String

Return Value

Type: void

getGatewayResponse(gatewayResponse)

Returns the string containing the gateway's response from a capture payment transaction.

Signature

```
global void getGatewayResponse()
```

Return Value

Type: Void

setResponseMessage(responseMessage)

Sets ResponseMessage.

Signature

```
global void setResponseMessage(String responseMessage)
```

Parameters

responseMessage
    Type: String

Return Value

Type: void

getResponseMessage()

Gets ResponseMessage.

Signature

```
global String getResponseMessage()
```

Return Value

Type: String

setResponseCode(responseCode)

Sets ResponseCode.

Signature

```
global void setResponseCode(String responseCode)
```

Parameters

responseCode
    Type: String

Return Value

Type: void

getResponseCode()

Gets ResponseCodeMessage.

Signature

```
global String getResponseCode()
```

Return Value

Type: String

setResponseToValidate(responseToValidate)

Sets ResponseToValidate.

Signature

```
global void setResponseToValidate(String responseToValidate)
```

Parameters

responseToValidate
    Type: String

Return Value

Type: void

getResponseToValidate()

Gets Response To Validate.

Signature

`public String getResponseToValidate()`

Return Value

Type: String

setGatewayId(gatewayId)

Sets Gateway ID.

Signature

`global void setGatewayId(String gatewayId)`

Parameters

gatewayId
    Type: String

Return Value

Type: void

getGatewayId()

Gets Gateway ID.

Signature

`global String getGatewayId()`

Return Value

Type: String

setType(type)

Sets Type.

Signature

`global void setType(String type)`

Parameters

type
    Type: String

Return Value

Type: void

getType()

Gets Type.

Signature

`global String getType()`

Return Value

Type: String

setRequestTypeOfPayment(requestTypeOfPayment)

Sets RequestTypeOfPayment.

Signature

`global void setRequestTypeOfPayment(String requestTypeOfPayment)`

Parameters

requestTypeOfPayment
    Type: String

Return Value

Type: void

getRequestTypeofPayment()

Gets RequestTypeOfPayment.

Signature

`global String getRequestTypeofPayment()`

Return Value

Type: String

setRequestTransactionType(requestTransactionType)

Sets RequestTransactionType.

Signature

```
global void setRequestTransactionType(String requestTransactionType)
```

Parameters

requestTransactionType
   Type: String

Return Value

Type: void

getRequestTransactionType()

Gets RequestTransactionType.

Signature

```
global String getRequestTransactionType()
```

Return Value

Type: String

setIsSuccess(isSuccess)

Sets IsSuccess

Signature

```
global void setIsSuccess(Boolean isSuccess)
```

Parameters

isSuccess
   Type: Boolean

Return Value

Type: void

getIsSuccess()

Gets IsSuccess.

Signature

```
global Boolean getIsSuccess()
```

Return Value

Type: Boolean

setError(error)

Sets Error.

Signature

```
global void setError(String error)
```

Parameters

error
   Type: String

Return Value

Type: void

getErrors()

Gets Errors.

Signature

```
public List<String> getErrors()
```

Return Value

Type: List<String>

setResponseValueByKey(getResponseValueByKey)

Sets ResponseValueByKey

Signature

```
public void setResponseValueByKey(Map<String,String> getResponseValueByKey)
```

Parameters

getResponseValueByKey
   Type: Map<String,String>

Return Value

Type: void

getResponseValueByKey()

Gets ResponseValueByKey.

Signature

```
global Map<string, string> getResponseValueByKey()
```

Return Value

Type: Map<string, string>

setId(Id)

Sets ID.

Signature

```
global void setId(String Id)
```

Parameters

Id
    Type: String

Return Value

Type: void

getId()

Gets ID.

Signature

```
global String getId()
```

Return Value

Type: String

setMessage(message)

Sets Message.

Signature

```
global void setMessage(String message)
```

Parameters

message
    Type: String

Return Value

Type: void

getMessage()

Gets Message.

Signature

```
global String getMessage()
```

Return Value

Type: String

setEntity(entity)

Sets Message.

Signature

```
global void setEntity(String entity)
```

Parameters

entity
    Type: String

Return Value

Type: void

getEntity()

Gets Message.

Signature

```
global String getEntity()
```

Return Value

Type: String

setGatewayStatus(gatewayStatus)

Sets gatewayStatus.

Signature

```
global void setGatewayStatus(GatewayStatusType gatewayStatus)
```

Parameters

gatewayStatus
    Type: GatewayStatusType

Return Value

Type: void

getGatewayStatus()

Gets GatewayStatus.

Signature

```
global GatewayStatusType getGatewayStatus()
```

Return Value

Type: GatewayStatusType

TransactionResult.GatewayStatusType Enum

An enum defining the appropriate gateway status values returned by the payment gateway.

Enum Values

The following are the values of the `blng.TransactionResult.GatewayStatusType` enum.

Value	Description
Success	The gateway has processed the transaction succesfully.
Decline	Declined responses occur when the gateway's call to the customer bank fails. This usually happens for customer issues such as insufficient funds, a frozen credit card, or a brief disconnection during the call.
ValidationError	Gateways respond with validation errors when Salesforce Billing sends incorrect customer payment information, such as misspelled credit card data or a CVV with missing numbers. Payment runs won't create payments from transactions with a gateway status of Validation Error. Users should change the incorrect payment information on their payment method so that future payment runs can use it for payments.

Value	Description
PermanentFail	Gateways respond with a permanent failure code when the customer's bank recognizes their payment account as closed, terminated, or fraudulent. In this case, the gateway won't accept any further calls from the payment method associated with this transaction. Following a permanent fail response, a transaction changes its gateway status to Permanent Fail.
RequiresReview	Gateways respond with a review code when the gateway call initially fails, but the payment method may still work following extra processing. For example, some banks send out this type of response when they have further questions about the payment request, and will provide an authorization code manually when the payment manager calls the processor.
Indeterminate	Gateways send indeterminate responses when they require the user to check the status of the transaction request. This often occurs following server timeouts, system failure, or any action that unexpectedly interrupts the gateway's ability to process the payment.
SystemError	Salesforce Billing provides a system fail response when it has to end its gateway payment request before receiving a response. This can happen due to gateway server errors, invalid customer credentials, or anytime Salesforce Billing's request times out before it receives a gateway response. System failures occur before Salesforce Billing's request reaches the gateway, so there's no risk of an unaccounted payment remaining in the gateway. If your transaction has a System Fail gateway status, you can either manually create the payment or wait for another payment run to pick up your invoice.

Custom Gateway Classes

Salesforce Billing provides three classes that users can customize to integrate with their chosen payment gateway.

YourGatewayAPI

YourGatewayAPI is a base class that connects the Salesforce Billing package to your payment gateway package. All user gateway API classes will be different based on your needs. However, we've provided a template that you can use to help create your own.

YourGatewayName

YourGateWayName is a base class. All user gateway API classes will be different based on your needs. However, we've provided a template that you can use to help create your own.

YourGatewayNameUtils

YourGatewayNameUtils is a singleton utility class that interacts with an external payment gateway. All user gateway API classes will be different based on your needs. However, we've provided a template that you can use to help create your own.

YourHttpService

YourHttpService is an example of common HTTP utilities that might be required while building a partner package. All user gateway API classes will be different based on your needs. However, we've provided a template that you can use to help create your own.

YourGatewayAPI

YourGatewayAPI is a base class that connects the Salesforce Billing package to your payment gateway package. All user gateway API classes will be different based on your needs. However, we've provided a template that you can use to help create your own.

`YourGatewayAPI` references the following public classes.

- YourGatewayName
- YourGatewayNameUtils

```
global class YourGatewayAPI implements blng.PaymentGateWay, blng.PaymentGateWays,
blng.PaymentGateWayStatus
{
    // ============================================================================
    // CONSTANT
    // ============================================================================

    // ============================================================================
    // STATIC VARIABLES
    // ============================================================================

    // Attribute to implement singleton pattern for YourGatewayName  class
    private static YourGatewayName YourGatewayNameInstance;

     // ============================================================================
    // VARIABLES
    // ============================================================================

    // Attribute to implement singleton pattern for AuthorizeDotNet class
    private static final String GENERATE_TOKEN = 'generateToken';
    private static final String AUTHORIZE_TRANSACTION = 'authorizeTransaction';
    private static final String CHARGE_TRANSACTION = 'chargeTransaction';
    private static final String VOID_TRANSACTION = 'voidTransaction';
    private static final String REFUND_TRANSACTION = 'refundTransaction';
    private static final String CAPTURE_TRANSACTION = 'captureTransaction';
    private static final String VOID_REFUND_TRANSACTION = 'voidRefundTransaction';
    private static final String VOID_TOKEN_TRANSACTION = 'voidTokenTransaction';
    private static final String GET_PAYMENT_STATUS = 'getPaymentStatus';
    private static final String GET_REFUND_STATUS = 'getRefundStatus';
    private static final String NON_REFERRED_REFUND = 'nonReferredRefund';
    private static final String PROCESS_PAYMENTS = 'processPayments';

    // ============================================================================
    // Methods
    // ============================================================================
    /**
    * @name service
    * @description service method to create an only instance of TransactionService class
    *              if serviceInstance is NULL then only create a new instance otherwise
    *              return the existing one
    * @param   NA
    * @return TransactionService instance
    * @exception
    * @author
```

```
     * @created
     * @remark
     * @change
     */
    @TestVisible
    private static YourGatewayName service()
    {
        if (NULL == YourGatewayNameInstance)
        {
            YourGatewayNameInstance = YourGatewayName.getInstance();
        }
        return YourGatewayNameInstance;
    }


    /**
     * @name processPayments
     * @description Method to generate Token
     * @param
     * @return
     * @exception
     * @author
     * @created 2016-11-03
     * @remark
     * @change
     */
    global static Map<String, blng.TransactionResult> processPayments(Map<String,
blng.TransactionParameter> mapOfTransactionParameterById, blng.PaymentGatewayParameter
paymentGatewayParameter)
    {
        if(paymentGatewayParameter.getTransactionType() == GENERATE_TOKEN)
        {
            return generateToken(mapOfTransactionParameterById);
        }
        else if(paymentGatewayParameter.getTransactionType() == AUTHORIZE_TRANSACTION)
        {
            return authorizeTransaction(mapOfTransactionParameterById);
        }
        else if(paymentGatewayParameter.getTransactionType() == CHARGE_TRANSACTION)
        {
            return chargeTransaction(mapOfTransactionParameterById);
        }
        else if(paymentGatewayParameter.getTransactionType() == VOID_TRANSACTION)
        {
            return voidTransaction(mapOfTransactionParameterById);
        }
        else if(paymentGatewayParameter.getTransactionType() == REFUND_TRANSACTION)
        {
            return refundTransaction(mapOfTransactionParameterById);
        }
        else if(paymentGatewayParameter.getTransactionType() == CAPTURE_TRANSACTION)
        {
            return captureTransaction(mapOfTransactionParameterById);
        }
        else if(paymentGatewayParameter.getTransactionType() == VOID_REFUND_TRANSACTION)
```

```
                {
                    return voidRefundTransaction(mapOfTransactionParameterById);
                }
                else if(paymentGatewayParameter.getTransactionType() == VOID_TOKEN_TRANSACTION)
                {
                    return voidTokenTransaction(mapOfTransactionParameterById);
                }
                else if(paymentGatewayParameter.getTransactionType() == GET_PAYMENT_STATUS)
                {
                    return getPaymentStatus(mapOfTransactionParameterById);
                }
                else if(paymentGatewayParameter.getTransactionType() == GET_REFUND_STATUS)
                {
                    return getRefundStatus(mapOfTransactionParameterById);
                }
                else if(paymentGatewayParameter.getTransactionType() == NON_REFERRED_REFUND)
                {
                    return nonReferredRefund(mapOfTransactionParameterById);
                }

                return NULL;
        }

        /**
        * @name generateToken
        * @description Method to generate Token for a Payment Method
        * @param Map[Key => String [unique Id],Value => TransactionParameter]
        * @return Map[Key => String [unique Id],Value => TransactionResult]
        * @exception
        * @author
        * @created
        * @remark
        * @change
        */
        global static Map<String, blng.TransactionResult> generateToken(Map<String,
    blng.TransactionParameter> mapOfTransactionParameterById)
        {
            return service().generateToken(mapOfTransactionParameterById);
        }

        /**
        * @name voidTokenTransaction
        * @description Method to void Token for a Payment method Token
        * @param Map[Key => String [unique Id],Value => TransactionParameter]
        * @return Map[Key => String [unique Id],Value => TransactionResult]
        * @exception
        * @author
        * @created
        * @remark
        * @change
        */
        global static Map<String, blng.TransactionResult> voidTokenTransaction(Map<String,
    blng.TransactionParameter> mapOfTransactionParameterById)
        {
```

```
        return service().voidTokenTransaction(mapOfTransactionParameterById);
    }


    /**
    * @name authorizeTransaction
    * @description Method to Authorize a payment for a Invoice
    * @param Map[Key => String [unique Id],Value => TransactionParameter]
    * @return Map[Key => String [unique Id],Value => TransactionResult]
    * @exception
    * @author
    * @created
    * @remark
    * @change
    */
    global static Map<String, blng.TransactionResult> authorizeTransaction(Map<String,
blng.TransactionParameter> mapOfTransactionParameterById)
    {
        return service().authorizeTransaction(mapOfTransactionParameterById);
    }

    /**
    * @name chargeTransaction
    * @description Method to Charge a payment for a Invoice
    * @param Map[Key => String [unique Id],Value => TransactionParameter]
    * @return Map[Key => String [unique Id],Value => TransactionResult]
    * @exception
    * @author
    * @created
    * @remark
    * @change
    */
    global static Map<String, blng.TransactionResult> chargeTransaction(Map<String,
blng.TransactionParameter> mapOfTransactionParameterById)
    {
        return service().chargeTransaction(mapOfTransactionParameterById);
    }

    /**
    * @name captureTransaction
    * @description Method to capture a payment for a Authorize Transaction
    * @param Map[Key => String [unique Id],Value => TransactionParameter]
    * @return Map[Key => String [unique Id],Value => TransactionResult]
    * @exception
    * @author
    * @created
    * @remark
    * @change
    */
    global static Map<String, blng.TransactionResult> captureTransaction(Map<String,
blng.TransactionParameter> mapOfTransactionParameterById)
    {
        return service().captureTransaction(mapOfTransactionParameterById);
    }
```

```
    /**
    * @name voidTransaction
    * @description Method to Void a payment for a Invoice
    * @param Map[Key => String [unique Id],Value => TransactionParameter]
    * @return Map[Key => String [unique Id],Value => TransactionResult]
    * @exception
    * @author
    * @created
    * @remark
    * @change
    */
    global static Map<String, blng.TransactionResult> voidTransaction(Map<String,
blng.TransactionParameter> mapOfTransactionParameterById)
    {
        return service().voidTransaction(mapOfTransactionParameterById);
    }


    /**
    * @name refundTransaction
    * @description Method to Refund a payment for a payment Transaction
    * @param Map[Key => String [unique Id],Value => TransactionParameter]
    * @return Map[Key => String [unique Id],Value => TransactionResult]
    * @exception
    * @author
    * @created
    * @remark
    * @change
    */
    global static Map<String, blng.TransactionResult> refundTransaction(Map<String,
blng.TransactionParameter> mapOfTransactionParameterById)
    {
        return service().refundTransaction(mapOfTransactionParameterById);
    }


    /**
    * @name nonReferencedRefund
    * @description Method to non Referenced Refund a payment for a Invoice
    * @param Map[Key => String [unique Id],Value => TransactionParameter]
    * @return Map[Key => String [unique Id],Value => TransactionParameter]
    * @exception
    * @author
    * @created
    * @remark
    * @change
    */
    global static Map<String, blng.TransactionResult> nonReferredRefund(Map<String,
blng.TransactionParameter> mapOfTransactionParameterById)
    {
        return service().nonReferencedRefund(mapOfTransactionParameterById);
    }


    /**
```

```
    * @name voidRefundTransaction
    * @description Method to Void refund for a Refund Transaction
    * @param Map[Key => String [unique Id],Value => TransactionParameter]
    * @return Map[Key => String [unique Id],Value => TransactionResult]
    * @exception
    * @author
    * @created
    * @remark
    * @change
    */
    global static Map<String, blng.TransactionResult> voidRefundTransaction(Map<String,
blng.TransactionParameter> mapOfTransactionParameterById)
    {
        return service().voidRefundTransaction(mapOfTransactionParameterById);
    }

    /**
    * @name getPaymentStatus
    * @description Method to get payment status for a Payment Transaction
    * @param Map[Key => String [unique Id],Value => TransactionParameter]
    * @return Map[Key => String [unique Id],Value => TransactionResult]
    * @exception
    * @author
    * @created
    * @remark
    * @change
    */
    global static Map<String, blng.TransactionResult> getPaymentStatus(Map<String,
blng.TransactionParameter> mapOfTransactionParameterById)
    {
        return service().getPaymentStatus(mapOfTransactionParameterById);
    }

    /**
    * @name getRefundStatus
    * @description Method to get refund status for a Refund Transaction
    * @param Map[Key => String [unique Id],Value => TransactionParameter]
    * @return Map[Key => String [unique Id],Value => TransactionResult]
    * @exception
    * @author
    * @created
    * @remark
    * @change
    */
    global static Map<String, blng.TransactionResult> getRefundStatus(Map<String,
blng.TransactionParameter> mapOfTransactionParameterById)
    {
        return service().getPaymentStatus(mapOfTransactionParameterById);
    }
    /**
     * @name populateGatewayStatus
     * @description Method that populates the GatewayStatus enum on the TransactionResult,

     * given return codes that are already set in the TransactionResults
```

```
     * @param transactionResult
     */
    global static void populateGatewayStatus(blng.TransactionResult transactionResult)
    {
        service().populateGatewayStatus(transactionResult);
    }
}
```

YourGatewayName

YourGateWayName is a base class. All user gateway API classes will be different based on your needs. However, we've provided a template that you can use to help create your own.

This class shows an example of a void token calling the CyberSource payment gateway. The gateway receives a request and then provides a response through the TransactionResult object. Salesforce Billing evaluates the response and filters it into one of several fields shown in the payment transaction record's Gateway Status field. For more information on payment transactions and managing gateway responses in Salesforce Billing, check out Managing Payment Transactions.

```
Note : Map<String, blng.TransactionParameter> is the parameter in all methods of this class

      blng.TransactionParameter class is a global class which list of global method exposed
 detailed in next section
        and String in the map is a Unique Id which could be Invoice or Account Id

       Map<String, blng.TransactionResult> is the return for all the Methods in this class

      blng.TransactionResult class is a global class with a list of global method detailed
 in next section
        the string should be the same string of parameter

       public class YourGatewayName
{
    //Add variables here
    //Gateway Status related variables
    private static Map<string, blng.TransactionResult.GatewayStatusType>
mapGatewayStatusEnumTypesByStrings =
            new Map<string, blng.TransactionResult.GatewayStatusType>();
    static {
        List<blng.TransactionResult.GatewayStatusType> enumValues =
blng.TransactionResult.GatewayStatusType.values();
        for (Integer i = 0; i < enumValues.size(); i++) {
            mapGatewayStatusEnumTypesByStrings.put(enumValues.get(i).name(),
enumValues.get(i));
        }
    }


 //
    // ========================================================================
    // CONSTANT
    // ========================================================================
```

```
        private static final String ACCEPT = 'ACCEPT';
        private static final String REJECT = 'REJECT';
        private static final String SUCCESS = 'SUCCESS';
        private static final String FAILURE = 'FAILURE';
        private static final String ERROR = 'ERROR';
        private static final String DECISION = 'decision';

    /**
    * @name handleError
    * @description populates TransactionResult with error values
    * @Param
    * @return NA
    * @exception NA
    */
    private void handleError(blng.TransactionResult transactionResult,
            String pointOfFailure,
            blng.TransactionResult.GatewayStatusType gatewayStatus,
            Exception e) {
        String newErrorMessage = pointOfFailure + (e != null ? ': ' + e.getMessage() :
'');

        // if there are previous errors, set the response message by concatenating them
in the right order
        if (!transactionResult.getErrors().isEmpty()) {
            String prevErrorMessage = '';
            for (String err : transactionResult.getErrors()) {
                prevErrorMessage += (String.isEmpty(prevErrorMessage) ? '' : '. ') + err;

            }
            transactionResult.setResponseMessage(prevErrorMessage);
        } else {
            transactionResult.setResponseMessage(newErrorMessage);
        }

        // set the rest of the values
        transactionResult.setIsSuccess(false);
        transactionResult.setError(newErrorMessage);
        transactionResult.setResponseToValidate(FAILURE);
        if (transactionResult.getResponseCode() != null) {
            populateGatewayStatus(transactionResult);
        } else {
            // set the response code to ERROR if it does not exist, which will be in all
cases except when the response returns from the gateway
            transactionResult.setResponseCode(ERROR);
            transactionResult.setGatewayStatus(gatewayStatus);
        }
    }

    /**
    * @name populateTransactionResultForVoidToken
    * @description Method to populate Transaction Result for CyberSource Void Token
    * @param Map[Key => String [unique Id],Value => TransactionParameter]
    * @return Map[Key => String [unique Id],Value => TransactionResult]
    */
```

```
    public Map<String, blng.TransactionResult>
populateTransactionResultForVoidToken(Map<String, blng.TransactionParameter>
mapOfTransactionParameterById)
    {
        Map<String, blng.TransactionResult> mapOfTransactionResultById = new Map<String,
blng.TransactionResult>();
        for(String idToProcess : mapOfTransactionParameterById.KeySet())
        {
            blng.TransactionParameter transactionParameterToProcess;
            try {
                transactionParameterToProcess =
mapOfTransactionParameterById.get(idToProcess);
                if (NULL !=
transactionParameterToProcess.transactionResult.getGatewayStatus()) {
                    mapOfTransactionResultById.put(idToProcess,
transactionParameterToProcess.transactionResult);
                    continue;
                }
                //Populating the transaction Parameter class based on the response is
success or failure
                if(transactionParameterToProcess.getResponseValueByKey().get(DECISION) ==
 ACCEPT)
                {
                    //The response is success

transactionParameterToProcess.transactionResult.setResponseToValidate(SUCCESS);
                    transactionParameterToProcess.transactionResult.setIsSuccess(true);
                }
            else if(transactionParameterToProcess.getResponseValueByKey().get(DECISION)
 == REJECT)
                {
                    //The response is failure

transactionParameterToProcess.transactionResult.setResponseToValidate(FAILURE);
                    transactionParameterToProcess.transactionResult.setIsSuccess(false);
                }
                //Populating payment gateway response
                transactionParameterToProcess.transactionResult.setId(idToProcess);
                transactionParameterToProcess.transactionResult.setResponseCode
                 (transactionParameterToProcess.getResponseValueByKey().get(REASONCODE));

            if(NULL != transactionParameterToProcess.transactionResult.getResponseCode())

                {
                    transactionParameterToProcess.transactionResult.setResponseCodeMessage

(YourGatewayNameUtils.getGatewayReturnCode().get(transactionParameterToProcess.transactionResult.getResponseCode()));

                }
                //Populating payment gateway response
                transactionParameterToProcess.transactionResult.setPaymentToken
```

```
(transactionParameterToProcess.getResponseValueByKey().get(SUBSCRIPTIONID));
                //Populate gateway status
                populateGatewayStatus(transactionParameterToProcess.transactionResult);

mapOfTransactionResultById.put(idToProcess,transactionParameterToProcess.transactionResult);

            } catch (Exception e) {
                handleError(transactionParameterToProcess.transactionResult,
FAILED_POPULATING_RESULT + IN_VOID_TOKEN,
blng.TransactionResult.GatewayStatusType.SystemError, e);
            }
        }
        //Return map of transaction results
        return mapOfTransactionResultById;
    }




    // default Gateway Status
    private static final blng.TransactionResult.GatewayStatusType defaultGatewayStatus =
blng.TransactionResult.GatewayStatusType.Indeterminate;

    //Generate Token method Salesforce billing interface class will use this class to
Generate token
    public static Map<String, blng.TransactionResult> generateToken(Map<String,
blng.TransactionParameter> mapOfTransactionParameterById)
    {
       //Actual Implementation for Token takes place here
    }

    //Void Token method Salesforce billing interface class will use this class to void
Token Transaction
    public static Map<String, blng.TransactionResult> voidTokenTransaction(Map<String,
blng.TransactionParameter> mapOfTransactionParameterById)
    {
       //Actual Implementation for Void Token takes place here
       if(!mapOfTransactionParameterById.IsEmpty())
        {
            for(blng.TransactionParameter transactionParameterToProcess :
mapOfTransactionParameterById.Values())
            {
                // validate gateway settings
                try {

YourGatewayNameUtils.validateGatewaySettings(transactionParameterToProcess);
                    if
(!transactionParameterToProcess.transactionResult.getErrors().IsEmpty()) {
                        handleError(transactionParameterToProcess.transactionResult,
FAILED_GATEWAY_SETTINGS + IN_VOID_TOKEN,
blng.TransactionResult.GatewayStatusType.ValidationError, null);
                        continue;
                    }
                } catch (Exception e) {
                    handleError(transactionParameterToProcess.transactionResult,
```

```
FAILED_GATEWAY_SETTINGS + IN_VOID_TOKEN,
blng.TransactionResult.GatewayStatusType.ValidationError, e);
                    continue;
                }
                // Calling YourGatewayNameUtils class to populate the Request for generate
 void token XML
                try {

YourGatewayNameUtils.getInstance().generateVoidTokenXML(transactionParameterToProcess);
                } catch (Exception e) {
                    handleError(transactionParameterToProcess.transactionResult,
FAILED_XML_GENERATION + IN_VOID_TOKEN, blng.TransactionResult.GatewayStatusType.SystemError,
 e);
                    continue;
                }
                try {
                    // Calling YourHttpService class to send a Request
                    YourHttpService sendHttpRequest = YourHttpService.getInstance();
                    sendHttpRequest.addHeader('Content-type', 'text/xml');

sendHttpRequest.setTokenisationHeader(transactionParameterToProcess.getGateWay().MerchantId__c,
 transactionParameterToProcess.getGateWay().TransactionSecurityKey__c);
                    // Sends the request to payment gateway

sendHttpRequest.post(transactionParameterToProcess.getGateWay().TestMode__c ?
YOUR_GATEWAY_TEST_ENDPOINT_URL_SANDBOX :
YOUR_GATEWAY_TEST_ENDPOINT_PRODUCTION,transactionParameterToProcess.getRequestBody());
                    if(!Test.isRunningTest())
                    {
                        // Populating the map of Response for transaction parameter class
 from the response received by payment gateway
                        transactionParameterToProcess.mapOfResponseValueByKey.putAll

(YourGatewayNameUtils.getElements(sendHttpRequest.getResponse().getBodyDocument().getRootElement()));

                    }
                    else
                    {
                        Dom.Document doc = new Dom.Document();
                        doc.load(TEST_RESPONSE_BODY); // You can provide a string for
TEST_RESPONSE_BODY in test mode
                        transactionParameterToProcess.mapOfResponseValueByKey.putAll
                        (YourGatewayNameUtils.getElements(doc.getRootElement()));
                    }
                } catch (Exception e) {
                    handleError(transactionParameterToProcess.transactionResult,
FAILED_CALLOUT + IN_VOID_TOKEN, blng.TransactionResult.GatewayStatusType.SystemError, e);

                }
            }
        }
        // Calling populate Transaction Result For Void Token class to Return map of
transaction results
        return populateTransactionResultForVoidToken(mapOfTransactionParameterById);
```

444

```
    }

    //Authorize Transaction method Salesforce billing interface class will use this class
 to Authorize Transaction
    public static Map<String, blng.TransactionResult> authorizeTransaction(Map<String,
blng.TransactionParameter> mapOfTransactionParameterById)
    {
        //Actual Implementation for Authorize takes place here
    }

    //Capture Transaction method Salesforce billing interface class will use this class
to capture Transaction
    public static Map<String, blng.TransactionResult> captureTransaction(Map<String,
blng.TransactionParameter> mapOfTransactionParameterById)
    {
        //Actual Implementation for capture takes place here
    }

    //Charge Transaction method Salesforce billing interface class will use this class to
 Charge Transaction
    public static Map<String, blng.TransactionResult> chargeTransaction(Map<String,
blng.TransactionParameter> mapOfTransactionParameterById)
    {
        //Actual Implementation for Charge takes place here
    }

    //void Transaction method Salesforce billing interface class will use this class to
void Transaction
    public static Map<String, blng.TransactionResult> voidTransaction(Map<String,
blng.TransactionParameter> mapOfTransactionParameterById)
    {
        //Actual Implementation for void Transaction takes place here
    }

    //RefundTransaction method Salesforce billing interface class will use this class to
Refund Transaction
    public static Map<String, blng.TransactionResult> refundTransaction(Map<String,
blng.TransactionParameter> mapOfTransactionParameterById)
    {
        //Actual Implementation for Refund takes place here
    }

    //Non Referenced Refund method Salesforce billing interface class will use this class
 to non referenced refund Transaction
    public static Map<String, blng.TransactionResult> nonReferencedRefund(Map<String,
blng.TransactionParameter> mapOfTransactionParameterById)
    {
        //Actual Implementation for non referenced refund takes place here
    }

    //void Refund method Salesforce billing interface class will use this class to non
void Refund Transaction
    public static Map<String, blng.TransactionResult> voidRefundTransaction(Map<String,
blng.TransactionParameter> mapOfTransactionParameterById)
```

```
    {
        //Actual Implementation for void Refund Transaction takes place here
    }

    //Get payment status method Salesforce billing interface class will use this class to
 get payment status Transaction
    public static Map<String, blng.TransactionResult> getPaymentStatus(Map<String,
blng.TransactionParameter> mapOfTransactionParameterById)
    {
        //Actual Implementation for get payment status takes place here
    }

    //Get refund status method Salesforce billing interface class will use this class to
get refund status Transaction
    public static Map<String, blng.TransactionResult> getRefundStatus(Map<String,
blng.TransactionParameter> mapOfTransactionParameterById)
    {
        //Actual Implementation for get refund status takes place here
    }
    /**
     * Does a SOQL lookup on the mapper table and gets the gateway status mapped to the
return code
     * Returns the default enum if no match is found
     * @param transactionResult
     */
    public void populateGatewayStatus(blng.TransactionResult transactionResult) {
        blng.TransactionResult.GatewayStatusType gatewayStatus = defaultGatewayStatus;
        // actual implementation of mapping
        transactionResult.setGatewayStatus(gatewayStatus);
    }
}
```

YourGatewayNameUtils

YourGatewayNameUtils is a singleton utility class that interacts with an external payment gateway.
All user gateway API classes will be different based on your needs. However, we've provided a
template that you can use to help create your own.

This code shows an example of how to generate a void token request for CyberSource. The class
contains several methods.

- A utility method to construct the void token request payload for CyberSource
- A utility method to validate gateway settings
- A utility to return a map of CyberSource Gateway Return code to a friendly description
- A utility method to pass the XML node and retrieve a map of element to value

```
public class YourGatewayNameUtils
{

    // ===========================================================================
    // STATIC VARIABLES
    // ===========================================================================

    // private Attribute to implement singleton pattern for YourGatewayNameUtils class
```

```
    private static YourGatewayNameUtils yourGatewayNameUtilsInstance;



    /**
     * @name getInstance
     * @description get an Instance of Service class
     * @param NA
     * @return YourGatewayNameUtils Generator Class Instance
     */
    public static YourGatewayNameUtils getInstance() {
        if (NULL == yourGatewayNameUtilsInstance) {
            yourGatewayNameUtilsInstance = new YourGatewayNameUtils();
        }
        return yourGatewayNameUtilsInstance;
    }

    /**
    * @name validateGatewaySettings
    * @description Validates Gateway Details
    * @param TransactionParameter
    * @return NA
    * @exception NA
    */
    public static void validateGatewaySettings(blng.TransactionParameter
transactionParameterToProcess) {
        if (String.IsBlank(transactionParameterToProcess.getGateWay().MerchantId__c)) {
            transactionParameterToProcess.transactionResult.setError('MerchantId is
missing');
        } else if
(String.IsBlank(transactionParameterToProcess.getGateWay().MerchantReference__c)) {
            transactionParameterToProcess.transactionResult.setError('MerchantReference
is missing');
        } else if
(String.IsBlank(transactionParameterToProcess.getGateWay().TransactionSecurityKey__c)) {
            transactionParameterToProcess.transactionResult.setError('TransactionSecurityKey
 is missing');
        }
    }

        /**
        * @name getErrorCyberSourceCode
        * @description Return's Map of Cyber source error description By code
        * @param NA
        * @return Map [Key => String [Code] , Value => String [Error Message]]
        */
     /// include mapping of all  codes based on gateway you are implementing
    public static map<string, string> getGatewayReturnCode() {
        Map<string, string> mapOfMessageByErrorCode = new Map<string, string>();
        mapOfMessageByErrorCode.put('100', 'Successful transaction');
        mapOfMessageByErrorCode.put('101', 'The request is missing one or more required
fields.
            Possible action: see the reply fields missingField_0...N for which fields are
missing. Resend the request with the complete information. For information about missing
```

447

```
or invalid fields');
        mapOfMessageByErrorCode.put('102', 'Invalid data');
        mapOfMessageByErrorCode.put('151', 'This error does not include timeouts between
the client and the server.
            To avoid duplicating the transaction, do not resend the request until you have
 reviewed the transaction status at the Business Center');
        mapOfMessageByErrorCode.put('152', 'The request was received, but a service did
not finish running in time.
            To avoid duplicating the transaction, do not resend the request until you have
 reviewed the transaction status at the Business Center');
        mapOfMessageByErrorCode.put('201', 'The issuing bank has questions about the
request.
            You will not receive an authorization code programmatically, but you can obtain
 one verbally by calling the processor');
      /// include mapping of all error codes based on gateway you are implementing
        return mapOfMessageByErrorCode;
    }


    /**
     * @name getElements
     * @description Populates's map Of Response Value By Key
     * @param Dom XML Node
     * @return Map [Key => String [Name] , Value => String [text]]
     * @exception NA
     */
    public static Map<string, string> getElements(DOM.XMLNode node) {
        if (node.getNodeType() == DOM.XMLNodeType.ELEMENT) {
            if (String.IsNotBlank(node.getText().trim())) {
                mapOfResponseValueByKey.put(node.getName(), node.getText().trim());
            }

            for (Dom.XMLNode child : node.getChildElements()) {
                getElements(child);
            }
        }
        return mapOfResponseValueByKey;
    }


    public void generateToken(List<blng.TransactionParameter> listOfTransactionParameter)

    {
        //Frame xml/json for generate token api method
    }


    // Example from Cybersource to generate a voidToken Request
    public void generateVoidToken(List<blng.TransactionParameter> listOfTransactionParameter)

    {
        //Frame xml/json for void token api method
         XmlStreamWriter writer = new XmlStreamWriter();
        // Populate Document start
```

```
        writer.writeStartDocument('utf-8', '1.0');
        // Populate Envelope start
        writer.writeStartElement('s', 'Envelope',
'http://schemas.xmlsoap.org/soap/envelope/');
        writer.writeAttribute('xmlns', 'http://schemas.xmlsoap.org/soap/envelope/', 's',
'http://schemas.xmlsoap.org/soap/envelope/');
        // Populate Header start
       writer.writeStartElement('s', 'Header', 'http://schemas.xmlsoap.org/soap/envelope/');

        // Populate Security start
        writer.writeStartElement('wsse', 'Security',
'http://schemas.xmlsoap.org/soap/envelope/');
        writer.writeAttribute('xmlns',
'http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd', 'wsse',
 'http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd');
        // Populate UsernameToken start
        writer.writeStartElement('wsse', 'UsernameToken', '');
        // Populate the Username start
        writer.writeStartElement('wsse', 'Username', '');
        writer.writeCharacters(transactionParameterToProcess.getGateWay().MerchantId__c);

        writer.writeEndElement();
        // Populate the Username end
        // Populate the Password start
        writer.writeStartElement('wsse', 'Password', '');
        writer.writeAttribute(NULL, NULL, 'Type',
'http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText');


writer.writeCharacters(transactionParameterToProcess.getGateWay().TransactionSecurityKey__c);

        writer.writeEndElement();
        // Populate the Password end
        writer.writeEndElement();
        // Populate UsernameToken end
        writer.writeEndElement();
        // Populate Security end
        writer.writeEndElement();
        // Populate Header end
        // Populate Body start
        writer.writeStartElement('s', 'Body', '');
        // Populate RequestMessage start
        writer.writeStartElement('', 'requestMessage', '');
        writer.writeAttribute('', '', 'xmlns',
'urn:schemas-cybersource-com:transaction-data-1.137');
        // Populate merchantID start
        writer.writeStartElement('', 'merchantID', '');
        writer.writeCharacters(transactionParameterToProcess.getGateWay().MerchantId__c);

        writer.writeEndElement();
        // Populate merchantID end
        // Populate merchantReferenceCode start
        writer.writeStartElement('', 'merchantReferenceCode', '');
```

449

```
writer.writeCharacters(transactionParameterToProcess.getGateWay().MerchantReference__c);
        writer.writeEndElement();
        // Populate merchantReferenceCode end
        // Populate recurringSubscriptionInfo start
        writer.writeStartElement('', 'recurringSubscriptionInfo', '');
        // Populate subscriptionID start
        writer.writeStartElement('', 'subscriptionID', '');
        //Added by DL,W-3931366, for cardmapping

writer.writeCharacters(getCardMapping(transactionParameterToProcess.getPaymentMethod()));

        writer.writeEndElement();
        // Populate subscriptionID end
        writer.writeEndElement();
        // Populate recurringSubscriptionInfo end
        // Populate paySubscriptionDeleteService start
        writer.writeStartElement('', 'paySubscriptionDeleteService', '');
        writer.writeAttribute('', '', 'run', 'true');
        writer.writeEndElement();
        // Populate paySubscriptionDeleteService end
        // Populate RequestMessage end
        writer.writeEndElement();
        // Populate Body end
        writer.writeEndElement();
        // Populate Envelope end
        writer.writeEndDocument();
        transactionParameterToProcess.setRequestBody(writer.getXmlString());
        writer.close();
    }

    public void generateAuthorize(Map<String, blng.TransactionParameter>
mapOfTransactionParameterByInvoiceId)
    {
        //Frame xml/json for generate authorize api method
    }

    public void generateCapture(Map<String, blng.TransactionParameter>
mapOfTransactionParameterByInvoiceId)
    {
        //Frame xml/json for generate capture api method
    }

    public void generateCharge(Map<String, blng.TransactionParameter>
mapOfTransactionParameterByInvoiceId)
    {
        //Frame xml/json for generate charge api method
    }

    public void generateVoid(Map<String, blng.TransactionParameter>
mapOfTransactionParameterByInvoiceId)
    {
        //Frame xml/json for generate void api method
    }
```

```
    public void generateRefund(Map<String, blng.TransactionParameter>
mapOfTransactionParameterByInvoiceId)
    {
        //Frame xml/json for generate refund api method
    }

     public void generatenonReferencedRefund(Map<String, blng.TransactionParameter>
mapOfTransactionParameterByInvoiceId)
    {
        //Frame xml/json for generate non referenced refund api method
    }

    public void generateVoidRefund(Map<String, blng.TransactionParameter>
mapOfTransactionParameterByInvoiceId)
    {
        //Frame xml/json for generate void refund api method
    }

    public void generateGetPaymentStatus(List<blng.TransactionParameter>
listOfTransactionParameter)
    {
        //Frame xml/json for generate get payment status api method
    }

    public void generateGetRefundStatus(List<blng.TransactionParameter>
listOfTransactionParameter)
    {
        //Frame xml/json for generate get refund status api method
    }
}
```

YourHttpService

YourHttpService is an example of common HTTP utilities that might be required while building a
partner package. All user gateway API classes will be different based on your needs. However, we've
provided a template that you can use to help create your own.

```
public with sharing class YourHttpService {
    // ============================================================================
    // CONSTANT
    // ============================================================================

    // ============================================================================
    // STATIC VARIABLES
    // ============================================================================

    private static YourHttpService httpServiceInstance;

    // ============================================================================
    // VARIABLES
    // ============================================================================
```

```
    private HttpResponse httpResponse;
    private Map<String,String> mapOfHeaderParameter = new Map<String,String>();
    private enum Method {GET, POST}

    /**
    * @name getInstance
    * @description get an Instance of Service class
    * @params NA
    * @return Http Service Class Instance
    * @remark
    * @change
    */
    public static YourHttpService getInstance()
    {
        if (NULL == httpServiceInstance)
        {
            httpServiceInstance =  new YourHttpService();
        }
        return httpServiceInstance;
    }

    /**
    * @name get
    * @description Get Method to get a HTTP request
    * @param endPoint
    * @return
    * @remark
    * @change
    */
    public void get(String endPoint)
    {
        send(newRequest(Method.GET, endPoint));
    }

    /**
    * @name post
    * @description Post Method to Post a HTTP request
    * @param endPoint and requestBody
    * @return
    * @exception
    * @remark
    * @change
    */
    public void post(String endPoint, String requestBody)
    {
        send(newRequest(Method.POST, endPoint, requestBody));
    }

    /**
    * @name addHeader
    * @description addHeader Methods to add all the defualt Header's required fo rthe
request
    * @param name and value
```

```
    * @return
    * @exception
    * @remark
    * @change
    */
    public void addHeader(String name, String value)
    {
        mapOfHeaderParameter.put(name, value);
    }

    /**
    * @name setHeader
    * @description setHeader Methods to set setHeader for the request
    * @param request
    * @return
    */
    private void setHeader(HttpRequest request)
    {
        for(String headerValue : mapOfHeaderParameter.keySet())
        {
            request.setHeader(headerValue, mapOfHeaderParameter.get(headerValue));
        }
    }

    /**
    * @name setAuthorizationHeader
    * @description setAuthorizationHeader Methods to set Authorization Header for the
request
    * @param userName and password
    * @return
    */
    public void setAuthorizationHeader(String userName,String password)
    {
        if(String.isNotBlank(userName) && String.isNotBlank(password))
        {
            Blob headerValue = blob.valueOf(userName + ':' + password);
           String authorizationHeader = 'Basic ' + EncodingUtil.base64Encode(headerValue);

            addHeader('Authorization', authorizationHeader);
        }
    }

    /**
     * @name setTokenisationHeader
    * @description setTokenisationHeader Methods to set Tokenisation Header for the request

     * @param userName and password
    * @return
    */
    public void setTokenisationHeader(String userName,String password)
    {
        if(String.isNotBlank(userName) && String.isNotBlank(password))
        {
            Blob headerValue = blob.valueOf(userName + ':' + password);
```

```
            String tokenisationHeader = 'Basic ' + EncodingUtil.base64Encode(headerValue);

            addHeader('Tokenisation', tokenisationHeader);
        }
    }

    /**
    * @name newRequest
    * @description newRequest Methods to make a new request
    * @param method and endPoint
    * @return newRequest
    * @exception
    */
    private HttpRequest newRequest(Method method, String endPoint)
    {
        return newRequest(method, endPoint, NULL);
    }

    /**
    * @name newRequest
    * @description newRequest Methods to make a new request
    * @param method, endPoint and requestBody
    * @return request
    * @exception
    */
    private HttpRequest newRequest(Method method, String endPoint, String requestBody)
    {
        HttpRequest request = new HttpRequest();
        request.setMethod(Method.name());
        setHeader(request);
        request.setEndpoint(endPoint);
        if (String.isNotBlank(requestBody))
        {
            request.setBody(requestBody);
        }
        request.setTimeout(120000);
        return request;
    }

    /**
    * @name send
    * @description send Methods to send a request
    * @param request
    * @return
    * @exception Throws Exception
    */
    private void send(HttpRequest request)
    {
        try
        {
            httpResponse = new Http().send(request);
        }
        catch(Exception e)
        {
```

```
            throw e;
        }
    }

    /**
     * @name getResponse
     * @description getResponse Method to get the Response
     * @param NA
     * @return httpResponse
     * @exception
     */
    public HttpResponse getResponse()
    {
        return httpResponse;
    }

    /**
     * @name getResponseToString
     * @description getResponse Method to get the Response
     * @param NA
     * @return getResponse
     * @exception
     */
    public String getResponseToString()
    {
        return getResponse().toString();
    }

}
```

Salesforce Billing Tax Integration Developer Guide

Use Salesforce Billing tax integration API to enable communication between Salesforce Billing and external tax engines.

Get Started With Tax Integration API

Configuring settings in the Salesforce Billing package so that your org can communicate with an external tax engine.

Tax Integration Core Classes

Your tax engine references these classes to manage tax information with Salesforce Billing. They can't be modified.

Get Started With Tax Integration API

Configuring settings in the Salesforce Billing package so that your org can communicate with an external tax engine.

Required Tax Integration Fields

All tax integration records require several Salesforce Billing fields to communicate with tax engines, regardless of tax engine type. Certain tax engines also require fields not mentioned here.

Configure Tax Integrations in Salesforce Billing

Add custom settings and remote site settings for your tax engine in Salesforce Billing.

Required Tax Integration Fields

All tax integration records require several Salesforce Billing fields to communicate with tax engines, regardless of tax engine type. Certain tax engines also require fields not mentioned here.

Tax Integration Fields

If you need more fields to hold other information regarding your tax engine, add them on the tax integration object.

Label	API Name	Notes
Active	blng__Active__c	If this field is inactive, tax treatments won't list this integration as a possible value for their Tax Integration field.
Default	blng__Default__c	Salesforce Billing uses the default tax integration for all tax calculations in your Salesforce Billing org. You can have only one default tax integration at once.
Tax Engine	blng__TaxEngine__c	Declares the type of tax engine that Salesforce Billing uses for tax calculation. Standard and Avalara AvaTax are available by default. If you're using a different tax engine, add it as a picklist value.

Configure Tax Integrations in Salesforce Billing

Add custom settings and remote site settings for your tax engine in Salesforce Billing.

1. Create a custom setting for the class that Salesforce Billing calls and sends to your tax package.

   a. From Setup, in the Quick Find box, enter `Custom Settings`, and then select **Custom Settings**.

   b. Select **Tax Config**.

   c. Select **Manage**, and then select **New**.

   d. Enter the name of your tax engine.

   e. Enter the tax package class name that your tax engine will call for tax calculations. The class name should have the format `{prefix.classname}`.

2. Create a custom setting for the tax engine that Salesforce Billing calls and sends to your tax package.

   a. From Setup, in the Quick Find box, enter `Custom Settings`, and then select **Custom Settings**.

   b. Select **Billing Config**.

   c. Select **Manage**, and then select **New**.

   d. Enter the name of your tax engine.

   e. In the Category field, enter `Tax`.

    f. In the String Value field, enter the name of your tax engine. This value must match the Tax Engine field on the tax integration record for the tax integration that you're configuring. For example, if your tax integration has a value of Avalara Avatax, your String Value field must have the same value.

3. Configure remote site settings.

    a. From Setup, in the Quick Find box, enter `Remote Site Settings`, and then select **Remote Site Settings**.

    b. Select **New Remote Site**.

    c. Add the domain production URL for your tax engine.

    d. If needed, create another remote site for your tax engine's domain sandbox URL.

Tax Integration Core Classes

Your tax engine references these classes to manage tax information with Salesforce Billing. They can't be modified.

The following are the `blng` namespace classes required for integrating Salesforce Billing with an external tax engine.

TaxParameters Class
Contains a list of setters to populate the data needed for a tax calculation. The data is passed in the tax callout.

TaxResults Class
Contains a list of setter methods to populate the data returned from a tax callout. The data is passed back to Salesforce Billing from the tax connector.

TaxEngines Interface
Your tax engine implements this interface. It contains methods for calculating tax, clearing in-memory entities from the previous call, and clearing legal references from the previous call.

TaxParameters Class

Contains a list of setters to populate the data needed for a tax calculation. The data is passed in the tax callout.

Namespace
blng

TaxParameters Methods
Learn more about the methods available with the `TaxParameters` class.

TaxParameters Methods

Learn more about the methods available with the `TaxParameters` class.

The `TaxParameters` class includes these methods.

getIsCommitOnGetTax()
Returns the value of the `isCommitOnGetTax` parameter.

getLegalEntityReferences()
Returns the legal entity references.

getSObjectID()
Returns the ID of a single object that requires tax calculation, if any.

getSObjectListofIds()
Returns the list of IDs for objects that require tax calculation.

getSObjectType()
Returns the object type, which can be InvoiceLine, DebitNoteLine, CreditNoteLine, and OrderProduct object.

getTaxMethod()
Returns the type of tax method, which is POST or GET.

setSObjectListofIds(listOfSobjectIds)
Sets the list of IDs for objects that require tax calculation.

setSObjectID(sObjectId)
Sets the ID of a single object that requires tax calculation., if any.

setLegalEntityReferences(legalEntityReference)
Sets the legal entity for the tax treatment. For example, a company can create legal entities to represent their different branches.

setSObjectType(sObjectTypeInstance)
Sets the object type to InvoiceLine, DebitNoteLine, CreditNoteLine, or OrderProduct object.

setTaxMethod(taxMethod)
Sets the method to POST or GET.

setIsCommitOnGetTax(isCommitOnGetTax)
Indicates to external tax providers whether to commit tax.

getIsCommitOnGetTax()

Returns the value of the isCommitOnGetTax parameter.

Signature

```
global Boolean getIsCommitOnGetTax()
```

Parameters

isCommitOnGetTax
    Type: Boolean

Return Value

Type: Void

getLegalEntityReferences()

Returns the legal entity references.

Signature

```
global static void getLegalEntityReferences()
```

Parameters

legalEntityReference
    Type: String

Return Value

Type: void

getSObjectID()

Returns the ID of a single object that requires tax calculation, if any.

Signature

```
global static void getSObjectID()
```

Parameters

sObjectId
    Type: Id

Return Value

Type: void

getSObjectListofIds()

Returns the list of IDs for objects that require tax calculation.

Signature

```
global List<Id> getSObjectListOfIds()
```

Parameters

listOfSobjectIds
    Type: List<Id>

Return Value

Type: Void

getSObjectType()

Returns the object type, which can be InvoiceLine, DebitNoteLine, CreditNoteLine, and OrderProduct object.

Signature

```
global static void getSObjectType()
```

Parameters

sObjectTypeInstance
   Type: SObjectType

Return Value

Type: void

getTaxMethod()

Returns the type of tax method, which is `POST` or `GET`.

Signature

```
global static void setTaxMethod()
```

Parameters

taxMethod
   Type: String

Return Value

Type: void

setSObjectListofIds(listOfSobjectIds)

Sets the list of IDs for objects that require tax calculation.

Signature

```
global static Void setSObjectListofIds(List<Id> listOfSobjectIds)
```

Parameters

listOfSobjectIds
   Type: List<Id>

Return Value

Type: Void

setSObjectID(sObjectId)

Sets the ID of a single object that requires tax calculation., if any.

Signature

```
global static void setSObjectID(Id sObjectId)
```

Parameters

sObjectId
    Type: Id

Return Value

Type: void

setLegalEntityReferences(legalEntityReference)

Sets the legal entity for the tax treatment. For example, a company can create legal entities to represent their different branches.

Signature

```
global static void setLegalEntityReferences(String legalEntityReference)
```


Parameters

legalEntityReference
    Type: String

    Unique identification number of the legal entity.

Return Value

Type: void

setSObjectType(sObjectTypeInstance)

Sets the object type to InvoiceLine, DebitNoteLine, CreditNoteLine, or OrderProduct object.

Signature

```
global static void setSObjectType(SObjectType sObjectTypeInstance)
```


Parameters

sObjectTypeInstance
    Type: SObjectType

Return Value

Type: void

setTaxMethod(taxMethod)

Sets the method to POST or GET.

Signature

```
global static void setTaxMethod(String taxMethod)
```

Parameters

taxMethod
　Type: String

Return Value

Type: void

setIsCommitOnGetTax(isCommitOnGetTax)

Indicates to external tax providers whether to commit tax.

Usage

If the **Automatically Post Invoices** checkbox is selected on the Invoice Scheduler, tax providers examine the `isCommitOnGetTax` parameter to commit the tax.

The `isCommitOnGetTax` parameter value is configured within the `Taxparameter` instance and transmitted to the external tax provider implementation.

Signature

<code>global static Void setIsCommitOnGetTax(Boolean isCommitOnGetTax)</code>

Parameters

isCommitOnGetTax
　Type: Boolean

　Indicates to external tax providers whether to commit tax on the Get Tax method (`true`) or not (`false`).

Return Value

Type: Void

TaxResults Class

Contains a list of setter methods to populate the data returned from a tax callout. The data is passed back to Salesforce Billing from the tax connector.

Namespace

blng

[TaxResults Methods](#)

TaxResults Methods

The following are methods for `TaxResults`.

getId()

Returns the ID of the entity where tax was calculated.

getRate()

Returns the tax rate.

getSummary()

Returns the summary that contains the error message.

getTax()

Returns the tax amount.

isSuccess

Returns `true` if tax calculation was successful or `false` if tax calculation fails.

setRate(rate)

Sets the tax rate to the percentage specified by the parameter.

setTax(tax)

Sets the tax amount.

setIsSuccess(isSuccess)

Sets `isSuccess` to true if tax calculation was successful, otherwise sets isSuccess to false.

setId(id)

Sets the ID of the entity where tax was calculated.

setSummary(summary)

If tax isn't calculated successfully, assign error message to the summary.

getId()

Returns the ID of the entity where tax was calculated.

Signature

```
global static void getId()
```

Parameters

id
    Type: String

Return Value

Type: void

getRate()

Returns the tax rate.

Signature

```
global static void getRate()
```

Parameters

rate
    Type: Decimal

Return Value

Type: void

getSummary()

Returns the summary that contains the error message.

Signature

```
global static void getSummary()
```

Parameters

summary
    Type: String

Return Value

Type: void

getTax()

Returns the tax amount.

Signature

```
global static void getTax()
```

Parameters

tax
    Type: Decimal

Return Value

Type: void

isSuccess

Returns `true` if tax calculation was successful or `false` if tax calculation fails.

Signature

```
global static void isSuccess()
```

Parameters

isSuccess
> Type: Boolean

Return Value

Type: void

setRate(rate)

Sets the tax rate to the percentage specified by the parameter.

Signature

```
global static void setRate(Decimal rate)
```

Parameters

rate
> Type: Decimal

Return Value

Type: void

setTax(tax)

Sets the tax amount.

Signature

```
global static void setTax(Decimal tax)
```

Parameters

tax
> Type: Decimal
>
> Assigns tax.

Return Value

Type: void

setIsSuccess(isSuccess)

Sets `isSuccess` to true if tax calculation was successful, otherwise sets isSuccess to false.

Signature

```
global static void setIsSuccess(Boolean isSuccess)
```

Parameters

isSuccess
    Type: Boolean

Return Value

Type: void

setId(id)

Sets the ID of the entity where tax was calculated.

Signature

```
global static void setId(String id)
```

Parameters

id
    Type: String

```
In case of GET Tax Call: Assign Invoiceline/Orderproduct Id
In case of Post/Cancel Tax Call: Assign Invoice Id
```

Return Value

Type: void

setSummary(summary)

If tax isn't calculated successfully, assign error message to the summary.

Signature

```
global static void setSummary(String summary)
```

Parameters

summary
    Type: String

Return Value

Type: void

TaxEngines Interface

Your tax engine implements this interface. It contains methods for calculating tax, clearing in-memory entities from the previous call, and clearing legal references from the previous call.

Namespace

blng

TaxEngines Methods
Learn more about the methods available with the TaxEngines interface.

TaxEngines Methods

Learn more about the methods available with the TaxEngines interface.

The `TaxEngines` interface includes these methods.

resetSObjectListofIds()
Sets the in-memory IDs, such as credit note IDs, debit note IDs, to null.

resetLegalEntityReferences()
Sets the in-memory list of legal entities to null.

calculateTax(taxParametersInstance)
Returns the calculated tax for the specified `taxParametersInstance` parameter.

resetSObjectListofIds()

Sets the in-memory IDs, such as credit note IDs, debit note IDs, to null.

Signature

```
global static Void resetSObjectListofIds()
```

Return Value

Type: Void

resetLegalEntityReferences()

Sets the in-memory list of legal entities to null.

Signature

```
global static Void resetLegalEntityReferences()
```

Return Value

Type: Void

calculateTax(taxParametersInstance)

Returns the calculated tax for the specified `taxParametersInstance` parameter.

Signature

```
global static Map<Id,TaxResults> calculateTax(TaxParameters taxParametersInstance)
```

Parameters

taxParametersInstance
  Type: TaxParameters

  The `taxParametersInstance` parameter is populated based on the process that uses tax calculation. These examples show
  the sample values for each process.

  Order is created

```
TaxParameters:[
 isCommitOnGetTax=false,
 isStandardFromBatch=false,
 legalEntityReference=80105000000i1MqAAI,
 listOfSObjectIds=(),
 listOfSObjects=(),
 mapOfSObjectByUniqueId={},
 mapOfTaxResultsById={},
 mapOfTaxTreatmentById={},
 sObjectId=null,
 sObjectTypeInstance=OrderItem,
 sobjectInstance=null,
 taxMethod=null
]
```

  Bill Now or Invoice is created

```
TaxParameters:[
 isCommitOnGetTax=false,
 isStandardFromBatch=false,
 legalEntityReference=a2305000000BKIoAAO,
 listOfSObjectIds=(),
 listOfSObjects=(),
 mapOfSObjectByUniqueId={},
 mapOfTaxResultsById={},
 mapOfTaxTreatmentById={},
 sObjectId=null,
 sObjectTypeInstance=blng__InvoiceLine__c,
 sobjectInstance=null,
 taxMethod=null
]
```

  Invoice is manually posted

```
TaxParameters:[
isCommitOnGetTax=false,
 isStandardFromBatch=false,
 legalEntityReference=null,
 listOfSObjectIds=(a2305000000BKIoAAO),
 listOfSObjects=(),
 mapOfSObjectByUniqueId={},
 mapOfTaxResultsById={},
 mapOfTaxTreatmentById={},
```

```
    sObjectId=null,
    sObjectTypeInstance=blng__Invoice__c,
    sobjectInstance=null,
    taxMethod=postTax
]
```

Return Value

Type: Map<Id,TaxResults>

Revenue Recognition Service Developer Guide

Salesforce Billing Revenue Recognition API lets you run revenue recognition for any Salesforce object in response to triggers, process builders, and REST API calls. The service uses Salesforce Billing revenue recognition rules, treatments, and distribution methods to create a revenue schedule and transaction hierarchy.

Get Started with the Revenue Recognition Service

Review the Revenue Recognition Service's input and output parameters, and general guidelines.

Revenue Recognition Service Setup

Prepare the Revenue Recognition Service by configuring several revenue fields in Salesforce Billing.

Revenue Recognition Service Use Cases

Use process builders, APEX triggers, or REST API to call the Revenue Recognition Service.

Get Started with the Revenue Recognition Service

Review the Revenue Recognition Service's input and output parameters, and general guidelines.

The Revenue Recognition service is helpful when you want to forecast revenue on objects other than invoice lines or order products. For example, you can forecast revenue on quote lines once they're approved, or on a contract after it's activated. You can access the service by calling the `blng.recognizeRevenue` method through process builders, workflow rules, REST API, and APEX triggers.

The service takes several input parameters and passes them to Salesforce Billing. You can define these parameters based on equivalent fields on your source object. For example, if you want to recognize revenue after a contract is activated, you could set the `startDate` parameter to the contract's Activated Date field, and the `revenueAmount` parameter to the List Amount field from the contract's originating quote.

If the service processes all parameters successfully, it returns parameters indicating the success and creates a revenue schedule with revenue transactions. If there are any errors, the service returns parameters for the number and types of errors encountered.

> 💡 **Tip:** By default, the Revenue Recognition Service API doesn't return success messages after successfully creating a revenue schedule and its transcations. We recommend building automation to confirm that your revenue schedule and revenue transactions were created successfully.

Revenue Recognition Service Input Parameters

Name	Type	Required	Definition
currencyIsoCode	String	No	Required for multicurrency orgs. Defines the currency used in revenue schedules made from this process builder. If not defined in a single-currency org, the revenue recognition service uses the org's default currency.
endDate	Date	Required only for daily or monthly revenue recognition	The date when the revenue recognition period ends.
legalEntityId	ID	No	Required for orgs that use legal entities. Used to determine the correct revenue recognition treatments to use for a revenue recognition rule. ⊘ Important: If your org uses legal entities, and you're using revenue recognition API on an object that doesn't have a Legal Entity field, you must create a custom legal entity field and enter its value on your own, or look up the value from a related record.
revenueAmount	Decimal	Yes	The amount that the revenue recognition service uses to determine available, deferred, recognized, and total revenue based on revenue recognition rules. Users must ensure that the amount value they pass to revenue recognition service has the currency as the `currencyIsoCode` field.
revenueRecognitionRuleId	ID	Yes	The service applies this revenue recognition rule, then applies the rule's appropriate revenue recognition treatment.
source	ID	Yes	ID of the entity that the revenue recognition service evaluates for

Name	Type	Required	Definition
			revenue recognition or forecasting. When you configure your revenue recognition service setup, create a lookup field on the revenue schedule object. This field stores the ID of the record where you're recognizing revenue. When you call the revenue recognition service, pass the source record's ID to the lookup field.
sourceFieldName	String	Yes	ID of the source record that's displayed in the revenue schedule.
startDate	Date	Yes	The date that the revenue recognition period begins. When the revenue distribution method is set to full recognition, the revenue recognition service uses the start date for full recognition and ignores the end date.

Revenue Recognition Service Output Parameters

Name	Type	Definition
jobId	ID	ID of the APEX job that passed the input parameters to the service.
inputValidationStatus	String	A list of input parameters that encountered validation errors.
totalNumberOfInputsWithErrors	Integer	The number of input parameters that the service couldn't use due to errors (for example, a missing startDate, or a revenueAmount with a non-numeric character).
totalNumberOfInputsWithoutErrors	Integer	The number of input parameters that were successfully passed to the service.

Revenue Recognition Service Setup

Prepare the Revenue Recognition Service by configuring several revenue fields in Salesforce Billing.

Your admins may already have configured objects and fields in your org for use with the revenue recognition service. If they haven't, read the following information.

To successfully pass a revenue recognition rule to the revenue recognition service, the rule's revenue distribution methods and revenue recognition treatments require several picklist fields to have a value of `Other`. This value indicates that Salesforce Billing should override the default revenue recognition process and use parameters passed to the APEX service instead. When you're configuring a revenue recognition rule for use with the service, make sure that the following fields have a value of `Other`.

Revenue Distribution Method

Full Recognition Date

Revenue Term End Date (Needed only for daily or monthly recognition)

Revenue Term Start Date

Type

Revenue Recognition Treatment

Revenue Schedule Creation Action

Revenue Schedule Amount

> **Note:** If you're upgrading to Salesforce Billing Spring '20 from an earlier version, add `Other` as a picklist value to each of these fields before using the service.

Revenue Recognition Service Use Cases

Use process builders, APEX triggers, or REST API to call the Revenue Recognition Service.

Using the Revenue Recognition Service With Process Builders

Process builders provide a convenient way to call the revenue recognition service. After you've defined your Revenue Recognition class, your process builder can call the service while passing it key revenue information from the object where you want to recognize revenue.

Calling the Revenue Recognition Service with APEX Triggers

You can call the Revenue Recognition service with a custom APEX trigger.

Revenue Recognition REST API

Use REST API to call invocable methods from the Revenue Recognition Apex class.

Using the Revenue Recognition Service With Process Builders

Process builders provide a convenient way to call the revenue recognition service. After you've defined your Revenue Recognition class, your process builder can call the service while passing it key revenue information from the object where you want to recognize revenue.

When you use a process builder to trigger the revenue recognition process, configure your conditions carefully so you don't accidentally create multiple revenue schedules at once for the same record. For example, if you want to recognize revenue when a contract's status changes to Activated and your trigger condition checks only that the payment's status equals Posted, any value change on an activated contract will trigger your process. Instead, use conditions that trigger the process only when a contract's status *changes* to Activated.

Your process builder calls the `blng.recognizeRevenue` APEX method, which accepts the input parameters defined in Revenue Recognition Service Developer Guide. Take Remember, your RevenueRecognitionInputs class must at least define a Revenue Amount, Revenue Recognition Rule ID, Start Date, ID Source, and Source Field name. Your builder's Apex variables have to pass at least these five variables to successfully invoke the Revenue Recognition class, and any optional variables that you need.

We recommend defining Apex variables that inherit the value of an equivalent field. When you map a field in the process builder, we recommend using a formula field such as $VALUE(TEXT(Object.[Field]))$. Here are a few examples.

📝 **Note:** Currently, currency fields aren't supported as invocable variables, so we use decimals to define the revenueAmount.

Object	Apex Variable Type	Object Field	Apex Variable Value
Quote Line	Formula	SBQQ__ListTotal__c	VALUE(Text([SBQQ__QuoteLine__c].SBQQ__ListTotal__c
Order	Formula	TotalPrice	VALUE(Text([OrderItem].TotalPrice))
Contract	Formula	Amount	VALUE(Text(Contract__c.Amount))

Here are a few ways you can define a value for Start Date.

Object	Apex Variable Type	Object Field	Apex Variable Value
Quote Line	Date	SBQQ__Start__Date__c	[SBQQ__QuoteLine__c].SBQQ__StartDate__c
Order	Date	Effective-Date	[OrderItem].Order.Effective-Date
Contract	Date	ActivatedDate	Contract.ActivatedDate

👁 **Example:** In this example, we want a process builder that recognizes quote line revenue after a quote is approved.

Criteria

Executes when any of the conditions are met.

Conditions

Field	Operator	Type	Picklist
[SBQQ__QuoteLine__c].SBQQ__Quote__r.SBQQ__Status__c	Equals	Picklist	Approved

APEX Variables

Apex Class: Recognize Revenue

Field	Type	Value
startDate	Date	[SBQQ__QuoteLine__c].SBQQ__StartDate__c
sourceFieldName	String	SBQQ__Quote__Line__c
source	Field Reference	[SBQQ__QuoteLine__c].Id
revenueAmount	Formula	VALUE(Text([SBQQ__QuoteLine__c].SBQQ__ListTotal_c))
revenueRecognitionRuleId	Field Reference	[SBQQ__QuoteLine__c].SBQQ__Product__r.blng__RevenueRecognitionRule__r.Id
endDate	Date	[SBQQ__QuoteLine__c].SBQQ__EndDate__c

Field	Type	Value
LegalEntityID	Field Reference	*[SBQQ__QuoteLine__c].LegalEntity__c* 📝 Note: Use a custom quote line field called Legal Entity that looks up to the same legal entity that you plan to use for your order product.

👁 **Example:** In this example, we want a process builder that forecasts revenue when a contract is activated. We've defined a custom *blng__Contract__c* field on the revenue schedule to store the contract's ID value for `sourceFieldName`. Instead of using a variable from the source entity for our end date, we've defined a static date.

Criteria

Executes when all of the conditions are met.

Conditions

Field	Operator	Type	Picklist
Contract	Is changed	Boolean	True
Contract	Equals	Picklist	Posted

APEX Variables

Apex Class: Recognize Revenue

Field	Type	Value
startDate	Date	*Contract.ActivatedDate*
sourceFieldName	String	*blng__Contract__c*
source	Field Reference	*Contract.Id*
revenueAmount	Formula	*VALUE(Text(Contract.blng__Amount__c))*
revenueRecognitionRuleId	Field Reference	*[SBQQ__QuoteLine__c].SBQQ__Product__r.blng__RevenueRecognitionRule__r.Id*
endDate	Date	*12/31/2019*

👁 **Example:** In this example, we want a process builder that forecasts order product revenue after an order product is activated.

Criteria

Executes when all of the conditions are met.

Conditions

Field	Operator	Type	Picklist
[OrderItem].SBQQ__Status__c	Is changed	Boolean	True
[OrderItem].SBQQ__Status__c	Equals	Picklist	Activated

APEX Variables

Apex Class: Recognize Revenue

Field	Type	Value
startDate	Date	*[OrderItem].Order.Effective-Date*
sourceFieldName	String	*blng__OrderProduct__c*
source	Field Reference	*[OrderItem].Id*
revenueAmount	Formula	*[OrderItem].Id*
revenueRecognitionRuleId	Field Reference	*[OrderItem].Product2.blng__RevenueRecognitionRule__r.Id*
endDate	Date	*[OrderItem].Order.EndDate*

Calling the Revenue Recognition Service with APEX Triggers

You can call the Revenue Recognition service with a custom APEX trigger.

For a list of input and output parameters, see Revenue Recognition Service Developer Guide.

👁 **Example:** In this example, we want to create a revenue schedule for an invoice line after its status changes to Posted. The org uses legal entities, so we included the legal entity in an input.

🛑 **Important:** Always review field validations and mapping to ensure you can trigger your process without any errors.

```
//In this example, we want to create a revenue schedule for an invoice line after its
 status changes to Posted.
//The example uses the InvoiceLine object in the billing package, where the API name
is blng__InvoiceLine__c.
//For your specific use case, replace blng__InvoiceLine__c with the desired API name
object.

trigger GenerateRevenueOnInvoiceLineActivation on blng__InvoiceLine__c (before insert)
 {

    blng__InvoiceLine__c[] newInvoiceLine = Trigger.new;
    blng__InvoiceLine__c[] oldInvoiceLine = Trigger.old;

    blng.RevenueRecognitionInput[] inputs = new List<blng.RevenueRecognitionInput>();
```

```
    Integer i = 0;
    for (blng__InvoiceLine__c newInvoiceLine : newInvoiceLine) {
        if (newInvoiceLine.blng__InvoiceLineStatus__c !=
oldInvoiceLine[i].blng__InvoiceLineStatus__c &&
            newInvoiceLine.blng__InvoiceLineStatus__c == 'Posted') {
            blng.RevenueRecognitionInput input = new blng.RevenueRecognitionInput();
            input.source = newInvoiceLine.ID;
            input.sourceFieldName = 'InvoiceLine';
            input.revenueAmount = newInvoiceLine.blng__Balance__c;
            input.startDate = newInvoiceLine.blng__StartDate__c;
            input.endDate = (Date.today()).addMonths(2);
            input.revenueRecognitionRuleId =
newInvoiceLine.blng__Product__r.blng__RevenueRecognitionRule__c;
            input.legalEntityId = newInvoiceLine.blng__LegalEntity__c;
            inputs.add(input);
        }
        i++;
    }

    if (inputs != null && inputs.size() > 0) {
        List<blng.RevenueRecognitionResponse> response =
blng.RevenueRecognition.recognizeRevenue(inputs);
    }

}
```

Revenue Recognition REST API

Use REST API to call invocable methods from the Revenue Recognition Apex class.

This object is available in API version 48.0 and later.

Supported REST HTTP Methods

URI
  actions/custom/apex/blng__RevenueRecognition

Formats
  JSON, XML

HTTP Methods
  POST

Authentication
  Bearer Token

Parameters

Input	Details
currencyIsoCode	**Type** String

Input	Details
	Description Required only for multicurrency orgs. Defines the currency used in revenue schedules made from this process builder. If not defined, in a single-currency org, the revenue recognition service uses the org's default currency.
`legalEntityID`	**Type** ID **Description** Required for orgs that use legal entities. Used to determine the correct revenue recognition treatments to use for a revenue recognition rule. ⊗ **Important:** If your org uses legal entities, and you're using revenue recognition API on an object that doesn't have a Legal Entity field, you must create a custom legal entity field and enter its value on your own or look up the value from a related record.
`revenueAmount`	**Type** Decimal **Description** The amount that the revenue recognition service uses to determine available, deferred, recognized, and total revenue based on revenue recognition rules. Users must ensure that the amount value they pass to revenue recognition service has the currency as the currencyIsoCode field. Required.
`revenueRecognitionRuleId`	**Type** ID **Description** The service applies this revenue recognition rule, then applies the rule's appropriate revenue recognition treatment. Required.
`source`	**Type** ID **Description** ID of the entity that the revenue recognition service evaluates for revenue recognition or forecasting. Required.
`sourceFieldName`	**Type** String **Description** This field appears on the revenue schedule and stores a lookup to the source record's ID. Required.
`startDate`	**Type** Date

Input	Details
	Description
	The date that the revenue recognition period begins. When the revenue distribution method is set to full recognition, the revenue recognition service uses the start date for full recognition and ignores the end date. Required.
endDate	**Type**
	Date
	Description
	The date when the revenue recognition period ends. Required only for monthly or daily recognition.

Response Body

Name	Type	Definition
jobId	ID	The ID of the APEX job that sent your input parameters to the revenue recognition service.
inputValidationStatus	String	A list of input parameters that encountered validation errors.
totalNumberOfInputsWithErrors	Integer	The number of input parameters that the revenue recognition service couldn't use due to errors (for example, a missing startDate, or a revenueAmount with a non-numeric character).
totalNumberOfInputsWithoutErrors	Integer	The number of input parameters that were successfully passed to the revenue recognition service.

Samples

Sample Request Body

This payload passes a payment record to the Revenue Recognition Service. The service attempts to create a revenue schedule based off the revenue amount of $12000 and the defined revenue recognition rule.

```
{
  "inputs":[
    {
      "revenueAmount":"12000",
      "currencyIsoCode":"USD",
      "legalEntityId":"a1j5A000001P2ri",
      "revenueRecognitionRuleId":"a2A0x000000jM6REAU",
      "startDate":"2019-08-29",
      "source":"a2217000000G4ecAAC",
```

```
            "sourceFieldname":"blng__Contract__c"
        }
    ]
}
```

Sample Response Body

In this case, we receive a response with one error because the sample payload didn't include a `startDate` variable.

```
[
    {
        "actionName" : "blng__RecognizeRevenue",
        "errors" : null,
        "isSuccess" : true,
        "outputValues" : {
            "jobId" : "7070x00000s3Ut9AAE",
            "totalInputWithoutErrors" : "0",
            "totalInputWithErrors" : "1",
            "inputValidationStatus" : [
                [ "Queued"],
                [ "Start date is a required field"]
            ]
        }
    }
]
```

Hosted Card Payments Lightning Component

The `force:cardPayment` (Hosted Card Payments) component provides a user interface in a hosted payment page that uses Salesforce Billing to collect card payment information, make a payment, or perform both actions. Salesforce stores and processes the customer card information under PCI-compliant standards. You can host the payment page in an Experience Cloud site Page or Lightning Page.

EDITIONS

Available in: Salesforce Billing Winter '20 and later

To use this component, you must also have Salesforce Billing installed in your org. For more information, see Install Salesforce Billing in Salesforce Help.

The user interface contains a form that takes customer credit card and payment details, such as cardholder name and address, and a button for submitting them. You can customize the card component to hide fields, make fields required, and change the labels of fields and buttons. These customizations affect what admins see when they add the component to their Experience Cloud site or Lightning Pages, and what end customers see when they make a payment using your component. For example, in this component, we've defined the following customizations.

- The Expiration Month is required

- The Expiration Year is required

- The Email field is hidden

- The first address line has a label of "Street"

- The Pay Button's label has been changed to Save Card.

```
<force:CardPayment
    paymentProvider = "{!v.paymentProvider}"
    transactionType = "{!v.transactionType}"
```

```
            transactionParams = "{!v.transactionParams}"
            expiryMonthRequired = "true"
            expiryYearRequired = "true"
            hideEmail = "true"
            addressLine1Label = "Street"
            payButtonLabel="Save Card"/>
```

Once you've configured your component, you can embed it in a wrapper component that's available in Experience Builder. Here's a sample wrapper component with the cardPayment component embedded.

```
<aura:component>

    <aura:handler name="paymentTransactionCompleted"
              event="force:paymentTransactionCompleted"
              action="{!c.handleTransactionResponse}"/>

    <aura:attribute name="accountId" type="String" default="abcd"
        description="Account to which the transaction details are related"/>

    <aura:attribute name="gatewayId" type="String" default="1234"
        description="Gateway to be used for the transaction"/>

    <aura:attribute name="amount" type="Decimal" default="10"
        description="amount to be charged for the transaction"/>

    <aura:attribute name="paymentProvider" type="String" default='SalesforceBilling'
        description="Payments provider for this component"/>

    <aura:attribute name="transactionType" type="String" default='PaymentSale'
        description="Transaction to be done after submitting the button"/>

    <aura:attribute name="transactionParams" type="Object"
        description="Contains details about account id and gateway id to use"/>

    <aura:handler name="init" value="{!this}" action="{!c.doInit}"/>

    <force:cardPayment
            paymentProvider = "{!v.paymentProvider}"
            transactionType = "{!v.transactionType}"
            transactionParams = "{!v.transactionParams}"
            expiryMonthRequired = "true"
            expiryYearRequired = "true"
            hideEmail = "true"
            addressLine1Label = "Street"
            payButtonLabel="Save Card"/>
</aura:component>
```

You must handle the `force:paymentTransactionCompleted` event, which is fired after the transaction is completed.

```
    handleTransactionResponse : function(cmp, event, helper) {
      var response = event.getParam("response");
      if(response.isSuccess) {
```

```
                alert("isSuccess : " + response.isSuccess);
        }
    }
```

You can also add attributes to a design resource so that your admins have the option of hiding fields, making fields required, and changing labels. When your design resource is active, admins can select the payment component in Experience Builder to show a properties pane with checkboxes or text boxes for each of the attributes. Save your design resource as a `.design` file in the same directory as your wrapper component. For example, if you want to present your five customized attributes for your admins to select or deselect from the properties pane, add the following Design section.

```
<design:component>
    <design:attribute name="expiryMonthRequired" label="Expiry month is required"/>
    <design:attribute name="expiryYearRequired" label="Expiry year is required"/>

    <design:attribute name="hideSaveForFuture" label="Hide save for future"/>
    <design:attribute name="hideBillingAddressSection" label="Hide billing address
section"/>

    <design:attribute name="payButtonLabel" label="Pay button label"/>
</design:component>
```

Response Formats

Success

During a successful response, `isSuccess` becomes true. If `transactionType` had a value of PaymentSale, `salesforceResponse` response contains paymentId, paymentMethodId, and paymentTransactionId. If `transactionType` had a value of SavePaymentCard, `salesforceResponse` contains only paymentMethodId. The attributes of `gatewayResponse` varies based on the gateway. This example shows a Success response for a PaymentSale transaction and all possible attributes for `gatewayResponse`.

```
        {
         "isSuccess" : true,
         "salesforceResponse" : {
          "paymentId" : "paymentId",
          "paymentMethodId" : "paymentMethodId",
          "paymentTransactionId" : "paymentTransactionId"
         },
         "gatewayResponse": {
          "message": "mes",
          "responseCodeMessage": "rcm",
          "responseMessage": "rm",
          "responseCode": "rc",
          "responseStatus": "rs"
         }
        }
```

Failure in the gateway

```
        {
         "isSuccess" : false,
```

```
      "salesforceResponse" : {
        "paymentTransactionId" : "paymentTransactionId"
      },
      "gatewayResponse": {
        "message": "mes",
        "responseCodeMessage": "rcm",
        "responseMessage": "rm",
        "responseCode": "rc",
        "responseStatus": "rs"
      }
    }
```

Failure Internally

```
    {
     "isSuccess": false,
     "errorDetails": {
       "message": "message"
      }
    }
```

Security Considerations and PCI Compliance

When end users are done entering their credit card information, they click the payment button so that the component sends the information to the payment gateway. Based on how you've configured your component, the gateway either tokenizes the card information and sends it back to the component, or the gateway tokenizes the card information, processes it for payment, and sends the results to the component.

Salesforce Billing uses security guardrails to ensure that end users aren't charged more than once if they accidentally click the Pay Now button multiple times, or in the event of fraudulent activity. After the component sends a request to the payment gateway, it generates a nonce associated with that specific request. Salesforce Billing won't send another request unless the page is refreshed and the Hosted Card Payments component is re-rendered. The component validates the nonce before sending payment information to the gateway. If the nonce is valid, the component removes the nonce and sends the request to the gateway.

You can also use Javascript controllers to manage error handling and response handling. Here's a sample Javascript controller that uses the handler that we introduced earlier.

```
({
    doInit : function(component, event, helper) {
        var transactionParams = {};
        transactionParams.accountId = component.get("v.accountId");
        transactionParams.gatewayId = component.get("v.gatewayId");
        transactionParams.amount = component.get("v.amount");

        component.set("v.transactionParams", transactionParams);
    },

    handleTransactionResponse : function(cmp, event, helper) {
        var response = event.getParam("response");
        if(response.isSuccess) {
```

```
            alert("isSuccess : " + response.isSuccess);
        }
    }
})
```

Attributes for the Card Hosted Payments Component

Review the attributes for the Card Hosted Payments component.

Guidelines for the Hosted Card Payments Lightning Component

When you set up the wrapper and design for your hosted payment component, consider important guidelines.

HostedPaymentPageTransactionAPI Class

Global API Apex class for Salesforce Billing. Contains methods that customers can call to save a credit card with tokenization or make a transaction using a new credit card.

Attributes for the Card Hosted Payments Component

Review the attributes for the Card Hosted Payments component.

All attributes have a Default value of False and Access value of Global.

<table>
<tr><td>EDITIONS</td></tr>
<tr><td>Available in: Salesforce Billing Winter '20 and later</td></tr>
</table>

Name	Type	Description
paymentProvider	String	Defines the source for processing payments. The cardPaymentMethod component requires that this attribute has a value of SalesforceBilling.
transactionType	String	The action that the component performs when users select the component's button. Accepts values of SavePaymentCard or PaymentSale.
transactionParams	Object	Accepts values of amount, accountId, and gatewayId.
emailRequired	Boolean	Specifies whether the Email field is required.
cardHolderNameRequired	Boolean	Specifies whether the Card Holder Name field is required.
cardTypeRequired	Boolean	Specifies whether the Card Type field is required.
expiryMonthRequired	Boolean	Specifies whether the Expiration Month field is required.

Name	Type	Description
expiryYearRequired	Boolean	Specifies whether the Expiration Year field is required.
cvvRequired	Boolean	Specifies whether the CVV field is required.
firstNameRequired	Boolean	Specifies whether the First Name field is required.
lastNameRequired	Boolean	Specifies whether the Last Name field is required.
addressLine1Required	Boolean	Specifies whether the Address Line 1 field is required.
addressLine2Required	Boolean	Specifies whether the Address Line 2 field is required.
cityRequired	Boolean	Specifies whether the City field is required.
stateRequired	Boolean	Specifies whether the State field is required.
countryRequired	Boolean	Specifies whether the Country field is required.
postalCodeRequired	Boolean	Specifies whether the Postal Code field is required.
hideCardHolderName	Boolean	Specifies whether the Cardholder Name field is hidden.
hideCardType	Boolean	Specifies whether the Card Type field is hidden.
hideExpiryMonth	Boolean	Specifies whether the Expiration Month field is hidden.
hideExpiryYear	Boolean	Specifies whether the Expiration Year field is hidden.
hideCvv	Boolean	Specifies whether the CVV field is hidden.
hideSaveForFuture	Boolean	Specifies whether the Save for Future field is hidden.
hideAutoPay	Boolean	Specifies whether the AutoPay field is hidden.
hideFirstName	Boolean	Specifies whether the First Name field is hidden.
hideLastName	Boolean	Specifies whether the Last Name field is hidden.
hideEmail	Boolean	Specifies whether the Email field is hidden.

Name	Type	Description
hideAddressLine1	Boolean	Specifies whether the Address Line 1 field is hidden.
hideAddressLine2	Boolean	Specifies whether the Address Line 2 field is hidden.
hideCity	Boolean	Specifies whether the City field is hidden.
hideState	Boolean	Specifies whether the State field is hidden.
hideCountry	Boolean	Specifies whether the Country field is hidden.
hidePostalCode	Boolean	Specifies whether the Postal Code field is hidden.
creditCardInformationLabel	Boolean	Label for the Credit Card Information section.
addressInformationLabel	Boolean	Label for the Address Information section.
payButtonLabel	String	Label for the Pay button.
cardHolderNameLabel	String	Label for the cardHolderName field.
cardTypeLabel	String	Label for the cardType field.
cardNumberLabel	String	Label for the cardNumber field.
expiryMonthLabel	String	Label for the expiryMonth field.
expiryYearLabel	String	Label for the expiryYear field.
cvvLabel	String	Label for the CVV field.
saveForFutureLabel	String	Label for the saveForFuture field.
autoPayLabel	String	Label for the AutoPay field.
firstNameLabel	String	Label for the firstName field.
lastNameLAbel	String	Label for the lastName field.
emailLabel	String	Label for the email field.
addressLine1Label	String	Label for the addressLine1 field.
addressLine2Label	String	Label for the addressLine2 field.
cityLabel	String	Label for the city field.
stateLabel	String	Label for the state field.
countryLabel	String	Label for the country field.
postalCodeLabel	String	Label for the postalCode field.

Name	Type	Description
hideAddressSection	String	Specifies whether the address section is hidden.
clearFormData	String	Clears the component's form data.

Guidelines for the Hosted Card Payments Lightning Component

When you set up the wrapper and design for your hosted payment component, consider important guidelines.

- Customers can pay only against an account.

- In multicurrency orgs, customers can make payments only in the account's currency. In single-currency orgs, the component uses the org's currency. When handling currency, the component retrieves SavePaymentCard or PaymentSale's currency from the customer's account. It won't accept CurrencyIsoCode as an input parameter.

- Component field positions can't be changed.

- The component doesn't support custom fields.

- To prevent fraudulent activity, we recommend developing guardrails around the users that can access your payments page. We also recommend that your payments page is not the first page in your Experience Cloud site or Lightning page.

HostedPaymentPageTransactionAPI Class

Global API Apex class for Salesforce Billing. Contains methods that customers can call to save a credit card with tokenization or make a transaction using a new credit card.

Namespace

blng

Usage

The `HostedPaymentPageTransactionAPI` class contains the `chargeTransaction` method, which allows users to make a transaction using a new credit card, and the `savePaymentMethod` method, which allows users to save a credit card with tokenization.

Inputs

Table 7: Address Inputs

Name	Type	Description	Required or Optional	Available Version
addressLine1	String	First line in the address of the user making the payment.	Optional	48.0
addressLine2	String	Second line in the address of the user making the payment.	Optional	48.0

Name	Type	Description	Required or Optional	Available Version
city	String	City	Optional	48.0
country	String	Country	Optional	48.0
postalCode	String	Postal code	Optional	48.0
state	String	State	Optional	48.0

Table 8: CardPaymentMethod Inputs

Name	Type	Description	Required or Optional	Available Version
cardHolderName	String	Full name of the card holder.	Required	48.0
cardNumber	String	Number of the credit card.	Required	48.0
cardType	String	Card network type. Valid values are: • AmericanExpress • Discover • MasterCard • Visa	Required	48.0
cvv	String	CVV	Required	48.0
expiryMonth	Integer	Card expiration month.	Required	48.0
expiryYear	Integer	Card expiration year.	Required	48.0

Table 9: PaymentMethod Inputs

Name	Type	Description	Required or Optional	Available Version
address	String	Address of the payment method holder.	Required	48.0
autopay	Boolean	Defines whether Salesforce Billing saves a record of the payment method to the customer's account for use in future payment runs. One account can have only one payment method with autopay enabled.	Optional	48.0
email	String	Email of the payment method holder.	Optional	48.0
firstName	String	First name of the payment method holder.	Required	48.0
lastName	String	Last name of the payment method holder.	Required	48.0
cardPaymentMethod	cardPaymentMethod	Card details for the payment method.	Required	48.0

Name	Type	Description	Required or Optional	Available Version
`paymentType`	String	Type of payment method. Valid values are: • `Credit Card`	Required	48.0
`saveForFuture`	Boolean	Indicates whether Salesforce saves a record of the payment method for future use.	Optional	48.0

HostedPaymentPageTransactionAPI Methods

HostedPaymentPageTransactionAPI Methods

The following are methods for `HostedPaymentPageTransactionAPI`.

chargeTransaction(chargeRequest)

Use this method to charge the desired amount from the buyer's card.

savePaymentMethod(paymentMethodRequest)

Use this method to save a payment method. Currently supports saving only credit cards.

chargeTransaction(chargeRequest)

Use this method to charge the desired amount from the buyer's card.

Signature

```
global static String chargeTransaction(String chargeRequest)
```

Parameters

chargeRequest

Type: String

Represents a charge request made to the payment gateway.

Return Value

Type: String

Represents a charge transaction response.

Example

Example format for a charge request input.

```
{
"accountId": "0018A00000QcPDpQAN",
"gatewayId": "a1x8A000000OZG2",
"amount": 10,
```

```
    "paymentMethod": {
        "firstName": "Muneer Ahmed",
        "lastName": "Shaik",
        "email": "mas@sf.com",
        "saveForFuture": true,
        "autoPay": true,
        "paymentType": "Credit Card",
        "address": {
            "addressLine1": "Address 1",
            "addressLine2": "Address 2",
            "city": "SF",
            "state": "CA",
            "country": "United States",
            "postalCode": "41111"
        },
        "cardPaymentMethod": {
            "cardHolderName": "Muneer Ahmed Shaik",
            "cardType": "Visa",
            "cardNumber": "4111111111111111",
            "cvv": "111",
            "expiryMonth": "2",
            "expiryYear": "2021"
        }
    }
}
```

savePaymentMethod(paymentMethodRequest)

Use this method to save a payment method. Currently supports saving only credit cards.

Signature

```
global static String savePaymentMethod(String paymentMethodRequest)
```

Parameters

paymentMethodRequest
    Type: String

    Represents a request to save a payment method.

Return Value

Type: String

Represents a save payment method response.

Example

Example format for a save payment method request input.

```
{
  "accountId": "0018A00000QcPDpQAN",
```

```
        "gatewayId": "a1x8A000000OZG2",
        "paymentMethod": {
            "firstName": "Muneer Ahmed",
            "lastName": "Shaik",
            "email": "mas@sf.com",
            "paymentType": "Credit Card",
            "address": {
                "addressLine1": "Address 1",
                "addressLine2": "Address 2",
                "city": "SF",
                "state": "CA",
                "country": "United States",
                "postalCode": "41111"
            },
            "cardPaymentMethod": {
                "cardHolderName": "Muneer Ahmed Shaik",
                "cardType": "Visa",
                "cardNumber": "4111111111111111",
                "cvv": "111",
                "expiryMonth": "2",
                "expiryYear": "2021"
            }
        }
    }
```

REST API for Converting Invoice Lines with Negative Balances

Use the REST API service `blng__NegativeInvoiceToCreditNoteAction` to evaluate invoices in bulk and create credit notes for each invoice containing invoice lines with negative balances.

This object is available in API version 48.0 and later.

Supported REST HTTP Methods

URI
   `/services/data/`***XX.X***`/actions/custom/apex/blng__NegativeInvoiceToCreditNoteAction`

Formats
   JSON, XML

HTTP Methods
   POST

Authentication
   `Authorization: Bearer token`

Parameters

Parameter	Description
invoiceId	**Type** ID **Description** The REST service evaluates this invoice for invoice lines with a negative balance. For each invoice, it creates a credit note containing one credit note line for each negative invoice line, unless an error occurs during credit note line creation. If even one of the `-ve` invoice line conversions errors out, all of them fail. No partial conversion can occur. Each credit note line has a balance that's the positive equivalent of the corresponding negative invoice line. The credit note lines are unallocated, so users must manually allocate them as needed.

Response Body

Parameter	Description
invoiceId	**Type** String **Description** The invoice that the REST service evaluated.
isSuccess	**Type** Boolean **Description** Indicates whether the invoice was successfully evaluated (`True`) or could not be evaluated (`False`).
errorMessage	**Type** String **Description** Error message indicating why the REST service couldn't convert any negative invoice lines on the provided invoice. For example, the invoice did not have any negative invoice lines.

Samples

Sample Request Body

This payload passes two invoice IDs to the negative invoice line conversion service.

```
{
"inputs":[
```

491

```
        {
            "invoiceId":"a1oS0000001NEixIAG"
        },
        {
            "invoiceId":"a1oS0000001NEixIAE"
        }
    ]
}
```

Sample Response Body

In this case, we receive a response with one error because the sample payload contained an invoice that didn't have any negative invoice lines.

```
[
    {
        "actionName" : "blng__NegativeInvoiceToCreditNoteAction",
        "errors" : null,
        "isSuccess" : true,
        "outputValues" : {
            "errorMessage" : "Null"
            "invoiceId" : "a1oS0000001NEixIAG"
            "isSuccess" : "true"
        }
    }

    {
        "actionName" : "blng__NegativeInvoiceToCreditNoteAction",
        "errors" : null,
        "isSuccess" : true,
        "outputValues" : {
            "errorMessage" : "The invoice does not have any negative invoice lines."
            "invoiceId" : "a1oS0000001NEixIAE"
            "isSuccess" : "false"
        }
    }

]
```

Set Up Standalone Orders with API

Manage your orders and order products in Salesforce and Salesforce Billing without first creating a CPQ quote. We call this type of order a standalone order. They're useful if you manage your quotes and opportunities in an external platform but plan to use Salesforce Billing for order management and billing.

Our standalone orders documentation contains required fields and setup instructions unique to standalone orders and order products. For concepts shared between standard orders and standalone orders, we provide links to the original CPQ and Billing topics.

You still need the Salesforce CPQ package installed to create standalone orders and order products. Although it doesn't use any Salesforce CPQ features or objects, Salesforce Billing runs internal validations against the CPQ package for several features and workflows.

Create your objects by executing the following series of REST API requests in a single POST request. All object creation requests use the following endpoint, replacing VERSION with your required API version, such as v48.0 or v49.0:

```
/services/data/v48.0/composite
```

Your JSON request must contain a compositeRequest class that defines both the order and the order item. Here's a generic example with null values for the minimum required fields. Configurations for certain billing and order product features may require additional fields and entities.

👁 Example:

```
{
"allOrNone" : true,
"compositeRequest" : [{
  "method" : "POST",
  "url" : "/services/data/VERSION/sobjects/order",
  "referenceId" : "refOrder",
  "body" : {
      "Status" : "Draft" ,
      "EffectiveDate" : ""
      "Pricebook2Id" : "",
      "AccountId" : ""}
  },{
  "method" : "POST",
  "url" : "/services/data/VERSION/sobjects/OrderItem",
  "referenceId" : "",
  "body" : {
    "Quantity" : "",
    "Unitprice" : "",
    "ServiceDate" : "",
    "EndDate" : "",
    "SBQQ__ChargeType__c" : "",
    "blng__BillableUnitPrice__c" : "",
    "SBQQ__OrderedQuantity__c" : "",
    "SBQQ__Status__c" : "",
    "PricebookEntryId" : "",
    "orderId" : ""
    }
  }]
}
```

Required Configurations for Types of Standalone Order Products

Standalone orders require you to provide all the fields that would otherwise come from a CPQ quote. We provide a collection of topics showing all the required fields and their values for standalone order products used in different types of billing features. Certain fields require a specific value, while some are required but vary based on your specific configuration.

Required Configurations for Types of Standalone Order Products

Standalone orders require you to provide all the fields that would otherwise come from a CPQ quote. We provide a collection of topics showing all the required fields and their values for standalone order products used in different types of billing features. Certain fields require a specific value, while some are required but vary based on your specific configuration.

EDITIONS

Available in: Salesforce
Billing Winter '21 and later

Requirements for Standalone One-Time Order Products

Create a standalone one-time order product through API in Salesforce Lightning or Salesforce Classic.

Requirements for Standalone Termed Recurring Order Products

Create a termed recurring order product through API in Salesforce Lightning or Salesforce Classic.

Requirements for Standalone Evergreen Order Products

Create a monthly evergreen order product through API in Salesforce Lightning or Salesforce Classic.

Requirements for Standalone Invoice Plan Order Products

Create a standalone invoice plan order product through API in Salesforce Lightning or Salesforce Classic.

Requirements for Standalone Amendment Order Products

Create a standalone amendment order product through API in Salesforce Lightning or Salesforce Classic

Requirements for Standalone Cancellation Order Products

Create a standalone cancellation order product through API in Salesforce Lightning or Salesforce Classic.

Requirements for Standalone Order Products with Consumption Schedules

Create a monthly order product with an order product consumption schedule and consumption rates through API in Salesforce Lightning or Salesforce Classic.

Requirements for Standalone Order Products with Price Schedules

Create a standalone order product with price schedules through API in Salesforce Lightning or Salesforce Classic.

Requirements for Standalone One-Time Order Products

Create a standalone one-time order product through API in Salesforce Lightning or Salesforce Classic.

EDITIONS

Available in: Salesforce Billing Winter '21 and later

Required Fields

Table 10: Product

Field	Value
Billing Rule	[Required]
Charge Type	One-Time
Revenue Recognition Rule	[Required]
Tax Rule	[Required]

Table 11: Order

Field	Value
Account	[Required]
Effective Date	[Required]
Price Book	[Required]
Status	Draft

Field	Value
Billable Unit Price	Required for invoice runs or Bill Now to pick up the order product for invoicing. We recommend calculating a value based on the formula described in Recurring Billing and Billable Unit Price.
Billing Rule	[Get value from product]
Billing Type	[Required]
Charge Type	One-Time
Charge Type	[Required]
End Date	[Required]
Order	[Required]
Ordered Quantity	[Required]
Price Book Entry	[Required]
Quantity	[Required]
Reference ID	[Required]
Revenue Recognition Rule	[Get value from product]
Service Date	[Required]
Status	Draft
Tax Rule	[Get value from product]

👁 Example:

```
{
"allOrNone" : true,
"compositeRequest" : [{
  "method" : "POST",
  "url" : "/services/data/v48.0/sobjects/order",
  "referenceId" : "refOrder",
  "body" : {
      "Status" : "Draft" ,
      "EffectiveDate" : "2020-06-16",
      "Pricebook2Id" : "Pricebook records ID",
      "AccountId" : "Account records ID"}
},{
  "method" : "POST",
  "url" : "/services/data/v48.0/sobjects/OrderItem",
  "referenceId" : "refOrderItem",
  "body" : {
    "Quantity" : "1",
    "Unitprice" : "10",
    "ServiceDate" : "2020-06-16",
    "EndDate" : "2020-06-16",
    "SBQQ__ChargeType__c" : "One-Time",
```

```
      "blng__BillableUnitPrice__c" : "10",
      "SBQQ__OrderedQuantity__c" : "1",
      "SBQQ__Status__c" : "Draft",
      "PricebookEntryId" : "PricebookEntry records ID",
      "orderId" : "@{refOrder.id}"
      }
  }]
}
```

Requirements for Standalone Termed Recurring Order Products

Create a termed recurring order product through API in Salesforce Lightning or Salesforce Classic.

Required Fields

Table 12: Product

Field	Value
Billing Frequency	Monthly, Quarterly, Semiannual, or Annual
Billing Rule	[Required]
Charge Type	Recurring
Revenue Recognition Rule	[Required]
Tax Rule	[Required]

Table 13: Order

Field	Value
Account	[Required]
Effective Date	[Required]
Price Book	[Required]
Status	Draft

Table 14: Order Product

Field	Value
Billable Unit Price	Required for invoice runs or Bill Now to pick up the order product for invoicing. We recommend calculating a value based on the formula described in Recurring Billing and Billable Unit Price.
Billing Frequency	Monthly
Billing Rule	Get value from product
Billing Type	[Required]

Field	Value
Charge Type	Recurring
Charge Type	[Required]
Default Subscription Term	[Required]
End Date	[Required]
Order	[Required]
Ordered Quantity	[Required]
Price Book Entry	[Required]
Quantity	[Required]
Reference ID	[Required]
Revenue Recognition Rule	Get value from product
Service Date	[Required]
Status	Draft
Tax Rule	Get value from product

Example:

```
{
"allOrNone" : true,
"compositeRequest" : [{
  "method" : "POST",
  "url" : "/services/data/v48.0/sobjects/order",
  "referenceId" : "refOrder",
  "body" : {
      "Status" : "Draft" ,
      "EffectiveDate" : "2020-06-16",
      "Pricebook2Id" : "Price records ID",
      "AccountId" : "Account records ID"}
  },{
  "method" : "POST",
  "url" : "/services/data/v48.0/sobjects/OrderItem",
  "referenceId" : "refOrderItem",
  "body" : {
    "Quantity" : "1",
    "Unitprice" : "1200",
    "ServiceDate" : "2020-06-16",
    "EndDate" : "2021-06-15",
    "SBQQ__ChargeType__c" : "Recurring",
    "SBQQ__BillingType__c" : "Advance",
    "SBQQ__BillingFrequency__c" : "Monthly",
    "blng__BillableUnitPrice__c" : "100",
    "SBQQ__OrderedQuantity__c" : "1",
    "SBQQ__DefaultSubscriptionTerm__c" : "1",
```

```
    "SBQQ__Status__c" : "Draft",
    "SBQQ__ContractAction__c" : "New"
    "PricebookEntryId" : "PricebookEntry records ID",
    "orderId" : "@{refOrder.id}"
    }
  }]
}
```

Requirements for Standalone Evergreen Order Products

Create a monthly evergreen order product through API in Salesforce Lightning or Salesforce Classic.

Required Fields

Table 15: Product

Field	Value
Billing Frequency	Monthly, Quarterly, Semiannual, or Annual
Billing Rule	[Required]
Charge Type	Recurring
Revenue Recognition Rule	[Required]
Tax Rule	[Required]

Table 16: Order

Field	Value
Account	[Required]
Effective Date	[Required]
Price Book	[Required]
Status	Draft

Field	Value
Billable Unit Price	Required for invoice runs or Bill Now to pick up the order product for invoicing. We recommend calculating a value based on the formula described in Recurring Billing and Billable Unit Price.
Billing Frequency	Monthly
Billing Rule	Get value from product
Billing Type	[Required]
Charge Type	Recurring

Field	Value
Charge Type	[Required]
End Date	[Required]
Order	[Required]
Ordered Quantity	[Required]
Price Book Entry	[Required]
Product Subscription Type	Evergreen
Quantity	[Required]
Reference ID	[Required]
Revenue Recognition Rule	Get value from product
Service Date	[Required]
Status	Draft
Subscription Type	Evergreen
Tax Rule	Get value from product

Example:

```
{
"allOrNone" : true,
"compositeRequest" : [{
  "method" : "POST",
  "url" : "/services/data/v48.0/sobjects/order",
  "referenceId" : "refOrder",
  "body" : {
      "Status" : "Draft" ,
      "EffectiveDate" : "2020-06-16",
      "Pricebook2Id" : "Price records ID",
      "AccountId" : "Account records ID"}
  },{
  "method" : "POST",
  "url" : "/services/data/v48.0/sobjects/OrderItem",
  "referenceId" : "refOrderItem",
  "body" : {
    "Quantity" : "1",
    "Unitprice" : "50",
    "ServiceDate" : "2020-06-16",
    "SBQQ__ProductSubscriptionType__c" : "*Evergreen*",
    "SBQQ__SubscriptionType__c" : "*Evergreen*",
    "SBQQ__ChargeType__c" : "Recurring",
    "SBQQ__BillingType__c" : "Advance",
    "SBQQ__BillingFrequency__c" : "Monthly",
    "blng__BillableUnitPrice__c" : "50",
    "SBQQ__OrderedQuantity__c" : "1",
```

```
    "SBQQ__DefaultSubscriptionTerm__c" : "1",
    "SBQQ__Status__c" : "Draft",
    "SBQQ__ContractAction__c" : "New"
    "PricebookEntryId" : "PricebookEntry records ID",
    "orderId" : "@{refOrder.id}"
    }
  }]
}
```

Requirements for Standalone Invoice Plan Order Products

Create a standalone invoice plan order product through API in Salesforce Lightning or Salesforce
Classic.

Salesforce Billing invoices standalone invoice plan order products only when the order product has
a billing treatment with a New Order Invoice Plan value. You can configure your billing rules and
legal entities so that Salesforce Billing assigns your order product a billing treatment. Or, you can
create your order product, and then give it a billing treatment on your own.

Required Fields

Table 17: Product

Field	Value
Billing Frequency	Invoice Plan
Billing Rule	[Required]
Charge Type	Recurring
Revenue Recognition Rule	[Required]
Tax Rule	[Required]

Table 18: Order

Field	Value
Account	[Required]
Effective Date	[Required]
Price Book	[Required]
Status	Draft

Table 19: Order Product

Field	Value
Billable Unit Price	When Enable Billing Order Validations is active, Salesforce Billing sets this value to zero for order products created with a billing frequency of Invoice Plan.
Billing Frequency	Invoice Plan
Billing Rule	[Required]
Billing Rule	Get value from product
Billing Treatment	[Must be a billing treatment with an active invoice plan]
Charge Type	Recurring
End Date	[Required]
Price Book Entry	[Required]
Quantity	[Required]
Revenue Recognition Rule	[Required]
Revenue Recognition Rule	[Get value from product]
Service Date	[Required]
Status	Draft
Tax Rule	[Required]
Tax Rule	[Get value from product]
Unit Price	[Required]

👁 Example:

```
{
"allOrNone" : true,
"compositeRequest" : [{
  "method" : "POST",
  "url" : "/services/data/v48.0/sobjects/order",
  "referenceId" : "refOrder",
  "body" : {
      "Status" : "Draft" ,
      "EffectiveDate" : "2020-06-16",
      "Pricebook2Id" : "Price records ID",
      "AccountId" : "Account records ID"}
  },{
  "method" : "POST",
  "url" : "/services/data/v48.0/sobjects/OrderItem",
  "referenceId" : "refOrderItem",
  "body" : {
    "Quantity" : "1",
    "Unitprice" : "50",
```

```
    "ServiceDate" : "2020-06-16",
    "EndDate" : "2021-06-15",
    "BillingTreatment__c" : "BillingTreatment records ID",
    "SBQQ__ChargeType__c" : "Recurring",
    "SBQQ__BillingType__c" : "Advance",
    "SBQQ__BillingFrequency__c" : "Monthly",
    "blng__BillableUnitPrice__c" : "50",
    "SBQQ__OrderedQuantity__c" : "1",
    "SBQQ__DefaultSubscriptionTerm__c" : "1",
    "SBQQ__Status__c" : "Draft",
    "SBQQ__ContractAction__c" : "New"
    "PricebookEntryId" : "PricebookEntry records ID",
    "orderId" : "@{refOrder.id}"
    }
  }]
}
```

Requirements for Standalone Amendment Order Products

Create a standalone amendment order product through API in Salesforce Lightning or Salesforce
Classic

Make sure to provide a value for the revised order product ID. The Revised Order Product field shows
the ID of the original order product used before any amendments were made. Remember, an order
product can be related to multiple amendment order products.

Required Fields

Table 20: Product

Field	Value
Billing Frequency	Invoice Plan
Billing Rule	[Required]
Charge Type	Recurring
Revenue Recognition Rule	[Required]
Tax Rule	[Required]

Table 21: Order

Field	Value
Account	[Required]
Effective Date	[Required]
Price Book	[Required]
Status	Draft

Table 22: Order Product

Field	Value
Billable Unit Price	Required for invoice runs or Bill Now to pick up the order product for invoicing. We recommend calculating a value based on the formula described in Recurring Billing and Billable Unit Price.
Billing Frequency	[Required]
Billing Rule	[Get value from product]
Billing Type	[Required]
Charge Type	[Required]
Charge Type	[Required]
Default Subscription Term	[Required]
End Date	[Required]
Order	[Required]
Ordered Quantity	[Required]
Price Book Entry	[Required]
Quantity	[Required]
Reference ID	[Required]
Revenue Recognition Rule	[Get value from product]
Revised Order Product	[Required]
Service Date	[Required]
Status	Draft
Tax Rule	[Get value from product]

Requirements for Standalone Cancellation Order Products

Create a standalone cancellation order product through API in Salesforce Lightning or Salesforce Classic.

Salesforce Billing calculates canceled billings for a canceled standalone order product under the following settings.

EDITIONS

Available in: Salesforce Billing Winter '21 and later

- The cancellation order product has a Contract Action value of Cancel.
- The cancellation order product and related amendment order products all have a Revised Order Product field with the ID of the original order product used before any amendments or cancellations.
- The cancellation order products and related amendment order products all have the same terminated date. The Terminated Date must be the last field you populate on each order product before you activate the cancellation order product.

📝 Note: You can cancel standalone order products under both legacy cancellation and LIFO cancellation.

Section Title

Table 23: Product

Field	Value
Billing Frequency	Must match the value of the Revised Order Product field
Billing Rule	[Required]
Charge Type	Recurring
Revenue Recognition Rule	[Required]
Tax Rule	[Required]

Table 24: Order

Field	Value
Account	[Required]
Effective Date	[Required]
Price Book	[Required]
Status	Draft

Table 25: Order Product

Field	Value
Billing Frequency	Must match the value of the Revised Order Product field
Billing Rule	[Get value from product]
Billing Type	[Required]
Billable Unit Price	Required for invoice runs or Bill Now to pick up the order product for invoicing. We recommend calculating a value based on the formula described in Recurring Billing and Billable Unit Price.
Charge Type	Recurring
Contract Action	Cancel
Default Subscription Term	[Required]
End Date	[Required] for termed subscriptions, omit for evergreen subscriptions
Legal Entity	Must match the value of the Revised Order Product field
Order	[Required]
Ordered Quantity	Negative amount of the Revised Order Product field
Price Book Entry	Must match the value of the Revised Order Product field

Field	Value
Product Subscription Type	[Required]
Quantity	Negative amount of the Revised Order Product field
Revenue Recognition Rule	[Get value from product]
Revised Order Product	[ID number for the original order product used before any amendments]
Service Date	[Required]
Status	Draft
Subscription Type	[Required]
Tax Rule	[Get value from product]
Unit Price	[Required]
Terminated Date	[Set to the last day the canceled order product is billed] Terminated Date and Contract Action fields must be populated in a separate DML after the order product is created, all other required fields are populated, and the order product is activated. Terminated Date and Contract Action fields must be populated after the cancellation order product is created and activated. To do so: • Populate Terminated Date on the original order product (the one that is linked in the Revised Order Product field). • On the cancellation order product, populate Terminated Date and set Contract Action to Cancel. • Save.

👁 Example:

```
{
"allOrNone" : true,
"compositeRequest" : [{
  "method" : "POST",
  "url" : "/services/data/v48.0/sobjects/order",
  "referenceId" : "refOrder",
  "body" : {
      "Status" : "Draft" ,
      "EffectiveDate" : "2020-06-16",
      "Pricebook2Id" : "Price records ID",
      "AccountId" : "Account records ID"}
},{
"method" : "POST",
"url" : "/services/data/v48.0/sobjects/OrderItem",
"referenceId" : "refOrderItem",
"body" : {
```

```
        "Quantity" : "1",
        "Unitprice" : "1200",
        "ServiceDate" : "2020-06-16",
        "EndDate" : "2021-06-15",
        "SBQQ__ChargeType__c" : "Recurring",
        "SBQQ__BillingType__c" : "Advance",
        "SBQQ__BillingFrequency__c" : "Monthly",
        "blng__BillableUnitPrice__c" : "100",
        "SBQQ__OrderedQuantity__c" : "1",
        "SBQQ__DefaultSubscriptionTerm__c" : "1",
        "SBQQ__Status__c" : "Draft",
        "PricebookEntryId" : "PricebookEntry records ID",
       "SBQQ__RevisedOrderProduct__c" : "RevisedOrderProduct(original orderproduct) records
     ID",
        "SBQQ__TerminatedDate__c" : "2020-08-15",
        "SBQQ__ContractAction__c" : "Cancel",
        "orderId" : "@{refOrder.id}"
        }
      }]
    }
```

Requirements for Standalone Order Products with Consumption Schedules

Create a monthly order product with an order product consumption schedule and consumption rates through API in Salesforce Lightning or Salesforce Classic.

You must create the order, order product, order product consumption schedule, and at least one order product consumption rate. Your order product must be related to a product with a consumption schedule.

Salesforce Billing creates usage summaries when you activate an order product with the following configuration.

EDITIONS

Available in: Salesforce Billing Winter '21 and later

- The order product is related to a product with a consumption schedule.

- The order product has an active order product consumption schedule with at least one consumption rate.

Required Fields

Table 26: Product

Field	Value
Billing Frequency	Invoice Plan
Billing Rule	[Required]
Charge Type	Recurring
Revenue Recognition Rule	[Required]
Tax Rule	[Required]

Table 27: Order

Field	Value
Account	[Required]
Effective Date	[Required]
Price Book	[Required]
Status	Draft

Table 28: Order Product

Field	Value
Billable Unit Price	Required for invoice runs or Bill Now to pick up the order product for invoicing. We recommend calculating a value based on the formula described in Recurring Billing and Billable Unit Price.
Billing Frequency	[Required]
Billing Rule	[Get value from product]
Billing Type	[Required]
Charge Type	[Required]
Charge Type	[Required]
Default Subscription Term	[Required]
End Date	[Required]
Order	[Required]
Ordered Quantity	[Required]
Price Book Entry	[Required]
Quantity	[Required]
Reference ID	[Required]
Revenue Recognition Rule	[Get value from product]
Revised Order Product	[Required]
Service Date	[Required]
Status	Draft
Tax Rule	[Get value from product]

Table 29: Consumption Schedule

Field	Value
Billing Rule	[Required]

Field	Value
Billing Term	[Required]
Billing Term Unit	[Required]
Category	[Required]
Rating Method	[Required]
Revenue Recognition Rule	[Required]
Tax Rule	[Required]
Type	[Required]

Table 30: Consumption Rates

Field	Value
Lower Bound	[Required]
Price	[Required]
Pricing Method	[Required]
Processing Order	[Required]
Upper Bound	[Optional]

Table 31: Order Product Consumption Schedule

Field	Value
Billing Rule	[Required]
Billing Term	[Required]
Billing Term Unit	[Required]
Category	[Required]
Consumption Schedule	[Optional]
Order Product	[Required]
Rating Method	[Required]
Revenue Recognition Rule	[Required]
Tax Rule	[Required]
Type	[Required]

Field	Value
Lower Bound	[Required]

Field	Value
Order Product Consumption Schedule	[Required]
Price	[Required]
Pricing Method	[Required]
Processing Order	[Required]
Upper Bound	[Optional]

👁 Example:

```
{
"allOrNone" : true,
"compositeRequest" : [{
  "method" : "POST",
  "url" : "/services/data/v48.0/sobjects/order",
  "referenceId" : "refOrder",
  "body" : {
      "Status" : "Draft" ,
      "EffectiveDate" : "2020-06-16",
      "Pricebook2Id" : "Price records ID",
      "AccountId" : "Account records ID"}
  },{
  "method" : "POST",
  "url" : "/services/data/v48.0/sobjects/OrderItem",
  "referenceId" : "refOrderItem",
  "body" : {
    "Quantity" : "1",
    "Unitprice" : "1200",
    "ServiceDate" : "2020-06-16",
    "EndDate" : "2021-06-15",
    "SBQQ__ChargeType__c" : "Recurring",
    "SBQQ__BillingType__c" : "Advance",
    "SBQQ__BillingFrequency__c" : "Monthly",
    "blng__BillableUnitPrice__c" : "100",
    "SBQQ__OrderedQuantity__c" : "1",
    "SBQQ__DefaultSubscriptionTerm__c" : "1",
    "SBQQ__Status__c" : "Draft",
    "PricebookEntryId" : "PricebookEntry records ID",
    "SBQQ__ContractAction__c" : "New",
    "orderId" : "@{refOrder.id}"
    }
  },
  {
  "method" : "POST",
  "url" : "/services/data/v48.0/sobjects/SBQQ__OrderItemConsumptionSchedule__c",
  "referenceId" : "refOrderItemConsumptionSchedule",
  "body" : {
    "SBQQ__RatingMethod__c" : "Tier",
    "SBQQ__ConsumptionSchedule__c" : "ConsumptionSchedule record ID",
    "SBQQ__BillingTermUnit__c" : "Month",
```

```
       "SBQQ__BillingTerm__c" : "1",
       "SBQQ__Type__c" : "Range/Slab",
       "SBQQ__Category__c" : "Rates",
       "blng__BillingRule__c" : "BillingRule record ID",
       "blng__RevenueRecognitionRule__c" : "RevenueRecognitionRule record ID",
       "blng__TaxRule__c" : "TaxRule record ID",
       "SBQQ__OrderItem__c" : "@{refOrderItem.id}"
       }
    },{
    "method" : "POST",
    "url" : "/services/data/v48.0/sobjects/SBQQ__OrderItemConsumptionRate__c",
    "referenceId" : "refOrderItemConsumptionRate",
    "body" : {
       "SBQQ__ProcessingOrder__c" : "1",
       "SBQQ__LowerBound__c" : "1",
       "SBQQ__UpperBound__c" : "100 or null",
       "SBQQ__Price__c" : "100",
       "SBQQ__PricingMethod__c" : "Per Unit/ Flat Fee",
       "SBQQ__OrderItemConsumptionSchedule__c" : "@{refOrderItemConsumptionSchedule.id}"

       }
    }]
}
```

Requirements for Standalone Order Products with Price Schedules

Create a standalone order product with price schedules through API in Salesforce Lightning or
Salesforce Classic.

🛑 **Important:** Salesforce CPQ doesn't automatically create price schedules for standalone order
products. You must create the price schedule and its price tiers on your own. Each price
schedule requires at least one price tier.

EDITIONS

Available in: Salesforce
Billing Winter '21 and later

Required Fields

Table 32: Product

Field	Value
Billing Frequency	Invoice Plan
Billing Rule	[Required]
Charge Type	Recurring
Revenue Recognition Rule	[Required]
Tax Rule	[Required]

Table 33: Order

Field	Value
Account	[Required]
Effective Date	[Required]
Price Book	[Required]
Status	Draft

Table 34: Order Product

Field	Value
Billing Frequency	[Required]
Charge Type	Usage
End Date	[Required]
Price Book Entry	[Required]
Price Schedule	[Required]
Quantity	[Required]
Service Date	[Required]
Unit Price	[Required]

Table 35: Price Schedule

Field	Value
Discount Unit	[Required]
Price Type	[Required]

Table 36: Price Tier

Field	Value
Lower Bound	[Required]
Optional	[Optional]
Price Model	[Required]
Price Schedule	[Required]

Salesforce Billing Apex Reference

Salesforce Billing provides the blng namespace that contains classes to manage credit applications.

InvoiceAPI Class

Global API Apex class for Salesforce Billing. Contains methods that you can call to cancel and rebill an invoice, or to credit an invoice.

Namespace

blng

Usage

The `InvoiceAPI` class contains the `creditInvoice` method, which can be used to cancel and rebill an invoice, or to credit an invoice for a specified invoice ID.

InvoiceAPI Methods

Learn more about the methods available with the InvoiceAPI class.

InvoiceAPI Methods

Learn more about the methods available with the InvoiceAPI class.

The `InvoiceAPI` class includes these methods.

creditInvoice(invoiceId, action)

Use this method to cancel or rebill an invoice, or to credit an invoice.

creditInvoice(invoiceId, action)

Use this method to cancel or rebill an invoice, or to credit an invoice.

Signature

```
global static String creditInvoice(Id invoiceId, String action)
```

Parameters

invoiceId
   Type: String

   ID of the invoice.

action
   Type: String

   Action to perform against the invoice. For example, `Credit` or `Cancel and Rebill`.

   Valid values are:

- `Credit`—Creates a credit note equal to the invoice's total balance. The credit note contains credit note lines that are allocated in full to each of the invoice lines. The invoice must be in the `Posted` status.

- `Cancel and Rebill`—

  Creates a credit note equal to the invoice's total balance. The credit note contains credit note lines that are allocated in full to each of the invoice lines.

  If the invoice is in `Posted` status, then this action creates a credit note equal to the invoice's total balance. The credit note contains credit note lines that are allocated in full to each of the invoice lines. This action also reverts the billing fields of an order product to their previous values. The status of the invoice is changed to `Rebilled`.

  If the invoice is in `Draft` status, then Salesforce Billing sets the status to `Cancelled` for the invoice and all invoice lines.

  See Cancel and Rebill an Invoice.

Return Value

Type: String

Usage

Keep these considerations in mind when using this method.

- The invoice must not have any allocations before the credit invoice action is requested.
- The associated CreditNote, CreditNoteLine, and CreditNoteAllocation records are created against the posted invoice for both `Credit` and `Cancel and Rebill` actions. The status of the invoice is changed to `Credited`.

Example

```
blng.InvoiceAPI.creditInvoice('aF15t000000GmjKCAS', 'Cancel and Rebill');
```

```
blng.InvoiceAPI.creditInvoice('aF15t000000GmjKCAS', 'Credit');
```

INDEX