

Version 2013 Release 2 Release Notes

Revision Date: March 19, 2014

PDF File Version: v30

This document summarizes the changes to NetSuite between Version 2013 Release 2 and the previous release.

These release notes are subject to change. Please make sure to review the release notes section of the Documentation Revision History for a list of the changes made in this build of the release notes.

The Version 2013 Release 2 enhancements and changes listed in this document are not available to NetSuite customers until they are upgraded to Version 2013 Release 2. Please check the version number at the bottom of your Home page if you are unsure which NetSuite version you are using.

Click links in the following table for details about product changes:

Product	Section	Summary
NetSuite	SuiteApps (Bundles) Released by NetSuite	Enable Features Page for Tax SuiteApps, International Tax Reports Enhancements, Philippine RR09-2009 Audit Files, Mexico Compliance SuiteApp, Germany GDPdU Data Export, Transaction Timeline (Beta), Recurring Billing Enhancements, File Drag & Drop (Beta), Grid Order Management, Recurring Billing Enhancements, Electronic Payments Enhancements, Software Vertical Contract Renewals, SuiteSocial, Subsidiary Navigator SuiteApp (Beta), , Electronic Payments SuiteApp: Enhanced Email Payment Notifications
	User Interface	Support for Internet Explorer 8 Ending in Version 2013 Release 2
	Administration	Security Questions Enhancements, Security Challenge Column Added to Login Audit Trail, Prevent Login with URL that Contains NetSuite Password, Default Password Expiration of 180 Days, Enhancements to Lockout Email Notifications, Recent Records Tracking for Setup and Customization Pages, Email Notification when Sandbox Refresh Complete, End of Support for Gemalto Tokens
	Customer Relationship Management (CRM)	Scriptable Email Templates (Beta), NetSuite for Outlook, Voting for NetSuite Enhancements, Enhancements to CRM Data Management, Mapping Online Customer Form Fields to Customers and Contacts, Read Receipt Support, Issue Relationship Management, Change to Case Editing, New NetSuite for iPhone
	Enterprise Resource Planning (ERP)	Improved Credit Card Security, GL Audit Numbering, Charge-Based Billing, Deferred Revenue Reclassification Enhancements, Variable Revenue Recognition Schedule Changes, Customizing Company Records, Tax Period Posting Changes, Attaching Line Item Receipts to Expense Reports, Expense Report Line Item Reference Numbers, Quantities and Rates for Expense Categories, General Shipping Enhancements, UPS Shipping Enhancements, Costed BOM Inquiry,

Product	Section	Summary
		Component Yield Preferences, New Bulk Process Status Pages, Mark All Bulk Process Work Orders, Units of Measure Validation on Transactions, Standard Cost Material Overheads, Demand Planning and Inventory Allocation, Landed Cost Allocation Per Line
	Projects	Expanded Task Scheduling for Advanced Projects, Milestone Records for Advanced Projects, Gantt Chart Enhancements, Billing Class Enhancements for Advanced Projects, Project Center,
SuiteCommerce	Site Builder	Google Checkout Retirement Impact on NetSuite Web Stores, Alternate Payment Methods for Web Store Orders
	SuiteCommerce Advanced	Sitemap Generator for SuiteCommerce Advanced, Reference Implementation Enhancements
	SuiteCommerce Platform	New Item Search API Input Parameter for Bypassing the CDN Cache, Field Set Fields for Accessing Item Quantity On Hand Per Location, New Commerce API Methods
SuiteCloud Platform	General SuiteCloud Platform	NetSuite to Retire smbXML
	SuiteBundler	Selective Update of Sandbox Bundle Objects, Restoring Bundle Definitions after Sandbox Refresh, Dissolving Sandbox Bundles
	SuiteBuilder - Customization	Advanced PDF/HTML Templates (Beta), Translation of Custom Record Instance Names, Custom Fields with Values Derived from Summary Search Results, Change to Custom Field IDs
	SuiteAnalytics (Dashboards, Reporting and Search)	Advance Notice – NetSuite to Retire ODBC Driver 5.5
	CSV Import	Support for New Record Type Imports, New Option Prevents the Import of Duplicate Records, In Japan Edition, New Customer Field Available for Mapping
	SuiteFlow (Workflow Manager)	Enhanced Workflow Logging, Enhanced Workflow Scheduling, Enhancements for Add Button Actions in Workflows, Deleting Workflow States
	SuiteScript	Enhancement to Tax Group Record Exposure, Renaming of Duplicate Record Permission, Component Yield Preferences — SuiteScript Impact, Script Field ID Change, Recalc Event Change, HTTP Request Validation Change, Script and Script Deployment Exposure, Advanced Scheduled Script Management Enhancement, nlapiYieldScript and nlapiSetRecoveryPoint Error Handling, Alternate Time Zone Support, Credential Builder API, Void Transaction Support, Script Performance Monitor SuiteApp, Updated Version of SuiteCloud IDE, Records Exposed to SuiteScript, API Additions and Enhancements, Changed Behavior when Scripting Project Tasks 2013.2 SuiteScript Records Browser

Product	Section	Summary
	SuiteTalk (Web Services) Integration	NetSuite to End Support for SuiteTalk 2009.2 and Earlier Endpoints, Newly Supported Record Types in SuiteTalk, Search Enhancement for Customized Records, Enhancements to Tax Group Record Exposure, Improved Result Reporting for Failed Asynchronous Operations, Enhancement to AsyncInitializeList, Renaming of Duplicates Task ID, Improved Error Handling for Budget Record, Enhancement for Campaign Response Integrations, New Endpoint Exposes ScriptId of Saved Searches, With New Endpoint, Integrations Referencing Custom Fields Must Be Updated, Changed Behavior when Using SuiteTalk with Project Tasks, In New Endpoint, LandedCost Complex Type Is Renamed, New Abstract Type for Custom Fields, In Japan Edition, New Field Exposed on the Customer Record, Fields Added to 2013.2 Endpoint to Support New Features, Additional Values for Enumerations, New and Modified Search Objects, 2013.2 WSDL Now Generally Available, Araxis Merge Diff File for the 2013.2 Endpoint, 2013.2 Schema Browser

SuiteApps (Bundles) Released by NetSuite

Version 2013 Release 2 includes the following enhancements to SuiteApps released by NetSuite:

Important: SuiteApps are also known as bundles.

- Enable Features Page for Tax SuiteApps
- International Tax Reports Enhancements
- Philippine RR09-2009 Audit Files
- Mexico Compliance SuiteApp
- Germany GDPdU Data Export
- Transaction Timeline (Beta)
- File Drag & Drop (Beta)
- Subsidiary Navigator SuiteApp (Beta)
- Grid Order Management
- Recurring Billing Enhancements
- Electronic Payments Enhancements
- Software Vertical Contract Renewals
- SuiteSocial

- [Electronic Payments SuiteApp: Enhanced Email Payment Notifications](#)

Enable Features Page for Tax SuiteApps

As of Version 2013 Release 2, the NetSuite SuiteApps for tax can be accessed from the Enable Features page at Setup > Company > Enable Features. On the Tax subtab, links to the following SuiteApps are displayed:

- [International Tax Reports](#)
- [Tax Audit Files](#)
- [Withholding Tax](#)

To install a tax SuiteApp, click the SuiteApp link, and then, on the SuiteApp Details page, click Install.

Before installing a tax SuiteApp, you must enable the following SuiteCloud features in Setup > Company > Enable Features > SuiteCloud:

- [Client SuiteScript](#)
- [Server SuiteScript](#)
- [Custom Records](#)

For Withholding Tax, in addition to the SuiteCloud features, you must also enable the following features:

- [Advanced Taxes](#) (in Setup > Company > Enable Features > Accounting)
- [Expand Account Lists](#) (in Setup > Accounting > Accounting Preferences > General)

For more information, see the help topic [Taxation Features Provided by SuiteApps](#) from NetSuite.

International Tax Reports Enhancements

Enhancements to International Tax Reports include the following:

- [Date and Number Formatting on VAT Reports](#)
- [Enhanced VAT Drilldown Reports](#)
- [VAT Submission History Enhancements and Flagging of Transactions in VAT Returns](#)

Note: The flagging of transactions feature is currently available only for Austria, Belgium, Czech Republic, Denmark, Germany, Ireland, and United Kingdom nexuses. The release of the VAT Reconciliation Report feature has been deferred.

Date and Number Formatting on VAT Reports

This feature enables you to configure VAT reports to display numbers and dates in the formats required by your tax agency. For example, some countries require the reporting period on the VAT form to appear as 02-2013 instead of February 2013.

In addition, the VAT report PDF and the drilldown report now display the currency symbol, according to the base currency of the company or subsidiary.

Anyone who has access to the VAT report page for a particular nexus can set up the date and number formats for that nexus. The Formatting setup page can be accessed from the VAT report page (Reports > VAT/GST > Tax Reports (International)).

Once the formats have been set up, the system applies them to VAT form, PDF, and drilldown reports of that nexus, regardless of subsidiary. However, for Excel files, the normal behavior of Microsoft Excel is to follow the regional settings of the computer, not the browser. Your computer's regional settings control how numbers, dates, and currencies are displayed in your Excel files.

If you have not set up the formats for the VAT dates and numbers, NetSuite uses the company/subsidiary preferences defined on the General Preferences page (Setup > Company > General Preferences) for non-OneWorld accounts, and on the Preferences subtab of the subsidiary record (Setup > Company > Subsidiaries) for OneWorld accounts. The Formatting setup page on the VAT form provides more formatting options, including those that are not available on the company preferences. Once the date and number formatting for a VAT report has been set up for a nexus, NetSuite overrides the company/subsidiary preferences.

For more information, see the help topic [Formatting Dates and Numbers for VAT Reports](#).

Enhanced VAT Drilldown Reports

The enhanced VAT drilldown report is now available for all the country-specific VAT forms supported by NetSuite. On VAT reports, if you want to see the detailed breakdown of an amount, you click the system-generated value in a box or field. The system displays the Detailed Report, which shows all the transactions that make up the box value. The Detailed Report enhancements enable you to do the following:

- View the entity record by clicking the customer or vendor name
- View the transaction record by clicking the transaction number
- Sort rows by column

- Select a page to view
- Resize a column
- Move a column
- Export the drilldown report to Excel

Note that exported Excel files use the default column order and column sort, even if you change them in the UI. Also, note that the normal behavior of Microsoft Excel is to follow the regional settings of the computer, not the browser. Your computer's regional settings control how numbers, dates, and currencies are displayed in Excel.

For more information, see the help topic [Viewing VAT Report Transaction Details](#).

VAT Submission History Enhancements and Flagging of Transactions in VAT Returns

Important: This feature is available for countries that have online filing capability, namely Austria, Belgium, Czech Republic, Denmark, Germany, Ireland, and the United Kingdom.

Reconciling your transactions for VAT sets a 'flag' against each transaction included in a VAT return. This flag indicates that the transaction has already been reported, and so is excluded by default from subsequent VAT returns.

When you generate a VAT report, you can export it to CSV, XML, or DAT, depending on the filing format required for the country. When you click the Export button, a popup message asks if you want to flag the transactions included in the report. If you click Yes, the transactions will be flagged, and this action cannot be undone. If you click No, the file will be exported but transactions will not be flagged.

Note: When the Group box is checked on the VAT form UI, there is no option to flag the transactions included in the consolidated report.

If you want to flag transactions for a certain tax period, it is recommended that you first close the tax period before you generate and export the VAT report.

If you try to generate a VAT report for a period that already has flagged transactions, the VAT form displays a 'Submitted' stamp for the UK, and a 'Final' stamp for other countries.

For UK VAT online filing, when you click Submit Online, the system assumes the file you are submitting is final. However, transactions included in the VAT return are flagged only after successful submission to HMRC. If the submission fails, the transactions are not flagged.

The System Notes tab of the VAT form stores a copy of each exported file and a PDF of the submitted or final VAT return. The submission history shows the following information:

- Date – Date of VAT return generation

- User – Name of the user who generated the VAT return
- Date From – Submitted period start date
- Date To – Submitted period end date
- Status link (available for the UK only) – Indicates the status of the submission (success or failure), and opens the VAT Online Submitted Periods record.
- Type (available for the Czech Republic and Ireland only) – Indicates whether the file is an original (the first submission for the period), an amendment (correction), or an additional (supplementary) return.
- File Name link (not available for the UK) – Downloads the CSV, XML, or DAT file of the submitted VAT return for the period. The file name differentiates reports for flagged and unflagged transactions. If you chose to flag the transactions, the exported file has 'Final' in the file name. If you chose not to flag the transactions, the exported file has 'Draft' in the file name.
- View & Print link (available for submitted/final reports only) – Opens a copy of the VAT form containing the submitted or final values for the period.

When you click **View & Print**, a copy of the submitted or final form is displayed. On this page, you can do the following:

- You can save and print a PDF of the submitted VAT form by clicking Print. The header of the printed form has a 'Submitted' stamp (for UK) or 'Final' stamp (for other countries). The footer shows name of the user who generated the final report, and the time and date of the report generation.
- You can view the detailed reports of transactions that were included in the VAT return by clicking the amounts in the fields or boxes.

For more information, see the help topics [Flagging Transactions Included in a VAT Return](#) and [Viewing the VAT Return Submission History](#).

Philippine RR09-2009 Audit Files

Companies doing business in the Philippines can generate and export tax audit files from NetSuite in the format defined by the Bureau of Internal Revenue (BIR) in Revenue Regulation 09-2009.

Revenue Regulation 09-2009 defines the requirements for the maintenance, retention, and submission of electronic records. Each Philippine company or subsidiary must make its audit files available if requested by the BIR.

NetSuite provides Philippine RR09-2009 audit files, which are composed of the following separate data extracts:

- General Ledger

- General Journal
- Sales Journal
- Purchase Journal
- Cash Receipts Journal
- Cash Disbursements Journal

You can choose which data extract file to generate, and which accounting period to report. The files are saved in CSV format, and can support large volumes of data.

This feature is available in NetSuite or NetSuite OneWorld accounts that have the Tax Audit Files SuiteApp and an active Philippines nexus.

For more information, see the help topic Tax Audit Files - Philippines.

Mexico Compliance SuiteApp

The Mexico Compliance SuiteApp ensures a minimum level of tax compliance for Mexican businesses by providing custom fields, validations, and other customizations required for the Mexico DIOT file. The Declaración Informativa de Operaciones con Terceros (DIOT) is a monthly declaration of all payments made to third parties, with details of the applicable value added tax.

Validations and custom fields provided by the Mexico Compliance SuiteApp include the following:

- Registro Federal de Contribuyentes (RFC) code field and validation
- Transaction tax codes validation
- Operation type field and validation

This SuiteApp is intended to be used with the Tax Audit Files SuiteApp, which provides the Mexico DIOT reporting feature, and the International Tax Reports SuiteApp, which provides the tax code properties used for Mexico tax codes.

For more information, see the help topic Mexico Tax Topics.

Germany GDPdU Data Export

The Principles of Data Access and Verification of Digital Documents (GDPdU) standard defines the rules around the access, storage, and verification of financial data and the obligation of German taxpayers in case of a tax audit.

Users in Germany can generate and export tax audit files in NetSuite that comply with Finanzamt-mandated rules and with the format defined by the GDPdU standard.

NetSuite provides Germany GDPdU Data Export composed of the following files:

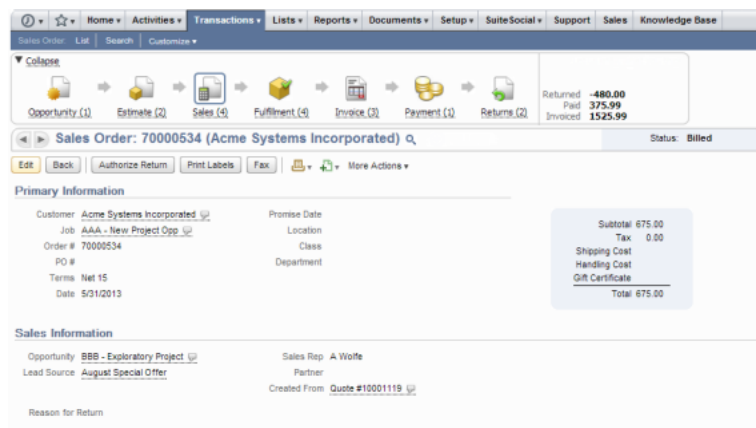
- Company Data
- Chart of Accounts
- Transaction Journal
- Sums and Balances List
- Account Sheets, Chart of Accounts
- Account Sheets, Debtors and Creditors
- Annual VAT Figures
- Tax Codes

A DTD file and an index are also generated during the export. GDPdU data export should be generated for a full financial year (based on accounting periods). Each export generates a total of eleven files for download.

This feature is available to users with NetSuite or NetSuite OneWorld accounts that have the Tax Audit Files SuiteApp and an active Germany nexus. For more information, see the help topic Germany GDPdU Data Export.

Transaction Timeline (Beta)

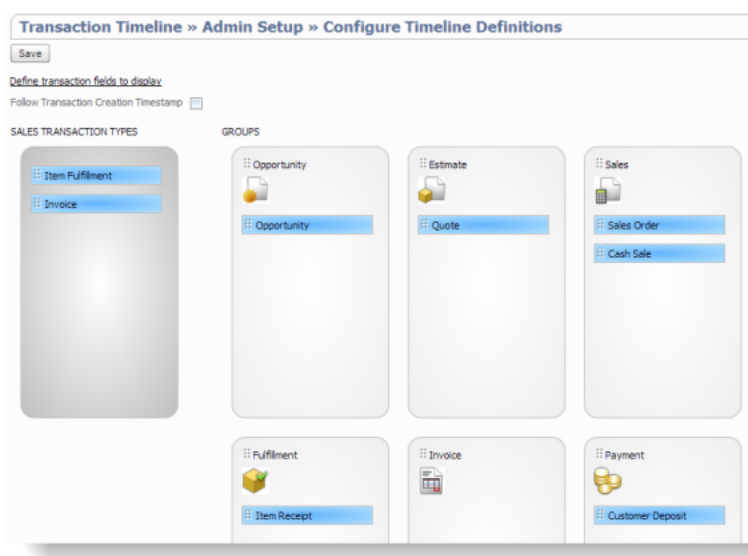
The Transaction Timeline SuiteApp, available in Version 2013 Release 2, enables you to see related transactions in an easily accessible format without navigating away from the current transaction. This SuiteApp groups related transactions in a timeline that you define and makes them accessible in the same view.



Previously, on the Related Records tab of some transactions you could access certain related transactions. Now, with the Transaction Timeline, you can customize which transactions are displayed, giving you the flexibility to see an entire order process from start to finish at the top of your transaction pages.

You have an idea of the current stage of a transaction, even without access to certain record types. The Timeline shows the transaction icon and the number of records, but does not show details if you do not have access to them.

You configure the Timeline by going to Setup > Customization > Transaction Timeline Setup, under Lists, Records, & Fields, and dragging relevant transactions into a record group. You can create multiple groups and change their sequence by dragging groups into the desired position, or change the look by selecting from a pre-loaded set of icons.



You can also configure the type of information that appears in the transaction summary on the right side of the timeline. This timeline uses saved searches to display summary information about this set of transactions. You can configure the timeline to show up to four different pieces of summary information by customizing existing saved searches.

The Transaction Timeline is available for installation as a SuiteApp:

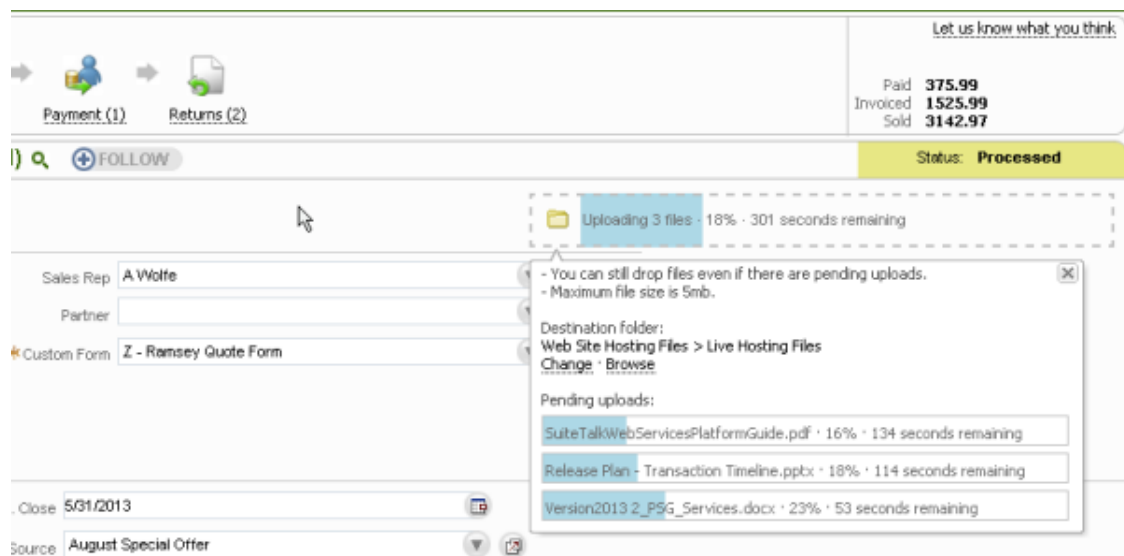
- Name – SuiteLabs Transaction Timeline
- Bundle ID – 40073
- Location – Production (Account ID 3727422)

For more information, see the help topic Transaction Timeline (Beta).

File Drag & Drop (Beta)

Important: The File Drag and Drop SuiteApp (Beta) is still under development. There may be significant changes between this beta version and the first production release.

With the File Drag & Drop SuiteApp, you can quickly attach files to records by dragging them from your desktop into NetSuite records.



File Drag and Drop is supported on all records and transactions that are exposed to scripting, including expense reports, quotes, custom records, and cases. You can also attach files to editable text fields and line items on transactions that support files at the line item level, including expense reports.

File Drag and Drop allows the attachment of multiple files with a maximum individual size of 5 MB. You can click the dropzone to see the status of uploads and can continue to add files even if there are items still in queue. Attached files are shown on the Files subtab of the record.

Your administrator can set default folders for each record type, or you can browse for other folders in the file cabinet before you upload.

File Drag and Drop is available for installation as a SuiteApp.

- Name – SuiteLabs File Drag and Drop
- Bundle ID – 41309
- Location – Production (Account ID 3751218)

For more information, see the help topic File Drag and Drop (Beta).

Subsidiary Navigator SuiteApp (Beta)

Important: Subsidiary Navigator SuiteApp (Beta) is still under development. There may be significant changes between this beta version and the first production release.

The Subsidiary Navigator (Beta) SuiteApp enables you to limit the information displayed on your dashboard, searches, and reports to a specific subsidiary.

The portlet contains a chart displaying all your subsidiaries. You can select from the subsidiaries shown to restrict access to records relevant to the chosen area. This is equivalent to setting the Restrict View field to a subsidiary option in the Set Preferences page.

Subsidiary Navigator is available for installation as a SuiteApp:

- Name – Subsidiary Navigator
- Bundle ID – 44281
- Location – Production (Account ID 3802169)

For more information, see the help topic Subsidiary Navigator (Beta).

Grid Order Management

NetSuite's Grid Order Management SuiteApp enables you to enter orders for inventory items in a grid entry form. You can configure matrix items and items with similar attributes in grid templates, and enter sales orders using the grid templates. You can also view item information such as Item Name, Location, Quantity Available, and Price as you enter orders. This streamlined format can save significant order entry time, increase the accuracy of orders for like items, and give users the opportunity to sell and fulfill orders faster. For more information, see the help topic Grid Order Management SuiteApp.

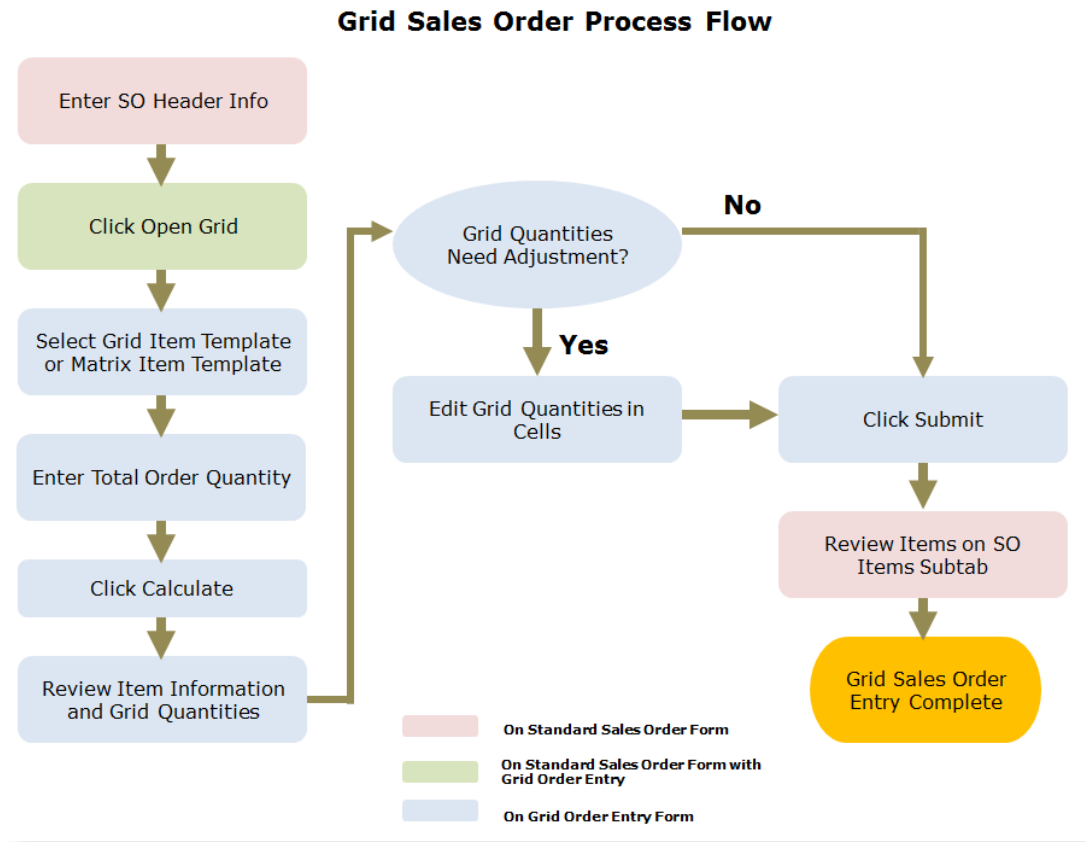
To begin using Grid Order Management, see the help topic Setting Up Grid Order Management.

Once setup is complete, you can use the Grid Order Management SuiteApp to quickly create sales orders.

Note: Grid Order Management is a managed SuiteApp. When a new version of this SuiteApp becomes available, it is automatically updated in accounts where it is installed.

Grid Sales Order Entry

When you have grid templates set up, you can create sales orders for the items assigned to grid templates. Using the Grid Order Entry form, you can use more than one grid template on the sales order at a time.



For more information, see the help topic [Entering a Grid Sales Order](#).

Recurring Billing Enhancements

Version 2013 Release 2 includes the following enhancements to the Recurring Billing SuiteApp:

- [Subscription Analytics](#)
- [Subscription Change Management Enhancement](#)
- [New Time Zone Implementation](#)
- [Unification of Subscription Billing Modes](#)
- [Streamlined Recurring Billing Processes](#)
- [Support for Core Charge Record](#)
- [Migration Support](#)

- Enhanced Framework for Recording Variable and Fixed Usage Data
- Multi-Queue Support for Rating Engine
- Other Recurring Billing Module Highlights

Subscription Analytics

The latest release of the Recurring Billing SuiteApp allows users to view revenue and usage metrics, billing forecasts, and KPIs from the NetSuite dashboard and subscription records.

The new version of the SuiteApp comes with following saved searches that users can configure and display on the dashboard as KPIs:

- Active subscriptions (ACV)
- Monthly recurring revenue (MRR)
- Total contract value
- Annual contract value
- MRR and ACV for direct new business
- Churn

The following metrics are added on the subscription record header:

- Monthly recurring revenue
- Annual contract value
- Total contract value
- Total usage per period

A new menu category is added under Subscription Billing which allows users to generate reports for:

- Monthly recurring revenue forecasts
- Total contract value forecasts
- New business forecasts
- Upsell forecasts
- Downsell business forecasts

The reports can be exported to PDFs, Microsoft Excel, and Microsoft Word.

Subscription Change Management Enhancement

This feature allows businesses to easily change active subscriptions by introducing user interface enhancements to the subscription, subscription item, and subscription plan records. A Change Order button is added to these records, and this button launches a window where you can update subscription, subscription item, or subscription plan information.

You can perform the following updates to a subscription, subscription item, or a subscription plan:

- Migrate one or more subscription items in to a subscription plan (subscription item record)
- Migrate subscriptions associated to a subscription plan to a new subscription plan (subscription plan record)
- Suspend one or more subscription items (subscription record)
- Suspend item from a subscription (subscription item record)
- Modify the quantity of usage units allocated for one or more secondary items per billing period (primary item)
- Modify the quantity of usage unit allocated for the item in a billing period (secondary item)
- Modify the start date of one or more subscription items (subscription record)
- Modify the start date of an item (subscription item record)
- Reactivate one or more suspended subscription items (subscription record)
- Reactivate a suspended subscription item (subscription item record)
- Terminate one or more subscription items (subscription record)
- Terminate a subscription item (subscription item record)

This feature addresses the following customer requirements:

- Streamlined process for migrating customers to a new plan
- Better control over subscription services in a subscription
- Easily accessible summary view of subscriptions
- More efficient interface for changing preset quantity on a subscription item

For more information on each subscription change order action, see *Managing a Subscription* in the Help Center.

New Time Zone Implementation

The latest release of Recurring Billing module provides a functionality that allows the billing system to use the time zone of the subscription and use it as reference for processing subscription events. This allows the SuiteApp to correctly process the subscription events by referencing the subscription's time zone setting instead of using the time zone specified in the user's NetSuite account preferences.

Note: It is possible for a single customer to have multiple subscriptions with different time zones.

Previously, the processing of subscription events uses the time zone setting of the current user which is set at Home > Set Preferences. This approach may result in processing errors specially when the daylight saving time and other offset rules differs between the two time zones. Under the new implementation, values in the Date/Time field, which is used to store date and time values, is converted using the subscription Time zone. The conversion is done using the time zone data of the IANA/Olson standard.

The date and time values displayed on recurring billing forms and dates entered on subscription change orders are converted relative to the subscription time zone. The date/time values entered on the **service date** field of a usage data should be entered in the subscription's time zone. Sibling or child subscriptions rated using the service date are processed using the subscription time zone as reference.

Note: Date/Time values of sublists are displayed relative to the time zone set in the account preferences.

The time zone value set in the user/company preferences will not be affected by this enhancement.

Unification of Subscription Billing Modes

This enhancement introduces code changes that affect how the bill cycle sets dates for a subscription with an Anniversary billing mode. Anniversary is a billing mode where customers are billed on the date that they signed up for the subscription. This enhancement is aimed at changing the behavior of anniversary billing to closely resemble the behavior of a subscription with Day of the Month (DOM) billing mode. Bill cycles of DOM subscriptions are aligned relative to the Bill cycle end date value set in the subscription billing preferences.

The Billing Date Type field in the sales order is deprecated to avoid confusion when the bill cycle end is set to the 28th, 29th, and 30th of the month. In the previous version of Recurring Billing, when you entered an order for items in anniversary subscriptions, users needed to specify whether these end-of-month dates represented an absolute value (earliest start date) or the last day of the month in the billing date type field. The bill cycle dates changed depending

the value selected. In the latest release of the SuiteApp, alignment of bill cycle dates is based on the item with the earliest start date from among the items in the subscription. The start dates of the other items vary depending on whether the subscription is new, or an existing subscription with new items to be added. Proration is applied on the first bill cycle of the other items, depending on how the items are aligned relative to the billing dates of the item with the earliest start date.

For examples of the various scenarios for determining billing dates of succeeding items and for applying proration relative to these dates, see the Billing Mode topic in the Help Center.

Streamlined Recurring Billing Processes

The latest release of the Recurring Billing SuiteApp provides new processing functionality that allows the billing system to record changes to the fixed usage quantities of subscription items and uses the new information for reporting purposes. For more information, see the help topic Recurring Billing KPIs and Reports.

The new release of the SuiteApp also introduces code changes that affect how usage charges are computed. These code refactoring involves transferring the tasks of computing usages charges from the billing engine to the usage rating script. This enhancement is aimed at reducing the time taken for the billing engine to generate invoices. The usage rating script is deployed to generate the usage-based charges, based on data from subscription charge custom records, regardless of the charge type used. Charge types specify the type of service (recurring service, variable usage, fixed usage, termination fee) for the subscription plan items that will use a charge plan. The usage rating script applies different charge calculations for different types of subscription items.

Support for Core Charge Record

The latest release updates the handling of charges in the Recurring Billing module.

Charges are now processed using the Core Charge functionality. Charges are reflected in the Charges tab in the Secondary Item form which replaces the Subscription Charges tab in earlier versions.

Subscription analytics are linked to charges via the Charge field.

Upon creation, a charge has the Stage field set to "Ready". Once the charge is billed, the field is automatically set to "Processed".

The following custom fields, previously associated with the custom Subscription Charge record, are now linked to the Core Charge record:

- Actual Usage
- Days

- Display Filter Counts
- End Date
- Historical Usage
- Included Usage
- Legacy
- Maximum Charge
- Minimum Charge
- Rate Plan
- Rating Details
- Sibling Usage
- Start Date
- Subscription
- Subscription Item Pri
- Subscription Item Sec
- Usage Unit

The workflow for calculating subscription item charges remains the same. No new fields were introduced in the form and the change does not affect how the charges are calculated.

For more information on the Core Charge functionality, please see the help topic [Charge-Based Project Billing](#).

Migration Support

The latest release provides migration support. Users can migrate their existing data from the older version of the Recurring Billing Module to the latest version using built-in scripts.

Enhanced Framework for Recording Variable and Fixed Usage Data

The latest release of the Recurring Billing SuiteApp implements an enhanced framework for the recording of variable and fixed usage data. This enhancement is aimed at streamlining the process of rating usage data or updates to usage data for invoicing. The enhancement also provides a functionality that captures new usage data or usage data changes and automatically update the Recurring Billing KPIs and reports. For more information, see the help topic [Recurring Billing KPIs and Reports](#).

Recording of Variable Usage Data of Subscriptions with Relative Billing Periods

The usage rating process searches for subscription analytics records associated with items for changes in variable usage data. The search is filtered within a date range specified by the user.

From the retrieved subscription analytics records, the usage rating process calculates the total usage, and derives the subscription item's variable usage total for that specified period.

A new analytics record is created if no analytics record is found and the total variable usage is stored in the new record. If no analytics record is retrieved within the specified date range, the rating processes searches for existing Subscription Charge records referenced by Subscription Analytics records within the billing periods that overlaps the date range. The total usage value returned from the subscription analytics records is compared with the total variable usage data on the usage table and adjusted on the subscription analytics table for any difference. The process of recording variable usage for an item is repeated until usage data has been recorded for all months in the relevant billing period. The complete billing period is then rated and totalled as the total usage quantity and total charge. The difference between the sum returned by the search and the charge calculated for the complete billing period is stored as an un-invoiced usage on the subscription charges record. The process of recording variable usage results in one or two analytics records created for each month depending on the billing period start date. Only one subscription analytics record per month if the billing period starts on the first day of the month.

Recording of Variable Usage Data with Historical Usage Data

If a subscription uses a rate plan wherein the rating of historical data is enabled and a fixed custom start date is set for the date range from which historical data is retrieved, three subscription analytics records are created to record usage data of the items for the first billing month. The three analytics records divide the usage data retrieved into specific date ranges. The first analytics record contains usage data from the start of the calendar month to the day prior to the start of the historical rating start date or the billing period anniversary date, whichever is earlier. The second analytics record contains usage data from the day after the end date of the first analytics record date range to the day prior to the start of the historical rating start date or the billing period anniversary date, whichever is dated later. The third analytics record contains usage data from the day after the end date of the second analytics record to the last day of the month. The creation of three analytics records is only applicable to the first billing month. After the creation of the three analytics records, the process of searching and recording variable usage data for subscriptions with historical data and fixed historical data start dates proceeds following the same process as the recording of variable usage data. For more information, see [Recording of Variable Usage Data of Subscriptions with Relative Billing Periods](#).

Recording of Changes in Fixed Usage Quantities

If a change in the fixed usage quantity of a subscription item is performed using the **modify quantity for fixed usage** change order action, the change is automatically recorded and saved as a subscription analytics record for use in reporting and for rating invoice charges. The

recording of fixed usage quantity changes is invoked as a step in the workflow of the modify quantity for fixed usage action. For more information, see the help topic [Updating the Fixed Usage Quantity of an Item](#). This step in the workflow invokes the creation of a new analytics record with an updated quantity and a new effectivity date. The previous analytics record associated with the item is automatically set to end on the effectivity date specified on the subscription change order action.

Multi-Queue Support for Rating Engine

The latest release of Recurring Billing SuiteApp includes Multi-Queue Support for Rating Engine. With multi-queue support, you can process higher volume of usage data with increased throughput.

Note: This feature is only available to users with SuiteCloud Plus License.

For more information, see the help topic [Multi-Queue Support for Rating Engine](#).

Other Recurring Billing Module Highlights

The new Recurring Billing module continues to support the following features:

- Flexible rate management that allows you to define the charging structure for a subscription, including defining whether items are usage-based or fixed recurring, usage units to be used, allocated free units per billing period, minimum and maximum charges, and pricing model (flat-rate, tiered, or volume pricing).
- Flexible invoice management that allows you to easily specify whether a customer is billed in advance or arrears, or if a subscription is invoiced on the day the customer signed up for the subscription (anniversary,) or on a specific day (day of the month).
- Improved customer retention through the specification of renewal options for subscription items at the end of the term. You can set up items for termination to automatically renew, or renew only after approval from the customer, or you can configure recurring billing to create a sales opportunity.

Electronic Payments Enhancements

Version 2013 Release 2 includes the following enhancement to the Electronic Payments SuiteApp:

- [Support for New Payment Formats](#)
- [Processing of Payments for Partner and Employee Commissions](#)

Support for New Payment Formats

This release of the Electronic Payments module supports the following payment file formats:

Electronic Fund Transfer

- PNC Bank format (US)
- JP Morgan (Multi-currency)
- Westpac – Deskbank (New Zealand)

Direct Debit

- BACS Direct Debit (UK)
- RID (Italy)

Processing of Payments for Partner and Employee Commissions

The latest release of the NetSuite Electronic Payments SuiteApp supports the processing of payments for partner and employee commissions. The previous version of the SuiteApp only supports the processing of outgoing payments for vendor bills, employee expenses, customer refunds, and incoming payments from customers.

You need to setup the partner record to allow the SuiteApp to process commission payments for that partner.

To setup a partner record for electronic payments processing:

1. Go to Lists > Relationships > Partners.
2. Click Edit next to the Partner record that you want to setup.
3. Click the Bank Payment Details subtab.
4. Check the EFT Bill Payment box Click the New Entity Bank Details button.
5. In the EFT Template field, select the payment file format that your bank uses to send electronic payments for your partner.
6. Enter the required bank details, and then save the Partner record.

You also need to setup employee records to allow the SuiteApp to process commission payments for those employees. For more information on setting up employee bank records, see the *Setting Up Bank Record of Employees* topics on the Help Center.

After setting up the bank details of partners and employees, you can start processing the payments of their commissions through the manual payment processing or automated payment

batch processing functionality of the SuiteApp. For more information, see the help topics [Processing Bills and Expenses in Batches](#) and [Processing Payments by Manually Selecting Bills and Expenses](#).

Support for Multiple Cluster Processing

This feature extends the current queue management functionality by allowing customers to simultaneously process payment batches from different subsidiaries.

The enhancement allows customers to:

- Assign specific payment processing queues to different subsidiaries
- Run simultaneous payment processing queues from different subsidiaries
- Simultaneously Run multiple payment processing queues from a single subsidiary

To setup the feature you need to configure the script queues and subsidiary assignments on the Queue Management tab under Payments > Setup > Electronic Payments Preferences. The tab now includes columns for Subsidiary, Parent Deployment, Payment Creator Deployment, Rollback Deployment, Reversal Deployment, Notification Deployment, and Default. You can select multiple subsidiaries to share a single script queue or multiple queues. To optimize the use of script queues, it is recommend that you assign script queues to subsidiaries in different time zones.

Note: You need to have a SuiteCloud Plus license to simultaneously process payment batches from one or more subsidiaries. Contact your NetSuite account representative for information on upgrading to SuiteCloud Plus.

Software Vertical Contract Renewals

Renewal Order Item Type Enhancement

The **Apply Upsell and Downsell on Renewal Transaction** preference has been added to the Software Vertical Contract Renewals SuiteApp. You can enable this preference to track and classify upsell and downsell contract items by order type in renewal transactions. See the help topic [Setting Contract Renewals Preferences](#).

By default, contract renewal automatically reflects upsell and returns that occurred during the original term of the contract. Upsell items that extended to the end of the contract term are renewed, and items that were returned are not renewed. Contract renewal processes upsell and returns in the following manner:

- New items added to a contract are classified as Contract — Renewal

- Downsell contract items are omitted from the renewal transaction

When the **Apply Upsell and Downsell on Renewal Transaction** preference is enabled, the system tracks the contract items you add and remove in renewal transactions, and identifies the items by order type. When a contract is renewed:

- Renewable upsell items are classified as Contract — Upsell
- Downsell items are indicated by two contract items:
 - the original contract item classified as Contract — Renewal
 - the negative line item classified as Contract — Downsell

Note: Only contract items removed through return authorizations are tracked as downsell. If the preference is enabled, you are asked to confirm the removal of contract items.

SuiteSocial

Version 2013 Release 2 includes the following enhancements to the SuiteSocial SuiteApp:

- [News Feed Content Search](#)
- [Support for Additional Record Types](#)
- [Support for Linked Records in Comments and Private Messages](#)

News Feed Content Search

You can now search for news feed posts using keywords from your SuiteSocial News Feed. Find the posts you want by entering part of or a complete keyword, or a string separated by spaces in the Search box above the news feed panel. The search treats each partial or complete word as a separate keyword and displays posts that contain matches per keyword. For more information, see the help topic [Using Search to Find News Feed Posts](#).

Support for Additional Record Types

The latest release of the SuiteSocial SuiteApp now supports Competitor records and the following standard item types:

- Assembly Item
- Lot Numbered Assembly Item
- Serialized Assembly Item
- Inventory Item

- Lot Numbered Inventory Item
- Serialized Inventory item
- Non-Inventory Item
- Download Item
- Other Charge Item
- Service Item

To activate SuiteSocial functionality on these records, see the help topic Step 4: Enable Records. Once SuiteSocial-enabled, you can work with these records in SuiteSocial. For more information, see the help topic Working with Records in SuiteSocial.

Support for Linked Records in Comments and Private Messages

You can now mention or tag colleagues and records in SuiteSocial comments and private messages. Tag or link records in your comments and private messages by typing @ followed by a part of or the entire record name like you would use NetSuite's global search, and selecting the record in the results. For more information, see the help topics in Using SuiteSocial News Feed.

Electronic Payments SuiteApp: Enhanced Email Payment Notifications

The NetSuite Electronic Payments SuiteApp has been updated to support email templates and add multiple recipients when sending remittance advice. Prior to this enhancement, the email recipient was sourced from the default email address of the vendor, employee, partner or customer record. It was not possible to change the intended recipient or add more recipients. Now, you can change the default email address and add more recipients, letting more people know that payment will be deposited or collected from their bank.

Support for email templates is also included in the update. The new version comes with an email template, or you can create and use your own email templates when sending payment notification. For more information, see the help topic Sending Payment Notifications.

User Interface

Support for Internet Explorer 8 Ending in Version 2013 Release 2

Be aware, that as of Version 2013 Release 2, the NetSuite application no longer supports the use of Internet Explorer 8. Users need to update to a more recent version of Internet Explorer, or switch to another supported browser such as Firefox or Chrome. Note that this end of support for IE8 does not apply to web stores.

As of Version 2013 Release 2, NetSuite will not be tested with IE8 and support will not be offered for defects specific to IE8. In addition, user interface enhancements planned for releases after Version 2013 Release 2 may prevent NetSuite from functioning at all in IE8.

For a list of currently supported browsers, see the help topic [Supported Browsers for NetSuite](#).

Administration

Version 2013 Release 2 includes the following enhancements to Administration features:

- [Security Questions Enhancements](#)
- [Security Challenge Column Added to Login Audit Trail](#)
- [Prevent Login with URL that Contains NetSuite Password](#)
- [Default Password Expiration of 180 Days](#)
- [Enhancements to Lockout Email Notifications](#)
- [Recent Records Tracking for Setup and Customization Pages](#)
- [Email Notification when Sandbox Refresh Complete](#)
- [End of Support for Gemalto Tokens](#)

Security Questions Enhancements

The first time users log in to NetSuite, they are prompted to set up three security questions. These answers are used to verify users' identity if they forget their password or if they log in from a new browser or a new computer. Version 2013 Release 2 includes the following enhancements to the security questions feature.

Security Questions Setup is Now Mandatory

Before this release, the security questions popup always included a Remind Me Later button that users could click to dismiss this prompt and log in without setting up security questions. Now, a user can only click the Remind Me Later button a maximum of 5 times. For each login after a user has clicked Remind Me Later, the popup includes text noting the number of remaining times that the user can dismiss this prompt. For the login after a user has dismissed the security questions prompt for the fifth time, the popup does not include the Remind Me Later button, and its text explains that the user is now required to set up security questions and answers before proceeding.

Other Security Questions Changes

- Users can now mask their answers to security questions by checking the Hide Answers box on the Set Up Security Questions popup, the Reset Password page, and the security

challenge page. When this option is enabled, answers to security questions are not displayed as the user types them, so onlookers cannot see this private text.

- The Set Up Security Questions popup is now displayed when users log in through any NetSuite URL, not just for the default login URL.
- More choices are now available for security questions, to provide better options for international users.
- Previously, a change to a user's email would erase existing security questions and answers, requiring a user to reenter this information. Now, a change to a user's email does not reset the user's security questions, just as it does not reset the user's password.

Security Challenge Column Added to Login Audit Trail

The Login Audit Trail is a specialized search that helps keep track of account users' login attempts. This search is available to account administrators at Setup > Users/Roles > View Login Audit Trail. Version 2013 Release 2 adds a Security Challenge column to Login Audit Trail results, to track the status of users' answers to security questions when these questions are presented.

Login Audit Trail Search: Results						
Date ▼	Email Address	User	Role	IP Address	Status	Security Challenge
6/26/2013 1:58 pm	mheisler@netsuite.com	Heisler, Melanie	Administrator	172.16.195.47	Success	-
6/26/2013 11:08 am	csloan@netsuite.com	Sloan, Cindy	Administrator	172.16.195.21	Success	-
6/26/2013 8:59 am	lgault@netsuite.com	Gault, Lewis	Administrator	172.16.30.72	Success	-
6/26/2013 8:09 am	lgault@netsuite.com	Gault, Lewis	Administrator	172.16.30.72	Success	-
6/26/2013 7:43 am	tadkins@netsuite.com	Tracy Adkins	Administrator	172.20.42.21	Success	-
6/26/2013 7:24 am	csloan@netsuite.com	Sloan, Cindy	Administrator	172.20.42.253	Success	-
6/26/2013 6:25 am	npittard@netsuite.com	Pittard, Nicole	Administrator	172.20.38.245	Success	-
6/24/2013 5:38 pm	agalero@netsuite.com	G, Anna		115.85.29.18	Failure	-
6/24/2013 2:26 pm	lgault@netsuite.com	Gault, Lewis	Administrator	172.16.30.72	Success	-
6/24/2013 12:46 pm	lgault@netsuite.com	Gault, Lewis	Administrator	172.16.30.72	Success	-
6/24/2013 10:05 am	lgault@netsuite.com	Gault, Lewis	Administrator	172.16.30.72	Success	-
6/24/2013 9:21 am	lgault@netsuite.com	Gault, Lewis	Administrator	172.16.30.72	Success	-

A security challenge is an instance when a user is prompted to answer a security question before login to NetSuite. This situation can occur when a user attempts to use a new browser or computer, when a user forgets their NetSuite password, or one year after a user sets up security questions. If the user answers the question correctly, the security challenge has a status of "Success", and login to NetSuite proceeds. If the user provides 6 consecutive wrong answers, the security challenge has a status of "Failure", the user is locked out of NetSuite for 30 minutes, and an email is sent to the account administrator.

The Security Challenge column in the Login Audit Trail indicates whether a security challenge occurred for each login attempt, and displays the status for any challenges that occurred. If this column does not display either Success or Failure for a row, it means no security challenge

occurred for that login. This information can help administrators to detect hacking attempts, which can be evidenced by a high number of failures in a short period of time.

Prevent Login with URL that Contains NetSuite Password

Before Version 2013 Release 2, it was possible for users to log in to NetSuite by entering a URL that included user name and password values as parameters. Users could bookmark a URL that contained their credentials to access pages in NetSuite. For example, they could use a URL like the following: `https://system.netsuite.com/app/login/nlogin.nl?submit=Submit&email=user1234@gmail.com&password=foobar1`.

As of this release, any login attempts that include a password as a GET parameter are rejected. In this case, a Login dialog displays the message “The method you are using to log in is not supported”, and the user is required to type in a password in order to log in.

Default Password Expiration of 180 Days

Account administrators can set the number of days that a login password can be used before a user is prompted to change it, in the **Password Expiration in Days** field at Setup > Company > General Preferences. Before this release, this field was blank by default, meaning it was possible for an account to have no requirement for users to ever reset their passwords. For all new accounts as of Version 2013 Release 2 or later, a value of 180 days is the default, ensuring password rotation at least every 6 months. Administrators can modify this value as needed.

It is not possible for NetSuite to reset the value of the Password Expiration in Days field for existing accounts. However, it is strongly recommended that administrators of existing accounts set this value to a minimum of 180 days.

Note the following:

- Days are calculated from the date that each user last changed their password, not from the date that the company preference is changed.
- In order to comply with Payment Card Industry (PCI) standards, employees with access to view unencrypted credit card numbers are automatically required to reset passwords every 90 days, unless the limit set here is shorter.

Enhancements to Lockout Email Notifications

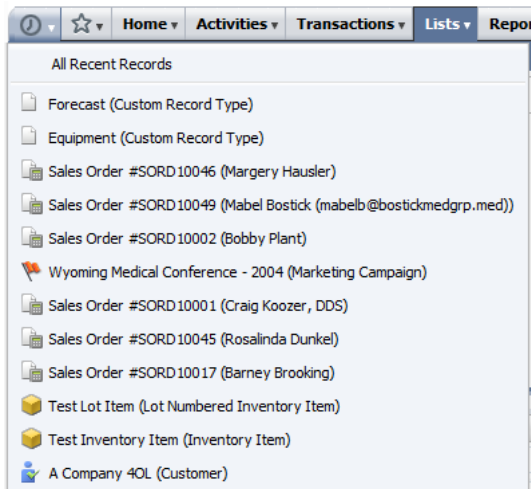
Administrators receive email notifications whenever users are locked out of their accounts. Lockout occurs after 6 consecutive failed login attempts. These failures result when users provide an incorrect password 6 times, or an incorrect answer to a security challenge question 6 times.

Version 2013 Release 2 provides enhancements to these administrator notifications. As of this release, each email message indicates whether the lockout resulted from an incorrect password

or an incorrect security question. Also, each message provides the name and ID of the account to which the locked out user has access, for better tracking. If the user has access to multiple accounts where the email recipient is an administrator, all of the accounts are listed in a single email message.

Recent Records Tracking for Setup and Customization Pages

The Recent Records portlet displays a list of NetSuite pages that you have recently created, modified, or viewed, with links you can click for quick access to these pages. Recent records also are listed under the clock icon menu in the leftmost tab of every NetSuite page.



Recent records tracking does not include all NetSuite pages. Version 2013 Release 2 supports recent records tracking for many setup and customization pages that were not previously tracked. The following table lists these newly supported pages, and the default navigation path for administrators to access each page.

Page / Record Type	Administrator Navigation Path to Record Type
Chart of Accounts	Setup > Accounting > Chart of Accounts
Accounting Periods	Setup > Accounting > Manage Accounting Periods
Customer Statuses	Setup > Sales > Customer Statuses
CRM Lists	Setup > Sales > CRM Lists
Sales Preferences	Setup > Sales > Sales Preferences
Users	Setup > Users/Roles > Manage Users
Roles	Setup > Users/Roles > Manage Roles
Role Permission Differences	Setup > Users/Roles > Role Permission Differences
Custom Lists	Setup > Customization > Lists

Page / Record Type	Administrator Navigation Path to Record Type
Custom Record Types	Setup > Customization > Record Types
Custom Entity Fields	Setup > Customization > Entity Fields
Custom Item Fields	Setup > Customization > Item Fields
Custom CRM Fields	Setup > Customization > CRM Fields
Custom Transaction Body Fields	Setup > Customization > Transaction Body Fields
Custom Transaction Column Fields	Setup > Customization > Transaction Column Fields
Custom Transaction Item Options	Setup > Customization > Transaction Item Options
Custom Item Number Fields	Setup > Customization > Item Number Fields
Other Custom Fields	Setup > Customization > Other Custom Fields
Custom Subtabs	Setup > Customization > Subtabs
Custom Entry Forms	Setup > Customization > Entry Forms
Custom Transaction Forms	Setup > Customization > Transaction Forms
Advanced PDF/HTML Templates	Setup > Customization > Advanced PDF/HTML Templates
Transaction Form PDF Layouts	Setup > Customization > Transaction Form PDF Layouts
Transaction Form HTML Layouts	Setup > Customization > Transaction Form HTML Layouts
Custom Address Forms	Setup > Customization > Address Forms
Custom Sublists	Setup > Customization > Sublists
Scripts	Setup > Customization > Scripts
Script Deployments	Setup > Customization > Script Deployments
Scripted Records list page	Setup > Customization > Scripted Records
SSP Applications	Setup > Customization > SSP Applications
Workflows	Setup > Customization > Workflows
Custom Plug-in Types	Setup > Customization > Custom Plug-in Types
Plug-in Implementations	Setup > Customization > Plug-in Implementations
Custom Centers	Setup > Customization > Centers
Custom Center Tabs	Setup > Customization > Center Tabs
Custom Center Categories	Setup > Customization > Center Categories
Custom Center Links	Setup > Customization > Center Links
KPI Scorecards	Setup > Customization > KPI Scorecards

Email Notification when Sandbox Refresh Complete

As of Version 2013 Release 2, an email notification will be sent each time a sandbox refresh completes. This email will be sent to all account administrators. It will include the sandbox account name and ID, the date the refresh was requested, the user who requested the refresh, the date of the production data used to refresh the sandbox, and the number of refreshes used and remaining.

End of Support for Gemalto Tokens

In Version 2013 Release 1, NetSuite began supporting RSA SecurID hardware and software tokens for two-factor authentication, in place of the previously supported Gemalto tokens. At that time, NetSuite stopped support for activation of new Gemalto tokens.

As of Version 2013 Release 2, NetSuite no longer supports preexisting Gemalto tokens. If any of your users still have Gemalto tokens, you need to replace them with RSA tokens. You can purchase RSA tokens from the vendor of your choice.

For more information, see the help topic Two-Factor Authentication.

Customer Relationship Management (CRM)

Version 2013 Release 2 includes the following enhancements to CRM features:

- Scriptable Email Templates (Beta)
- NetSuite for Outlook
- Voting for NetSuite Enhancements
- Enhancements to CRM Data Management
- Mapping Online Customer Form Fields to Customers and Contacts
- Read Receipt Support
- Issue Relationship Management
- Change to Case Editing
- New NetSuite for iPhone

Scriptable Email Templates (Beta)

With Version 2013 Release 2, you can create scriptable email templates that include application logic and use enhanced scripting. These templates can be created for email marketing campaigns and for system email. This enhancement allows you to define templates with personalized output for individual recipients based on the data from NetSuite (for example, from the customer's order history).

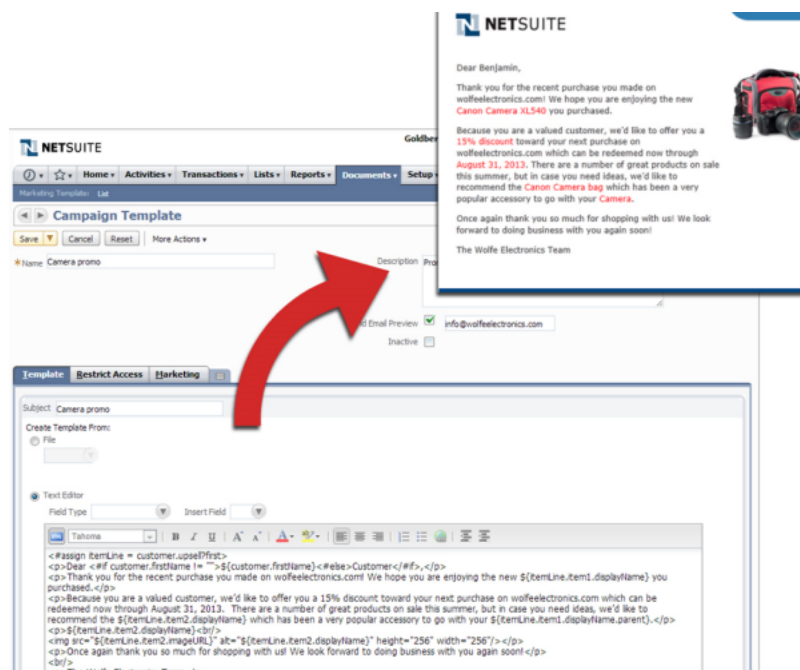
The new scriptable email templates use FTL syntax (FreeMarker Template Language). With this new feature, you can create dynamic templates that are personalized to individual recipients. For example, you could create an email template that refers to a customer's recent purchase and offers a list of upsell items, as well as a coupon code for their next purchase.

To create scriptable campaign email templates, go to Documents > Templates > Marketing Templates > New, in the Type column, click the Campaign type, and select Scriptable Templates (V2.0 Beta).

For scriptable system email templates, go to Setup > Company > System Email Templates, click New Template, and select Scriptable Templates (V2.0 Beta).

For a list of records and fields that are available for each type of template, use the dropdown lists in the fields Field Type and Insert Field on the template record.

Important: Scriptable email templates currently are available as a beta feature.



Important: Read receipts are not available for email sent through mail merge operations or email campaigns.

NetSuite for Outlook

The new NetSuite for Outlook client enables you to sync your Outlook email and records to your NetSuite account, giving you a complete view of your business relationships.

NetSuite for Outlook synchronizes your contacts, events, tasks, and phone calls, and allows you to save Outlook email messages to your NetSuite records. Duplicate record handling helps you avoid creating duplicate records in your NetSuite account.

Administrators can create configurations that control how users configure their data. This ability includes setting criteria to define the records that users can sync.

NetSuite for Outlook is compatible with Microsoft Outlook 2007 and 2010 (both 32- and 64-bit versions). You can use NetSuite for Outlook with Windows XP, Windows Vista, and Windows 7 operating systems.

To get started using NetSuite for Outlook, see the help topic NetSuite for Outlook Overview.

Voting for NetSuite Enhancements

With Version 2013 Release 2, you can visit SuiteIdeas to find and vote for your favorite NetSuite enhancements, or to enter new ideas.

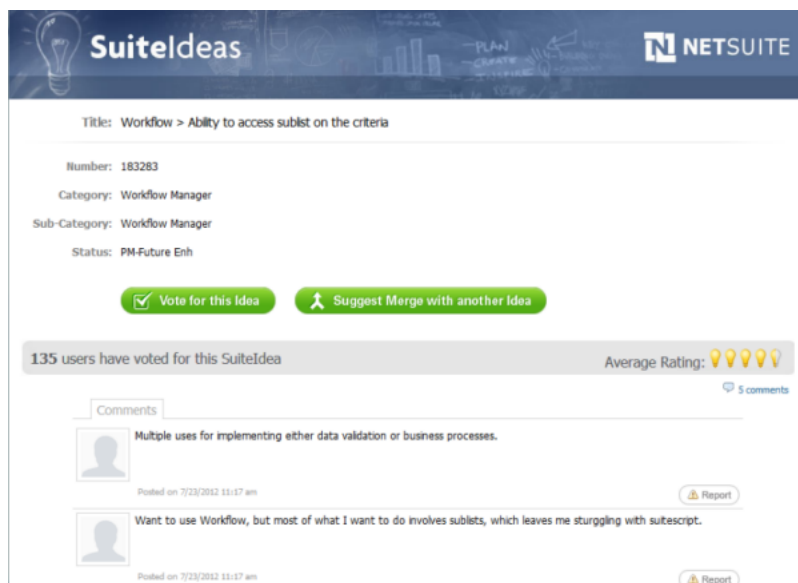


To access SuiteIdeas, you must first have access to the NetSuite Support Center. Click the Support tab, and click the link in the SuiteIdeas portlet. When the NetSuite Support Center opens, click the SuiteIdeas tab, and click the link in the portlet.

In SuiteIdeas, you can search for existing enhancement requests, browse for enhancements by category, and view lists of most popular and newest ideas.

Each enhancement request includes a title and description, the numeric ID for the request, its category, and the current status of the request.

You can click Vote for this Idea to add your vote, and you can enter comments to provide more information.



To add an idea of your own, click File a New Idea. You can then search to see if your idea is already in SuiteIdeas. If you don't find it, click File a SuiteIdea and enter your request. Each SuiteIdea is reviewed by NetSuite, and after it is approved, it is listed and can be viewed and voted on by other NetSuite users.

Enhancements to CRM Data Management

Version 2013 Release 2 includes the following improvements for managing CRM data:

- [Merging Large Numbers of Records](#)
- [Deleting Support Cases](#)
- [Merging Support Cases](#)

Merging Large Numbers of Records

Prior to Version 2013 Release 2, merge operations could only handle up to 200 duplicate records at the same time. With Version 2013 Release 2, duplicate merge mass updates give administrators the additional option to merge unlimited amounts of duplicate records.

With mass update duplicate resolution processes, you can set rules that the merge process uses to determine which record in a set of potential duplicates is the master record. To limit which records undergo the duplicate merge process, use the criteria to filter the search results.

To merge large numbers of duplicate records:

1. Go to Lists > Mass Update > Mass Updates.

2. In the Action column, expand the Duplicate Resolution options, and click the type of record you want to merge, such as Customers.
3. To define how NetSuite determines the master and duplicate records, use the dropdown list in the Master Selection Mode and field. You can use the following modes:
 - Created earliest — this mode selects the record that was created first as master record.
 - Most populated fields — this mode selects the record that has the most fields containing data as master record.
 - Most recent activity — this mode selects the record with the most recent activity as master record.
4. To define which records are selected, use the dropdown list in the Record Selection Mode field. You can use the following modes:
 - All from duplicate group — both the master record and the duplicate records are selected from the complete group of duplicate records, and search results are used only to determine the groups.
 - Master from duplicate group — the master record is selected from the whole group, and duplicates are selected from the search results only.
 - Master and duplicates from search only — both the master record and the duplicate records are selected only from the search results.

Like with other mass update operations, you can use various criteria to define which records are included in the operation and schedule regular duplicate merge operations to take place automatically at regular intervals.

For more information, see the help topic [Deleting Large Numbers of Records](#).

Deleting Support Cases

With Version 2013 Release 2, you can use a mass update operation to delete support cases. This operation allows you to delete an unlimited number of invalid support case records, such as those created by spam, at a single time.

To delete a support case:

1. Go to Lists > Mass Update > Mass Updates.
2. In the Action column, expand the Customer Support and Service options, and click Delete Cases.
3. Define the criteria for records you want to delete.
4. Click Preview.

5. All items are checked by default. You can clear the Apply box for any case you do not want to delete. To open and edit an record, click the record you want to edit.
6. Click Perform Delete to delete all checked support cases.

For more information, see the help topic Deleting Support Cases.

Merging Support Cases

With Version 2013 Release 2, NetSuite allows you to merge two support case records into one and reduce duplicates. All the data from the duplicate case record that is not already set in the master record is transferred to the master case record.

To merge duplicate cases click Merge. On the Merge Case window, select the case to be merged in the Merge From field and the master case in the Merge Into field.

The screenshot displays the NetSuite CRM interface. The top navigation bar includes 'Home', 'Activities', 'Transactions', 'Lists', 'Reports', 'Documents', and 'Setup'. The main content area shows the 'Edit Case' window for case 54, titled 'Cannot increase the video resolution (Subramani)'. The 'Merge' button is highlighted with a red box. Below the 'Edit Case' window, the 'Merge Case' dialog box is open, showing the 'Merge From' field set to '100087 Cannot increase the video resolution' and the 'Merge Into' field set to '100086 Need help with CD writer installation'. The dialog box also includes 'Save' and 'Cancel' buttons and a note: 'Cases from the same customer only'.

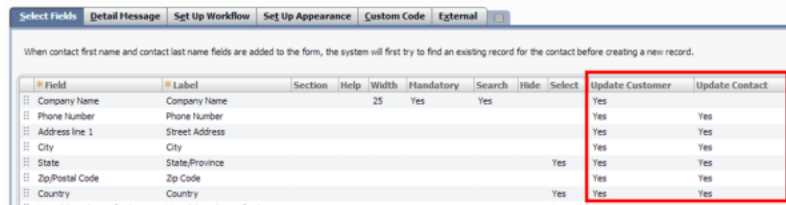
For more information, see the help topic Merging Support Cases.

Mapping Online Customer Form Fields to Customers and Contacts

Previously, both customer and contact records were updated when online customer forms were submitted. With Version 2013 Release 2, you can control which fields on online customer forms are used to update the customer or contact records.

For each selected field on the Select Fields subtab, you can check the boxes in the Update Customer and Update Contact columns to define which record type is updated when the form is submitted.

The new columns only apply to online customer forms that create customers as companies. The Update Contact column is not used for customers that are created as individuals.



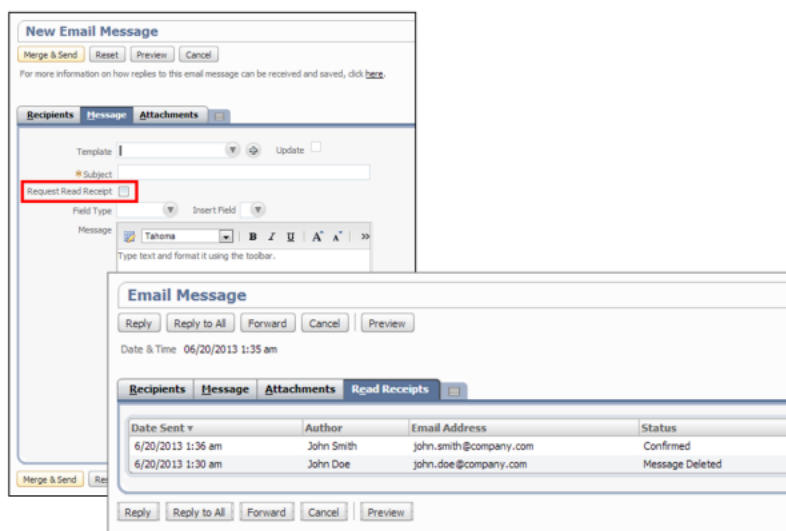
Field	Label	Section	Help	Width	Mandatory	Search	Hide	Select	Update Customer	Update Contact
Company Name	Company Name			25	Yes	Yes			Yes	
Phone Number	Phone Number								Yes	Yes
Address line 1	Street Address								Yes	Yes
City	City								Yes	Yes
State	State/Province						Yes		Yes	Yes
Zip/Postal Code	Zip Code								Yes	Yes
Country	Country								Yes	Yes

For more information, see the help topic [Selecting Fields](#).

Read Receipt Support

Read receipts ensure that recipients have received a message. This assurance is important when you are sending important information or time-sensitive transactions. With Version 2013 Release 2, NetSuite allows you to include read receipts on all outgoing email messages to individual recipients.

When you check the Request Read Receipt box on an email message in NetSuite, the email client of the recipient receives a read receipt request. Email clients handle read receipt requests in various ways. Depending on the configuration of the recipient's email client, the recipient can decide to respond to the read receipt request when the message is viewed. All received read receipts are forwarded to the original sender of the message and tracked on the message record in NetSuite, providing an audit trail within the application.



New Email Message

Merge & Send | Reset | Preview | Cancel

For more information on how replies to this email message can be received and saved, click [here](#).

Recipients | Message | Attachments

Template: [] Update

Subject: []

Request Read Receipt ☒

Field Type: [] Insert Field

Message: [Tahoma] [B] [I] [U] [A⁺] [A⁻] []

Type text and format it using the toolbar.

Email Message

Reply | Reply to All | Forward | Cancel | Preview

Date & Time: 06/20/2013 1:35 am

Recipients | Message | Attachments | **Read Receipts**

Date Sent	Author	Email Address	Status
6/20/2013 1:36 am	John Smith	john.smith@company.com	Confirmed
6/20/2013 1:30 am	John Doe	john.doe@company.com	Message Deleted

Reply | Reply to All | Forward | Cancel | Preview

Important: Read receipts are not available for email sent through mail merge operations or email campaigns.

For more information, see the help topic Requesting Read Receipts.

Issue Relationship Management

Prior to this release, you could not specify relationships or links between issues. With Version 2013 Release 2, the new Related Records subtab allows you to specify the relationship between two issue records. For example, you can define whether two issues are related or if one issue was caused or injected by the other.

You can select from the following filter criteria for relationships:

- Blocks
- Is blocked by
- Depends on
- Duplicates (only one relationship of the type "Duplicates" allowed per issue)
- Duplicated by
- Follow up for
- Followed up by
- Injects
- Injected by
- Relates to
- Is required for

Note: The Related Records subtab replaces the Duplicate Of field.

The screenshot shows the NetSuite CRM interface for editing an issue. The top navigation bar includes links for Home, Activities, Transactions, Lists, Reports, Documents, Setup, Support, Sales Tools, and Knowledge Base. The main form is titled 'Edit Issue: 1 Issue 1' and contains several sections: 'Primary Information' with fields for Issue Number, Issue Type, Product Team, Assigned To, Issue Status, Severity, Priority, and Showstopper; 'Abstract' with a text area; 'Problem Information' with fields for Item, Source, Reproduced, Product, Module, and Reported By; and a 'Related Records' subtab. The 'Related Records' subtab displays a table with columns for Relationship, Issue No., Abstract, Assigned To, Status, and Comment. It lists two related issues: 'Issue 2' (Submitted) and 'Issue 3' (Closed - Duplicate). The 'Related Records' subtab also includes a 'Duplicate Of' field and a 'Showstopper' checkbox.

For more information, see the help topic Working with Related Issues.

Change to Case Editing

Prior to Version 2013 Release 2, two users could edit the same case simultaneously and then both could save their changes. Now, case records have the same restrictions as other records. If you are editing a case, and another user edits and saves the same case record, when you click Save, an error is shown preventing you from saving the record.

Important: If you have existing SuiteScript that updates cases even when they are open for editing by another user, those scripts may need to be updated to continue to operate as intended.

New NetSuite for iPhone

Important: The new NetSuite for iPhone app is targeted to be available in the third quarter of 2013.

The new NetSuite for iPhone app is a completely new app built from the ground up. The previous NetSuite iPhone app was limited to read-only access to your data - allowing you to keep up to date with many aspects of your business. The new app allows you to actively interact with your business by inputting and editing data, and provides access to more information than before. The main features of this new app include:

- No setup required. It works "out of the box" with existing customizations, so you can begin working with your data remotely as soon as you have downloaded the app.

- Offers full (create, read, edit and delete) access to many record types, as well as access to your Home Dashboard, KPIs, Calendar, and Saved Searches. See [Mobile Access to Your Home Dashboard](#).
- Optimized calendar management includes a free/busy indicator, as well as a monthly, weekly, and daily views.
- One-way dashboard synchronization provides up to date access of all the data from your Home Dashboard.

For more details about the new NetSuite for iPhone app, see the following:

- [NetSuite for iPhone Setup Requirements](#)
- [NetSuite for iPhone Supported Roles](#)
- [Mobile Access to Your Home Dashboard](#)
- [NetSuite for iPhone Supported Record Actions](#)
- [NetSuite for iPhone Additional Features](#)
- [NetSuite for iPhone Current Limitations](#)

NetSuite for iPhone Setup Requirements

The app is primarily designed for use on the iPhone, but will work on any iOS device with Apple iOS 6.x or higher that has an internet connection.

Once you have downloaded and installed the app, no further setup is required. Most standard roles will be able to access NetSuite without further configuration.

NetSuite for iPhone Supported Roles

All standard roles, including Employee Center and Advanced Partner Center, are supported and enabled by default. Customized roles will need the Mobile Device Access permission added before they can access NetSuite from their mobile device. Administrators can add and remove access for individual roles as required.

To add the Mobile Device Access permission to a role:

Note: Only administrators can add this permission to a role. See the help topic [Customizing or Creating NetSuite Roles](#) for further information on setting permissions.

1. Go to Setup > User/Roles > Manage Roles.
2. Click Edit/Customize next to the relevant role.

3. In the Permissions tab, click on the Setup subtab.
4. Click Add and select the Mobile Device Access permission.
5. Click Done, then save the changes.

Note: Customer, Vendor, and Partner Center roles are not supported

Mobile Access to Your Home Dashboard

The new NetSuite for iPhone app provides one-way desktop to mobile device synchronization of all your Home Dashboard portlets, including Custom Searches, KPI Meters, Trend Graphs, Report Snapshots, Key Performance Indicators, Reminders, and KPI Scorecards.

Note: Device synchronization is not supported for Reports portlets and for any other custom portlets that are 'record type specific', such as Tasks and Shortcuts.

NetSuite for iPhone Supported Record Actions

Create, Read, Edit, and Delete are all available for the supported record types listed below. Additional supported actions include: converting an estimate to a sales order, approving a sales order, and workflow triggers such as upon record save.

Custom fields, custom (preferred) forms, and client and server custom scripts are all supported out of the box, so there is no need to make changes to your existing forms and scripts.

To make your data easy to work with, the app provides infinite subrecord drill-down, a Quick Add feature, and the Record Navigator for easily navigation through large data sets.

Supported Record Types

- Cases
- Contacts
- Customers
- Events
- Estimates
- Expense Reports
- Invoices
- Issues
- Items
- Leads

- Opportunities
- Phone Calls
- Prospects
- Sales Orders
- Tasks

NetSuite for iPhone Additional Features

- Calendar
- Lists
- Saved Searches
- Other Features

Calendar

The calendar view is optimized to simplify and speed up your routine tasks. Features such as list, week and month views, and a free/busy indicator help you easily plan your time. You can view other activity types inline, such as Phone Calls and Tasks, and the Swipe Action feature allows you to quickly accept invitations and complete phone calls. You can also view other calendars, such as those of employees and resources.

Lists

Lists enable you to see and interact with record types more effectively using the following features:

- Swipe Actions enable you to access common tasks, such as Call, Edit, and Email, directly from the list item.
- You can quickly filter or sort lists using the Quick-Sort and View Filters features.
- Lists can be viewed in landscape mode to improve readability for long record entries.

Saved Searches

All saved searches can be viewed, except for those that include a summary. The search results can be ordered by any column, and can also be viewed in landscape mode. You can use the Global Search feature to easily locate any of your saved searches.

Other Features

- You can switch roles on the fly, making it easier to manage multiple roles.

- The Recent Records feature allows you to quickly return to a recent activity or other record.
- Global Search has the same functionality as in the NetSuite application.
- Quick Add allows you to quickly add a new record from your current action.
- You can use the Remember Me feature to keep you logged in on your mobile device.

NetSuite for iPhone Current Limitations

There are several actions that cannot be performed via the NetSuite for iPhone app, or that do not provide the same behavior and access to data as in the NetSuite application.

- It is not possible to login using a Customer, Vendor, or Partner Center role as these roles are not supported. See [NetSuite for iPhone Supported Roles](#).
- Custom record actions and buttons are not supported, and it is not possible to create a new child record from a parent record. (For example, you cannot create a contact from a customer).
- Saved searches that include a summary are not supported.
- Reports are not supported, except for report snapshot graphs.
- The Advanced Bin Management feature is not supported.
- The vendor-specific objective evidence (VSOE) feature is not supported.
- Reminders open up in a web browser.
- Form customizations that alter the DOM are not supported.

Enterprise Resource Planning (ERP)

Version 2013 Release 2 includes the following enhancements to ERP features:

- Improved Credit Card Security
- GL Audit Numbering
- Charge-Based Billing
- Deferred Revenue Reclassification Enhancements
- Variable Revenue Recognition Schedule Changes
- Customizing Company Records
- Tax Period Posting Changes

- Attaching Line Item Receipts to Expense Reports
- Expense Report Line Item Reference Numbers
- Quantities and Rates for Expense Categories
- General Shipping Enhancements
- UPS Shipping Enhancements
- Component Yield Preferences
- Costed BOM Inquiry
- New Bulk Process Status Pages
- Mark All Bulk Process Work Orders
- Units of Measure Validation on Transactions
- Standard Cost Material Overheads
- Demand Planning and Inventory Allocation
- Landed Cost Allocation Per Line

Improved Credit Card Security

In an effort to increase the security of your customers' credit card information, beginning on **January 9th, 2014**, search criteria based on the Credit Card Number field will only be able to use the following operators: is empty or is not empty.

Existing saved searches that use other operators (like **is** or **is not**) with the Credit Card Number field will be deleted from your account after this date.

Additionally, any credit card number searches executed programmatically through Web Services, SuiteScript, or SuiteFlow that use disallowed operators will no longer work.

GL Audit Numbering

The GL Audit Numbering feature applies gapless numbering sequences to all GL posting transactions, allowing your company to meet international auditing requirements.

With this feature enabled, GL audit numbering is listed as a required task for the year-end close process. GL numbering is independent of other transaction numbering you have set up at Setup > Company > Auto-Generated Numbers.

See the help topic Using GL Audit Numbering for more information.

GL numbering can be done only in the last month of the fiscal year, period, or quarter (depending on the GL numbering method you use.)

The following transactions are numbered through this process:

- All transactions that post to the general ledger
- Voided transactions
- Transactions that have been voided through reversal transactions and the reversal transactions themselves
- Canceled vendor bills

Note: Transactions that do not post until they are approved (including commissions, journals, expense reports, and vendor bills) are only included in GL numbering after they are approved.

When you run GL audit numbering by quarter or year, even transactions in closed periods are numbered for the purposes of GL auditing.

You can enable GL Audit Numbering at Setup > Company > Enable Features > Accounting, under Basic Features. The Accounting Periods feature is required to use the GL Audit Numbering feature.

You can choose the frequency of GL audit numbering at Setup > Accounting > Accounting Preferences. On the General subtab under General Ledger, in the GL Audit Numbering Method field, select Base Accounting Period, Quarter, or Year. This choice determines when the GL audit numbering task is available to be processed as part of closing a period.

Warning: Changing the GL Audit Numbering Method preference can lead to previously numbered transactions being assigned new numbers.

Numbering GL transactions is controlled by the Manage Accounting Periods permission.

For more details about GL Audit Numbering, see the following:

- [Generating GL Audit Numbering Sequences](#)
- [GL Numbering Reports](#)
- [GL Numbering and Multiple Calendars](#)
- [GL Numbering and Intercompany Journal Entries](#)

Generating GL Audit Numbering Sequences

When you are in the last month of a period, you can set up GL numbering for that period and have NetSuite generate and assign the number sequence. If you use NetSuite OneWorld, you can create sequences for each subsidiary.

Note: The same sequence of numbers you assign to a period's transactions could be assigned to the transactions in another period if you define the sequence in the same way. To avoid this duplication, make sure the sequences you assign to each period are unique.

When you are in the process of closing a period, you can run GL numbering as often as needed to address gaps in numbering that might occur as a result of adjustments made to your GL. The GL numbering task is only available when you are in the last month of the period.

To run GL audit numbering:

1. From the Period Close Checklist, complete the tasks ahead of GL Audit Numbering.
2. Click the arrow beside GL Audit Numbering to open the Task: GL Audit Numbering page.

Period Close Checklist: Dec 2009			
Period Name: FY 2009 : Q4 2009 : Dec 2009		Start Date: 1.12.2009	
Status: Open		End Date: 31.12.2009	
Go To Task	Task	Status	Modified By
	Lock A/R		
	Lock A/P		
	Lock All		
	Resolve Date/Period Mismatches		Wolfe, J
	Review Negative Inventory		Wolfe, J
	Review Inventory Cost Accounting		Wolfe, J
	Revalue Open Foreign Currency Balances		Wolfe, J
	GL Audit Numbering		
	Close		

3. Click GL Numbering.
4. On the GL Audit Numbering Setup page, enter a name for the sequence.
5. If you use NetSuite OneWorld, select the subsidiaries with transactions you want to include in the sequence.

You can select multiple subsidiaries for a single sequence by holding CTRL and clicking the names of those subsidiaries in the Subsidiaries column. A subsidiary can only be assigned to a single sequence.

6. Enter the prefix and suffix you want to apply to the transaction numbers for this sequence.
7. Enter the maximum number of digits allowed for each transaction number.
8. Enter the first number to assign in this sequence.
9. In the Order by column, choose how you want to order the transactions for numbering.

You can order transactions by transaction date or transaction creation date.

10. Repeat these steps for each sequence you want to generate.
11. Check the box next to each sequence you want to run.

12. Click Run.

When you create GL numbering sequences by base accounting periods, the name, prefix, and suffix from the previous period are set for the next period's sequence automatically. The first number in the sequence is set to one higher than the last number of the previous period's sequence, helping you maintain some consistency in numbering between periods.

GL Numbering Reports

The new GL Audit Numbering report lists all GL posting transactions in gapless numbered order. This report can be submitted to government tax authorities where required. You can filter this report by sequence.

To view the GL Audit Numbering report, go to Reports > Financial > GL Audit Numbering.

To ensure that this report shows all transactions for the periods shown and that there are no gaps in the numbering, go to Home > Set Preferences > Reporting/Search and set the Report by Period preference to **Financials Only** or **All Reports**.

Note: If the **Allow Transaction Date Outside of Posting Period** preference is set to Warn or Allow in your NetSuite account, when you run the GL Audit report you may see transactions with dates that do not fall within the period for which you are viewing GL numbers.

You can add GL numbers to a standard report by customizing the report to include the **GL#** column. You can also include the GL sequence that was used to number each transaction.

GL Numbering and Multiple Calendars

If you use the Multiple Calendars feature and have multiple fiscal calendars that have different start and end periods or that are structured differently, you can only run GL numbering for the calendars in which you are in the last month of the period.

For example, you run GL numbering every quarter. You have two fiscal calendars. One calendar begins its fiscal year in March and the first quarter ends in May. The second calendar begins in January, and the first quarter ends in March. If you view the Manage Accounting Periods page in March, you can only run GL numbers for the second calendar. You cannot number the GL transactions for the subsidiary that uses the first calendar until May.

GL Numbering and Intercompany Journal Entries

Intercompany journal entries are associated with two different subsidiaries. When you run GL audit numbering for intercompany journal entries, a unique GL number is assigned for each subsidiary in the journal entry. Each intercompany journal entry is assigned two GL audit numbers.

Charge-Based Billing

Charge-Based Billing allows you to create charges that you can bill to your customers. When used with the Advanced Projects feature, you can apply rules to projects that generate charges that are included on project invoices.

Charge rules offer flexibility in calculating the billable value of project activity. Charge rules can be based on:

- the completion of project milestones
- project progress
- time entered for projects
- fixed amount generated on fixed dates (for example, up front materials costs)

Charges are generated according to charge rules that you define on projects. Charge rules determine the amount and sometimes the date of the charges created.

For more information about this feature, see the following:

- [Setting Up Charge-Based Billing](#)
- [Creating Charge-Based Billing Projects](#)
- [Defining Charge Rules](#)
- [Forecast and Actual Charges](#)
- [Service Items and Billing Schedules with Charge-Based Billing](#)
- [Generating Charges](#)
- [Project Billing Report](#)
- [Managing Pending Charges](#)
- [Billing Charges](#)
- [Copying Projects](#)
- [Scripting and Charges](#)

Setting Up Charge-Based Billing

You can enable the Charge-Based Billing feature at Setup > Company > Enable Features > Transactions under Billing. The Advanced Projects and Advanced Billing features are required to enable Charge-Based Billing.

When you enable the feature, a new Charge-Based billing type is available on project records.

You can set the default initial billing status for each charge rule type at Setup > Accounting > Accounting Preferences > Items/Transactions under Charge-Based Billing. For a charge to be billed, it must have the Ready stage.

Creating Charge-Based Billing Projects

Projects that generate charges as billable amounts use the billing type Charge-Based.

To create a charge-based project:

1. Create a new project record.
2. On the Financial subtab, under Billing Information, select **Charge-Based** in the Billing Type field.
3. Select a billing schedule.

This billing schedule is only used to calculate sales forecast amounts.

4. Enter other project information.
5. Click Save.

After you save the project, you can create charge rules to define how charges are calculated for the project.

Defining Charge Rules

Charge rules determine the billing rate, the timing of charges, and the stage of a charge when it is generated.

Charge rules can be either fixed fee or time-based. Fixed fee rules can generate charges on fixed dates, when milestones are reached, and upon project progress. Time-based rules generate charges when time entries are entered.

Charge rates are determined by the charge rules. Fixed fee charge rules require that a rate be entered on the rule. When you have a charge-based project with a milestone rule, the milestone record does not include the price fields. Pricing is instead determined by the charge rule.

Time-based charge rules can calculate rate from the service item on the time entry, from the rates entered on the Rates subtab of the employee record of project resources, or from an employee billing class.

You can define multiple charge rules for a single project, allowing you to create complex billing criteria based on a variety of factors. For example, you might want to bill a project based on time and materials. You would create two charge rules: one fixed date rule for the up-front billable amount for materials and one time-based rule that creates charges for time entered.

To create a fixed fee charge rule:

1. Click the Financial subtab.
2. Click the Fixed Fee Charge Rules subtab.
3. Click the button for the kind of rule you want to create.
4. Enter a name for the rule.
5. For fixed date rules, enter the date you want to create the charge.
6. For milestone rules, select the milestone of project task that causes a charge to be generated upon its completion.
7. In the Amount field, enter the charge amount for this rule.

For project progress rules, this amount is charged each time a charge is generated according to the recurrence you set below.
8. Choose whether charges created by this rule are created with the stage of **Ready** or **Hold**.

Charges can be set to Hold if you have an approval process for charges prior to billing them.
9. Select the service item that you want to use to determine the income account for charge revenue and to associate the charge with the sales order.

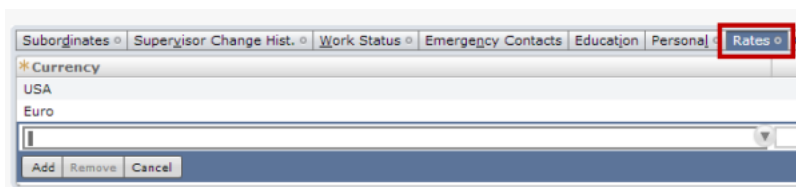
Note: The rate on this service item is not used as the rate for charges.
10. For project progress rules Under Recurrence, set the frequency that charges are generated based on the project's percent complete.
11. Click Save.
12. Repeat these procedures for each fixed fee charge rule you want to use on the project.

To create a time-based charge rule:

1. Click the Financial subtab.
2. Click the Time-Based Rules subtab.
3. Click the button for the kind of rule you want to create.
4. Enter a name for the rule.
5. On time-based rules, in the Rule Order field, enter where this rule should run relative to other time-based charge rules.
6. Choose whether charges created by this rule are created with the stage of **Ready** or **Hold**.

Charges can be set to Hold if you have an approval process for charges prior to billing them.

7. On the Rates subtab, in the Rate Basis field, choose how rates are determined for charges created with this rule. Choose one of the following:
 - **Billing Classes**— If you use the Per-Employee Billing Rates feature, you can calculate charge rates based on billing classes assigned to your project resources.
 - **Resources**— This option allows you to set rates for each resource. These rates are set on employee records under the Human Resource subtab on the Rates subtab. Rates set on the project task are not used for charge calculation.



- **Service Items**— This option takes the rate from the service items selected on the time entries logged for this project.
8. In the Rate Multiplier field, enter a decimal number you want to multiply the calculated rate by to determine the billable amount for the charges created by this rule.
 9. If you want to round the time logged for this project for the purpose of charge calculation, select a rounding method.
 10. Check the Do Not Bill Entries Exceeding Cap box if you do not want value above the cap to be billed to the customer.
 11. Enter the number of hours or currency amount you want to serve as the cap.
 12. Check the Adjust Time to Fit Under Cap box to change the total time logged for this project to the cap you entered.
 13. Click the Filters subtab.
 14. Enter any criteria you want to use to determine which time entries are used to generate charges.
 15. Click Save.
 16. Repeat these procedures for each time-based charge rule you want to use on the project.

Forecast and Actual Charges

For charge-based projects, there are two types of charges: actual and forecast.

Actual charges are amounts calculated from charge rules and can be billed to customers.

Forecast charges are used to calculate sales forecast amounts on reports and in the cost, revenue, and profit estimate fields on the project record.

Forecast charges are automatically generated for all projects with the In Progress status on a nightly basis. This means that forecast amounts are never more than 24 hours removed from their last update.

Once a forecast charge has been generated for a charge rule, a new forecast is not created unless a change is made that impacts that forecast (such as an update to the charge rule or to the estimated work for a task.) When a forecast charge requires an update, the old forecast charge is deleted and a new one is generated.

You can initiate the generation of forecast and actual charges on the Charge Run History subtab on the project.

Service Items and Billing Schedules with Charge-Based Billing

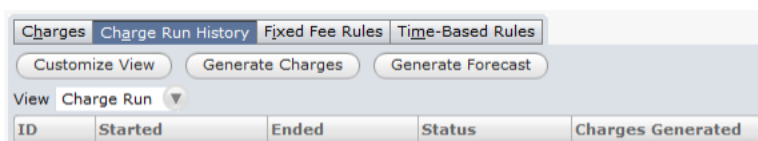
Charge-based billing uses service items to associate charges with sales transactions and to set the income account for charge-based project revenue. Service items are not always used to determine the billing rate for charges but are instead used to categorize charges on transactions. The pricing defined on service items is only used if you choose to use the service item price to determine rate for time-based charge rules.

A billing schedule must be set for each charge-based project, but billing frequency is always determined by the charge rule. The billing schedule is only used to calculate sales forecasts.

Generating Charges

Actual charges are generated daily for projects that have been updated. For milestone and project progress rules, charges are generated when a milestone is marked completed.

If you have made changes to a project and want to update the charges, click the Financials subtab on the project, and then click the Charge Run History subtab. Click the Generate Charges button to generate actual charges. Click Generate Forecast to update the forecast charges.



You can also create charges manually at Transactions > Customers > Create Charges.

Project Billing Report

The new Project Billing Report presents the billed value of projects. This report includes the billings to date, pending charges, and the total current billing amount as well as the forecasted

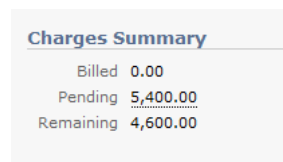
billing total for each project. You can filter the projects shown on this report by project status by changing the Customer Status field in the footer of the report. By default, this report is filter to show In Progress projects only.

To view the Project Billing Report, go to Reports > Time & Billables > Project Billing Report.

Managing Pending Charges

The Charges Summary fields on a project record provides a quick view of the amount billed, the charges that are pending billing, and the remaining forecasted revenue of the project.

After charges are generated, you can click the Pending amount on the project to manage any unbilled charges.



Charges Summary	
Billed	0.00
Pending	5,400.00
Remaining	4,600.00

From the Manage Pending Charges page, you can change the charge stage or zero charge amounts out.

Billing Charges

Charges with the stage Ready for Billing can be included on a sales order entered for a project. When you create a project sales order, those charges are on the Items subtab grouped by service item. The rates on these line items come from the charge record. Billing charge-based sales orders follows the standard invoicing process.

Billed charges are reflected in the Charges Summary fields on the project record.

Copying Projects

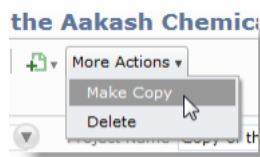
If your company uses similar charge rules for all projects, you can use the Make Copy option to create a duplicate project.

When you make a copy of a project, all project tasks and charge rules are included in the copy.

To copy a project:

1. Edit the project you want to copy.
2. Enter a name for the copy.

3. Make any other changes you need for the copy.
4. Under More Actions, select Make Copy.



Scripting and Charges

The charge record and the charge rule record are available for use in developing SuiteScript and Web Services solutions.

Important: Any custom solutions you build could be incompatible with enhancements and new billing features released in future versions of NetSuite.

Deferred Revenue Reclassification Enhancements

In Version 2012 Release 2, NetSuite introduced the Advanced Revenue Commitments process to reclassify deferred revenue for revenue commitments. Reclassification automatically adjusted the deferred revenue, unbilled receivables, and foreign currency gain/loss amounts at the order level.

With Version 2013 Release 2, the reclassification process for revenue commitments now includes the following enhancements:

- Line level deferred revenue reclassification properly adjusts multi-element sales contracts using item specific revenue and deferred revenue accounts, and reclassifying deferred revenue per transaction line. See [Line Level Deferred Revenue Process](#).
- Foreign currency gain/loss is calculated for each line and summed at the order level. Item specific revenue and deferred revenue account balances are adjusted for currency gain/loss instead of currency adjustments being posted to order level accounts. Open unbilled receivable balance accounts are now revalued.
- New reports are provided to audit reclassification details, including the Deferred Revenue Reclassification Report and the Deferred Revenue Reclassification Activity Report. See [New Reclassification Reports](#).

For more information about this feature, also see the following:

- [Revenue Reclassification UI Changes](#)
- [Adopting the Deferred Revenue Reclassification Enhancements](#)

Line Level Deferred Revenue Process

As before, you should reclassify deferred revenue balances *after* you post revenue recognition journal entries, by running Deferred Revenue Reclassification. Go to Transactions > Financial > Create Deferred Revenue Reclassification Journal Entries.

The Deferred Revenue Reclassification process automatically performs the following steps to calculate and post journal entries as needed:

1. **Billing account allocation:** This step redistributes the total billing amount of the order across the individual line items of the bundled sale, based on the revenue allocation ratio of the lines. It determines the billing amount to allocate to individual deferred revenue accounts for each line. This calculation logic is also referred to as 'carve in/carve out'. For lines where the total sales amount is larger than the total revenue allocation amount, the excess billing amount is allocated (carved out) to other lines (carved in).

Note: This step is skipped if the sales transaction with a revenue commitment is not treated as a multi-element bundle contract. No allocation is required because the revenue allocation amount is the same as the original sales amount for the item.

2. **Line level foreign currency adjustment:** The foreign currency gain/loss is calculated based on the billing and revenue recognition rates for each line. It then posts as an adjustment to the revenue and deferred revenue accounts associated with each line item. In prior releases, this adjustment was made at the order level.
3. **Order level unbilled receivables adjustment:** The unbilled receivable adjustment is still calculated at the order level, by comparing the accumulated base currency billing amount to the accumulated base currency revenue recognition amount. The unbilled receivable adjustment is denominated by the transaction currency, which makes it eligible for foreign currency revaluation at month end.

Revenue Reclassification UI Changes

The label for the batch process reclassifying deferred revenue has been changed from Create Revenue Reclassification Journal Entries to Create Deferred Revenue Reclassification Journal Entries.

New Reclassification Reports

The Deferred Revenue Reclassification Activity Report shows a detailed log of all transactions associated with a sales order, including:

- billing transactions
- revenue recognition journal entries

- billing amount allocation journal entries
- foreign currency gain/loss adjustments
- unbilled receivable adjustments

Use this report to reconcile the deferred revenue and unbilled receivable balances on the balance sheet from the prior month to the current month. The report includes sales orders with and without revenue commitments, as both types of orders can change the deferred revenue balance.

The Deferred Revenue Reclassification Report shows the detailed calculation information for line level deferred revenue reclassification. Use this report to audit reclassification activities shown on the Deferred Revenue Reclassification Activity Report. This report also helps forecast the recognition periods of current outstanding deferred revenue balances.

Note: The new reports for deferred reclassification are available in the Reports menu and should be used only after this enhancement is enabled for your account.

Adopting the Deferred Revenue Reclassification Enhancements

When your account is upgraded to Version 2013 Release 2, the deferred revenue reclassification enhancements *are not enabled* in your account. Take advantage of the following options to become familiar with the new reclassification automation.

- Test the new deferred revenue reclassification enhancements in your Release Preview account to become familiar with the new functionality. These enhancements *are enabled* in your Release Preview account.
- Contact Support to enable the enhancements in your NetSuite Sandbox account for further testing.
- Work with the Professional Services team for in-depth training.

After you have fully tested the new line level deferred revenue reclassification process, contact Support to enable the enhancements in your production account.

Important: All accounts using the Revenue Commitments feature are required to enable the new deferred revenue reclassification enhancements before Version 2014 Release 2.

When the new deferred revenue reclassification enhancements are enabled for your account, all existing open transactions with revenue commitments created since Version 2012 Release 2 will be automatically adjusted to follow the new reclassification calculation logic the first time you create revenue recognition journal entries.

Variable Revenue Recognition Schedule Changes

With Version 2013 Release 2, the Variable Revenue Recognition Schedule has been enhanced to provide revenue forecast information for fixed bid projects. Previously, the variable revenue recognition schedule showed only the actual recognized revenue as posted by the revenue recognition journal entries. It did not forecast the amount of revenue to be recognized in future periods.

The Revenue Recognition Schedule now shows both estimated revenue amounts for future periods and actual recognized revenue amounts posted by revenue recognition journal entries. The schedule itself has not changed. No new fields have been introduced to the schedule.

Note: The Variable Revenue Recognition Schedule is available if you use Advanced Projects and create fixed bid projects. Fixed bid projects have a project total amount based on the planned time entries for the project. Forecasted revenue amounts for the project are based on the planned time for the project.

These changes apply only to the Variable Revenue Recognition Schedule. Variable amortization schedules do not include forecast amounts.

NetSuite recalculates the forecasted revenue amounts on the Variable Revenue Recognition Schedule when you run Create Revenue Recognition Journal Entries or when you make changes to an invoice with a variable schedule.

Changes to the Variable Recognition Schedule

On the Variable Revenue Recognition Schedule, each revenue recognition line displays the details for Date Executed, Journal, Amount, Total Recognized, % Amort. in Period, and % Total Amort. Forecast revenue amounts for a period display only the Amount and % Amort. in Period information. The forecast amount is calculated from the estimated time entries listed on the Time Tracking subtab on the Project record.

Revenue recognition forecast amounts can be calculated once a project is attached to a sales transaction or to an item on a sales transaction. When you initially create an invoice for a project, the revenue recognition schedule amounts for a project are calculated as follows:

Total Revenue Amount x (Estimated Total Time Entries for the Period / Estimated Total Time for the Project)

When you run Create Revenue Recognition Journal Entries at the end of a period, the process:

- Creates the revenue recognition journal entry for the variable revenue recognition schedule. This amount is calculated as Current Project Completion amount minus the amount of project completion as of the last time revenue recognition was run.
- Updates the estimated or forecast revenue amounts for future periods on the variable revenue recognition schedule based on the planned time for the project

Forecast revenue is calculated as follows: Remaining Unrecognized Revenue of the Project * (Estimated Remaining Time Entries for a Future Period / Total Remaining Time Entries for All Future Periods)

If you enable the Use System Percentage of Completion for Schedules accounting preference, the forecast and revenue recognition amounts are based on actual time entered for the project. You can manually enter the percent complete for a project if you do not enable this preference.

Customizing Company Records

Previously, users could customize Subsidiary records by adding custom fields and user event scripts. It was not possible to customize the Company record. This meant that you could not adapt the Company Information page to reflect the country where your business is located. In OneWorld accounts, you could customize only subsidiary records.

With Version 2013 Release 2, you can now add custom fields to the Company record in both single instance and OneWorld accounts. In OneWorld accounts, you can also customize subsidiary records. Customizing company and subsidiary records enables you to add fields that are specific to the country or region in which your business operates.

To add custom fields:

1. Go to Setup > Customization > Other Custom Fields.
2. In single instance accounts, select Company as the Record Type.
In OneWorld accounts, select Subsidiary for Record Type.
3. Complete the remaining information as needed to create the custom field.

In addition, with this release, the layout of the Company Information page has changed. Company address fields now appear on the new Address subtab, next to the Shipping Address and Return Address subtabs.

Note: In OneWorld accounts, many fields on the Company record are copied to the root subsidiary record. You can only edit these fields on the Company record.

For more details about customizing company and subsidiary records, see the help topic [Creating a Custom Field](#).

Tax Period Posting Changes

Tax periods are applicable to non-US and non-Canada nexuses only. The use of tax periods facilitates audits, allows you to close individual periods, and enables you to run tax reports whether monthly, quarterly, or annually.

Typically, users set up months (base periods) as their lowest tier periods for tax reports. The aggregate nodes (period groupings) such as quarters and years are typically used for reporting purposes only. However, there are some users who only set up quarters or years as their lowest tier periods (Setup > Accounting > Manage Tax Periods). Version 2013 Release 2 is intended to address this practice in order to make tax periods compatible with the rest of our systems.

As before, when setting up tax periods, the recommended best practice is to use months for posting tax, even if you only intend to run quarterly or annual reports. This can be done by going to the Manage Tax Periods page, selecting the Set Up Full Year option, and then choosing the Calendar Months format.

If you are currently using quarters or years as your lowest tier of tax periods instead of months, or if you have set up periods in mixed mode, be aware that as of Version 2013 Release 2, a new Is Posting check box has been added to the tax period record for quarters and years. You will be required to define the quarter or year as either posting or non-posting. Review your tax period setup in your beta account or as soon as your account is upgraded to Version 2013 Release 2. If there is a quarter or year without children (months or base periods), and you want to use it for tax reporting, then you must make sure that the Is Posting box is checked on the tax period record. On the Manage Tax Periods page, you can identify posting periods by their status icons. Non-posting periods do not have status icons.

Note: If you set up quarters without months, you will only be able to run quarterly tax reports. If you set up years without children, you will only be able to run annual tax reports. Consequently, you will not be able to run monthly tax reports in the event of internal or external needs such as tax audits, unless you have tax periods set up as months.

Mixed mode posting means that within the same period, there are transactions that were posted both to a period and to its parent period. For example, if your tax period June 2013 is a child of the period Q3 2013, and you did not set up July and August tax periods, then when you close the June 2013 tax period, future transactions during the same quarter will be posted to the Q3 2013 tax period. Transactions with a July or August date will be posted to Q3 2013. Any June 2013 transaction posted after the closure of the June 2013 tax period will also get reported in the Q3 tax period.

From Version 2013 Release 2 onwards, it will no longer be possible to add children to a posting period. Hence, the ability to set up mixed mode periods has been discontinued. However, existing mixed mode periods will remain posting. The system looks at the last mixed mode posting date and determines that after that date, it is no longer possible to create mixed mode periods. Before this last mixed mode posting date, any period with transactions already posted to it will remain posting, and any period that does not have children will remain posting.

If you have been using posting quarters, we recommend that you switch to months for your future periods. Only use posting quarters or posting years if you are sure that you do not need months.

If you want to set up new posting tax quarters, you can use any of the following methods:

- Choose Set Up Full Year and select Quarters as the period format. Those quarters are defined as posting periods by default.
- Choose New Quarter Only and check the Is Posting box on the tax period record.
- Edit a non-posting quarter and check the Is Posting box. Note that you can only do so if the quarter has no months or base periods.

When editing tax periods, bear in mind the following changes in Version 2013 Release 2:

- You can change a period from posting to non-posting only if there is no transaction associated with it.
- You cannot change a period from non-posting to posting if it has sub-periods.
- You cannot make months (base) periods non-posting because months (base) periods can only be posting. They do not have the Is Posting box.
- You cannot delete a closed period that has transactions associated with it. However, you can change its parent period in case you need to reorganize your base periods. You can also edit the closed period's name and Is Posting box.

As with months and base periods, you can also close posting quarters and posting years on a per subsidiary basis. On the Manage Tax Periods page, double-clicking a period's status icon displays the Close Tax Period page for that period. To close the period, choose the subsidiaries to which the closed period applies, and then click Submit. Closing a tax period for a subsidiary closes prior tax periods for that subsidiary. Opening a closed period for a subsidiary also opens future tax periods for that subsidiary.

Note that for all existing mixed modes, both the parent and children are posting, and both can be closed on a per subsidiary basis. If you close all child periods of a subsidiary, and the last child has the same end date as the parent, then the parent period will also be closed for that subsidiary. It is advisable to close the tax periods for which you intend to generate a tax report, in order to prevent further postings to past tax periods that have already been filed.

The tax period assigned to a transaction is determined by the transaction date (applicable only if tax reporting is NOT set to cash basis). The system assigns the tax period in which the transaction date belongs, provided that the tax period is open. If the transaction date is within a tax period that is already closed, the system assigns the next open tax period. For example, if you have a transaction dated June 30, 2013 and your June 2013 tax period is either closed or does not exist, then the transaction is posted to the next tax period that is open, such as July 2013.

If you have overlapping posting tax periods that were set up prior to Version 2013 Release 2, the system chooses the lowest level period as the posting period if both periods are open. For

example, if you have a transaction dated June 1, 2013, the system will choose June 2013 as the posting tax period for this transaction if both June 2013 and its parent Q3 2013 are posting and open.

As of Version 2013 Release 2, the order of assignment is by start date, and then by the lowest tier periods. If there are no months set up, then the transaction will be assigned to a quarter, and if there is no quarter, then it will be assigned to a year, provided that they are posting.

In addition, the ability to create adjustment tax periods has been removed. Previously, it was possible to set up a year with a one-day adjustment tax period, and it was also possible to define a tax base period as an adjustment period. In Version 2013 Release 2, the One-day year-End Adj. Period box and the Period is Adjustment box are no longer available. Note that if you already have adjustment tax periods set up in your account before Version 2013 Release 2, those periods are retained, and the checked Period is Adjustment box is displayed on the period page as read-only.

Important: The use of quarters or years as the lowest tier posting periods is not yet compatible with the Multiple Calendars feature. If you enabled Multiple Calendars in an earlier release (pre-Version 2013 Release 2), you will not be able to create new posting quarters for tax. Hence, it is also not possible to enable Multiple Calendars if your quarter or year tax periods have the Is Posting box checked.

Attaching Line Item Receipts to Expense Reports

You can now upload attachments for individual line items on expense reports, allowing for easier and more precise receipt tracking. Previously, files were uploaded and attached on the Files subtab of the History tab.

A new Attach File column has been added to the Expenses tab. When you enter expenses you can attach your receipt directly to the line item by uploading a file from your computer or by selecting a file from the File Cabinet.

By default, the Attach File column appears on the right side. You can customize your expense report form to change the name and position of the column.

To attach a receipt to a line item, enter your expense information. In the Attach File column, select List to select your receipt from the File Cabinet, or select New to upload a new receipt. When you have finished, click Add. You can attach the same receipt to multiple rows.

Expense Report Line Item Reference Numbers

Expense reports now have a reference number for each line item entered. Previously, there were no visible reference numbers on individual expense report line items. Line item reference

numbers allow you to track entered receipts more precisely by associating the number with scanned receipts.

By default, the Ref No. column appears at the beginning of each line. You can customize your expense report form to change the name and position of the column, or to hide the column entirely.

Each reference number must be unique for the expense report on which it appears. On a new expense report, the reference number column begins at one by default. You can edit the reference number of any line, and the next new line will default to one more than the highest reference number.

Quantities and Rates for Expense Categories

You can now designate rates for expense categories. This capability supports quantity-based calculations for expense report line items.

For example, you can designate the rate at which you reimburse employees for mileage in the mileage expense category. When an employee enters their mileage on an expense report, they only need to enter the number of miles and the amount is automatically calculated based on the given rate.

Previously, only an amount was available to enter on expense reports, requiring calculations to be done prior to submitting expenses for reimbursement.

To add a rate to an existing expense category:

1. Go to Setup > Accounting > Expense Categories.
2. Click Edit next to the category you want to change.
3. Check the Rate is Required box, enter a rate in the Default Rate field, and when you have finished, click Save.

Note: If you use subsidiaries in NetSuite, a Rate subtab allows you to set a rate for each subsidiary utilizing this expense category.

When you are completing an expense report, if you choose any category with a defined rate, both the Rate and Quantity fields are added to the available entry fields for expenses.

General Shipping Enhancements

Version 2013 Release 2 includes the following enhancements to Shipping features:

- [Shipping Methods Per State](#)

- Exclude from Shipping
- Parcel Select for USPS
- UPS Shipping Enhancements

Shipping Methods Per State

For web store orders based in the United States, you can now restrict a shipping item to be used only for delivery to one or more states. For example, if you ship orders to destinations outside the continental United States, you can set up a shipping item to be used only for delivery to Hawaii.

This shipping item restriction is available only for web store orders shipping from the United States.

The Shipping and Handling Rules subtab of the Shipping Item record has a new section called Web Store Shipping Restrictions that groups the new shipping restriction for states, and also the existing Ship Only To restriction for countries.

To restrict a shipping item for use by state:

1. On the shipping item record, go to the Shipping and Handling Rules subtab.
2. In the States field in the Web Store Shipping Restriction section, select one or more states to ship to using this shipping item.

If you do not select any states, the shipping item can be used to ship to all states

Exclude from Shipping

A new check box, **Exclude Item from Rate Request**, is now available on the sales order form. Check this box for any item you want to exclude from the shipping rate calculation. The shipping rate total will not include this item. The check box is available only for inventory items and assembly items. The setting carries over to the item fulfillment record. If you recalculate shipping from the item fulfillment record, the shipping estimate includes all items.

Parcel Select for USPS

With this release, you can now add Parcel Select as a shipping method for USPS. Parcel Post is no longer supported and has been renamed to USPS Standard Post.

To add Parcel Select as a shipping item, go to Lists > Accounting > Shipping Items > New.

If you previously shipped orders using Parcel Post, do the following:

- Inactivate your existing USPS Parcel Post shipping item.

- Create a new shipping item for either USPS Parcel Select or USPS Standard Post.
- Update the shipping item on all sales orders that are not completely fulfilled and have USPS Parcel Post shipping item.

UPS Shipping Enhancements

Version 2013 Release 2 includes the following enhancements to UPS shipping integration with NetSuite:

- [UPS Rates Type Definitions](#)
- [Changed UPS Pickup Type Options](#)
- [New UPS Rate Code Field](#)
- [Additional UPS Shipping Service Changes](#)

UPS Rates Type Definitions

With this release, you can identify the type of rates to show and charge your customers, including a new option for negotiated rates for your account. A new field, UPS Rates Type, is available on the Preferences subtab of the Set Up Shipping page. Select one of the following types: Standard List Rates, Retail Rates, Daily Rates, and UPS Negotiated Rates.

For new UPS accounts, the UPS Rates Type will be set to Standard List Rates, but can be changed. For customers who have a UPS account set up in NetSuite before Version 2013 Release 2, the UPS Rates Type will be set based on the UPS Pickup type setting on the UPS carrier registration page.

- If the UPS Pickup Type was Daily Rates, the UPS Rate Type will be Daily Rates.
- If the UPS Pickup Type was Occasional, the UPS Rate Type will be Retail Rates.

Customers with an existing UPS account registered in NetSuite should review and select the best UPS Rates Types for their businesses.

Changed UPS Pickup Type Options

In addition, the options for UPS Pickup Type on the carrier registration page have changed. The new options are:

- Daily Pickup — existing option
- On Call Air — replaces Occasional
- One Time Pickup — replaces Suggested Retail Rates

- Customer Counter — new
- Letter Center — new
- Air Service Center — new

Customers with an existing UPS account registered in NetSuite should review the value selected for UPS Pickup Type on the carrier registration page and revise as necessary.

New UPS Rate Code Field

A new field, UPS Rate Code, is now on the UPS carrier registration page. The rate code is generated when you submit the registration page. A new rate code is generated if you update your UPS registration and click Submit. Contact your UPS account representative whenever you have a new rate code and want your negotiated rates to be enabled in NetSuite for your account.

Negotiated rates will work only for this specific combination of Rate Code and Account Number.

Additional UPS Shipping Service Changes

Version 2013 Release 2 provides the following additional changes for UPS shipping services:

- New package types are available with precoded dimensions for Express Box (large, medium, small), Express Tube, Express Pak.
- Dimensions are required when customers use their own packaging. Enter dimensions on the Packages subtab of the item fulfillment record.
- Weight is now required when using UPS Letter Package type.
- Large Package, a new field on the Item Fulfillment, is checked when the total package size exceeds 130 inches (330 cm). Additional charges apply.
- Descriptions for UPS Worldwide Express, UPS Worldwide Express Plus, and UPS Worldwide Saver shipping rates have been updated with more information about delivery time commitment.
- UPS Express Saver service can be used for shipments to, from, and within Europe.
- UPS Domestic service has been expanded to 34 countries. UPS registration is now possible for 21 countries.
- UPS Worldwide Expedited service is available to and from the following Middle East countries: Bahrain, Egypt, Iraq, Israel, Kuwait, Oman, Qatar, Saudi Arabia, and United Arab Emirates.
- UPS Standard service is now available to and from Mexico, in addition to Canada.

- Currency codes have been updated for Slovakia (Euro), and Russia (Rubles).
- Country codes have been updated for the Democratic Republic of Congo (CD), Guernsey (GG), Jersey (JE), Montenegro (ME), and Serbia (RS).
- Postal code formats have been updated for Armenia and Jordan.

Component Yield Preferences

Assembly component preferences are available to assist with component requirement calculations.

Note: To use these component preferences, you must have the Work Orders feature enabled.

To use these component preferences, you need to adjust settings on item records. The following settings are available:

- Use Component Yield – allows you to account for material component yield loss during the ordering and planning process
- Round Up Quantity as Component – rounds up the component quantity to a whole number in base units

For more details about this feature, see the following:

- [Use Component Yield Check Box](#)
- [Round Up Quantity as Component Check Box](#)

Use Component Yield Check Box

The Use Component Yield preference enables you to make quantity allowances for materials lost during processing due to issues such as the following:

- Quality of material
- Machine processing

Accounting for material variances enables you to plan for the usable quantity you need as a result, while ordering a quantity that is higher than what you actually need.

For example, you sell a coffee cup assembly that is made up of 3 components: 1 cup, 1 lid, and 1 sleeve. For every 100 assemblies, 5 cups are defective. Therefore, when you order the cup assembly, to get 100 units, you need to actually order 105.

To set component yield settings, create or edit an assembly item record. On the Purchasing/Inventory subtab of the assembly record, click the Components subtab. Check the Use Component Yield box to enable NetSuite to calculate the necessary quantity to build or order, based on component yield settings.

Note: When this box is checked, this preference applies this setting not only to this assembly, but also to any sub-assembly components required for the top level assembly.

Important: If there are scripts in place to manipulate the quantity field, these scripts will no longer be valid. Please ensure your customization scripts are reviewed prior to enabling this preference. For details, read [Component Yield Preferences — SuiteScript Impact](#).

When you have checked the Use Component Yield box, you can do the following:

- Set the BOM.
- Set the yield.
- Enable NetSuite to calculate the quantity for work orders.

Clear the Use Component Yield box if you do not want NetSuite to calculate the necessary quantity to build/order.

When this preference is enabled, the following changes show on the item record:

- Quantity– The Quantity field on the Components subtab is grayed out and cannot be edited because this quantity is calculated by NetSuite.

This field displays the amount calculated to be required for the assembly when one unit of the assembly is being built. This value is automatically calculated from the entries in the Component Yield and BOM Quantity fields.

- BOM Quantity – The BOM Quantity column displays the quantity of the component used to build an item assuming no loss.
- Component Yield – The Component Yield column displays the anticipated yield due to loss of component during the manufacturing process.

For example, when the Use Component Yield preference is enabled, you can enter the yield and BOM quantity.

Widget 1 is a component of assembly Item A. Each Item A assembly requires a quantity of one Widget 1. Widget 1 has an anticipated loss amount of 50%. Therefore, you enter the following on the assembly item record for Item A:

- Use Component Yield check box = checked
- Widget 1 Component Yield = 2
- Widget 1 BOM Quantity = 1

Using these settings, when a work order is created for Item A, NetSuite calculates a requirement of 2 for the component Widget 1, and displays 2 in the Quantity field on the Components subtab.

Round Up Quantity as Component Check Box

Using this new setting, the quantity for a component on a work order automatically rounds up to a whole number in base units.

For example, you have an assembly that requires 2 units of Component A. The component yield is 99%. To build 5 of these assemblies requires 10.1 units of Component A. Since you can consume components only in whole numbers, you cannot consume 10.1 units. Therefore, you need to round up to the next highest whole number in base units.

On a work order for an assembly that uses component yield, NetSuite indicates the component Quantity, BOM Quantity, and Component Yield for rounded-up components.

To set this preference, create or edit an item record for an inventory item or an assembly used as a sub-assembly component. In the Inventory Management section of the item record, do one of the following:

- Check the Round Up Quantity as Component box to enable NetSuite to round up the quantity consumed for this item.
- Clear this box if you do not want NetSuite to round up the quantity consumed for this item.

Note: When this preference is enabled, the Quantity field on the Components subtab is grayed out and cannot be edited because this quantity is calculated by NetSuite.

Costed BOM Inquiry

In Version 2013 Release 2, use the new Costed BOM Inquiry to examine the standard cost breakdown of an assembly. The Costed BOM Inquiry is a report that enables you to view how different cost components are rolled up to an assembly item based on the bill of materials (BOM) structure.

For example, a cost accountant in a manufacturing environment may want to view the different cost components associated with building an assembly. They also want to see how each of these components are calculated and rolled up to the finished goods. The Costed BOM Inquiry

can assist with these tasks. It also allows visibility for variances between inventory values at a given point in time (as displayed on the Inventory Valuation reports) and the disaggregated component costs of the assembly.

Costed Bill of Materials Inquiry

Assembly: Finished Goods 99787 Date: 6/7/2013
 Location: Warehouse - East Coast Total Unit Cost: 7,543.8
 Top Level Only: ☐ Routing Unit Cost:
 BOM Display Control: By Date

Customize

Component Name	Level	Component Yield	BOM Quantity per Assembly	Quantity per Assembly	Quantity per Top Level Assembly	On Hand	Available	Back Ordered	On Order	Unit Cost	Cost
Sub Assembly - 90876	1			4	4	0	0	0	0	1,464.95	5,859.8
Tubing 98999654	2			1	4	0	0	0	0	100	400
Plastic Part 110-222	2			2	8	0	0	0	0	16	128
Rim	2			3	12	0	0	0	0	6	72
Screws	2			4	16	0	0	0	0	300	4,800
10mm X 40mm Stainless Hex Bolt	2			5	20	0	0	0	0	22.99	459.8
Rod 987463	1			1	1	0	0	0	0	696	696
Casing 98878	1			1	1	0	0	0	0	988	988

Note: You must have the Standard Costing feature enabled to use the Costed BOM Inquiry. This inquiry can be run only for a standard cost assembly item. Component costs are displayed only for standard cost components. Non-standard cost components will be displayed with a cost of zero.

Roles that can view this inquiry are:

- Accounting Center
- Shipping Center

On the inquiry, once you select the subsidiary, location, and assembly, NetSuite displays the cost breakdown of the assembly and its member components. This assembly cost is calculated based on the standard cost of components.

For assemblies that have a conversion cost, the assembly cost is calculated as follows:

- The component cost is based on the component cost displayed in the inventory revaluation transaction.
- The conversion cost is based on the assembly inventory revaluation transaction.

For assemblies that do not use routing, the assembly cost is based on the component cost displayed in the inventory revaluation transaction.

To run a Costed Bill of Materials Inquiry, go to Transactions > Manufacturing > Costed Bill of Materials Inquiry.

New Bulk Process Status Pages

Now when you submit the following transactions for bulk processing, a new status page displays the progress of the transactions being processed:

- Revalue Standard Cost Inventory
- Build Work Orders

Process Status

Notice:
Since Release 2013.1, we modified the new Revalue Standard Cost Inventory status page to access legacy status page for previously processed items. [Click Here](#)

Submission ID #	Process Type	Submission Status	Percent Complete	Message	Date Created	Created By
7	Revalue Standard Cost Inventory	Complete	100.0%	0 Errors	7/6/2013 10:19 am	Jonathan Wolfe
16	Revalue Standard Cost Inventory	Complete	100.0%	0 Errors	7/6/2013 10:22 am	Jonathan Wolfe
12	Revalue Standard Cost Inventory	Complete	100.0%	0 Errors	7/6/2013 10:29 am	Jonathan Wolfe
15	Revalue Standard Cost Inventory	Complete	100.0%	0 Errors	7/6/2013 10:30 am	Jonathan Wolfe
17	Revalue Standard Cost Inventory	Complete	100.0%	0 Errors	7/6/2013 10:30 am	Jonathan Wolfe
20	Revalue Standard Cost Inventory	Complete	100.0%	0 Errors	7/6/2013 10:30 am	Jonathan Wolfe
22	Revalue Standard Cost Inventory	Complete	100.0%	0 Errors	7/6/2013 11:00 am	Jonathan Wolfe
23	Revalue Standard Cost Inventory	Complete	100.0%	0 Errors	7/6/2013 11:00 am	Jonathan Wolfe
25	Revalue Standard Cost Inventory	Complete	100.0%	0 Errors	7/6/2013 11:00 am	Jonathan Wolfe
26	Revalue Standard Cost Inventory	Complete	100.0%	0 Errors	7/6/2013 11:00 pm	Jonathan Wolfe
48	Revalue Standard Cost Inventory	Complete	100.0%	0 Errors	7/6/2013 11:14 pm	Jonathan Wolfe
36	Revalue Standard Cost Inventory	In Progress	40.0%	0 Errors	7/6/2013 3:14 pm	Jonathan Wolfe

Mark All Bulk Process Work Orders

Now when you are bulk processing work orders and click the Mark All button, NetSuite selects all rows across all page segments for processing. Previously, clicking the Mark All button selected only the orders visible in the currently displayed segment.

Units of Measure Validation on Transactions

In Version 2013 Release 2, when you save a transaction that uses a Unit of Measure (UOM), NetSuite performs a check to verify that the UOM settings are correct. An error message may alert you about items on the transaction that have an invalid setting.

Standard Cost Material Overheads

In Version 2013 Release 2, you can use a material overhead cost category to track and manage overhead associated with a product for a standard cost item. For inventory received, you can now track material overhead absorption by per quantity basis across several types of overhead such as freight, duty, or storage. This functionality is helpful to monitor product cost variances between actual and budgeted costs, and to avoid errant fluctuations of overhead costs during your procurement process.

For example, you have a storage cost associated with a product you sell. Whenever this item is received, you can allocate \$50 dollars of storage cost to be incorporated into the item cost.

An Inventory Cost template is available to enter and store the amount and details of the materials overhead which allows your accountant to do the following:

- View materials overhead information in the Planned Standard Cost record

- View materials overhead information in an assembly rollup

These material overheads and landed cost may be defined as one of the following:

- Fixed cost per base unit
- % of material cost
- % of total cost (material + material overhead)

To use standard Cost Material Overheads, complete these three steps:

1. Create appropriate Other Charge items.
2. Link the Other Charge item to an inventory cost template.
3. Apply the template to an inventory item.

Note: This functionality requires that the Standard Costing feature is enabled in your account.

Demand Planning and Inventory Allocation

In Version 2013 Release 2, when processing backorders, NetSuite helps automate your process for allocating anticipated future inventory when you use customer priority settings. To maintain strategic customer relations, you may choose to prioritize order processing to favor some customers, such as those who comprise a significant portion of your sales. NetSuite enables you to set up order processing to ensure that inventory is reserved for timely fulfillment of priority customer orders rather than simple first-come, first-served fulfillment.

Backordered items can be shipped based on the commitment priority you designate. Rather than fulfilling orders based on a transaction date or expected ship date sequence, employees who manage sales orders in your pipeline can use this prioritization option to reallocate the quantity committed to orders based on designated priority.

First, assign priority numbers for customers. Then, NetSuite calculates the expected ship dates for sales order items based on priority standing and when the inventory is expected to arrive in your warehouse. NetSuite examines the quantity on hand or uncommitted quantities and can automatically prioritize planned sales orders and transfer orders based on their importance.

To set up your account and use enhanced inventory allocation and customer prioritization, use the information below:

- To use enhanced inventory allocation, verify that these features are enabled:
 - Inventory
 - Sales Orders

- Demand Planning

Read the help topic Enabling Features for Enhanced Allocation.

- Set the following preferences:
 - Item Commitment Transaction Ordering
 - Perform Item Commitment After Transaction Entry
 - Default commit option on sales order
 - Default commit option on work order
 - Default commit option on transfer order

Read the help topic Inventory Allocation Preferences.

- Set customer priorities. Read the help topic Customer Commitment Priorities.
- Use the Commit Order form to allocate items to orders. Enable the Perform Item Commitment after Transaction Entry preference and NetSuite will automatically allocate items based on the set priority. The order with the highest priority will have items committed for fulfillment. Automatic calculation of quantities to be committed to prioritized orders can save you time and effort. Read the help topic Commit Orders.

Note: The Commit Orders page functions differently than the Reallocate Items page in the following way:

NetSuite automatically calculates the quantities on the Commit Orders page. You must manually update the quantities on the Reallocate Items page.

- Reallocate open quantities from one order to another. You can use the Reallocate Open Quantities function to change which orders items are committed to for fulfillment. NetSuite examines orders for item quantities that are committed but not firmed or fulfilled yet. These items are eligible to be reallocated to commit them to another order for fulfillment.

For example, a new order is entered that is a higher priority to fulfill than existing open orders. You can reallocate open quantities of items committed on existing orders to fulfill the highest priority order.

- Sales Order #1 is entered and is assigned a low priority.
- Items are committed to Sales Order #1 using the Commit Orders page.
- Sales Order #2 is entered and is assigned a high priority.
- The customer service rep prefers to fulfill order #2 before order #1. He wants to undo the commitment of items to Order #1 and commit those items to Order #2.

To reallocate items to different orders, check the Reallocate Open Quantities box on the Commit Orders page. Then the page shows all sales orders, work orders, and transfer orders that include committed quantities not yet picked, packed, or shipped. This list is sorted by transaction date by default. Click a column header to change the sort order.

Note: Orders that are firmed do not show in the list.

Landed Cost Allocation Per Line

Manufacturers may purchase goods many places, and it is increasingly important to track all costs associated with acquiring materials, including landed costs such as freight, duty, and taxes. In Version 2013 Release 2, NetSuite includes the Landed Cost Allocation per Line preference to enable distributors to track landed cost more closely when they are receiving items.

This preference gives you the ability to track multiple landed cost categories within a specific line item on the item receipt. You can also automatically distribute the cost, based on an attribute such as weight or value within a transaction.

When the Landed Cost Allocation per Line preference is enabled, you can enter the landed cost allocation amount on a per-line basis, rather than distributing the cost based on an allocation method at a per receipt level basis.

Use Landed Cost Allocation per Line as described in the following sections:

- [Setup for Landed Cost Allocation Per Line](#)
- [Landed Cost Allocation Per Line on Transactions](#)

Setup for Landed Cost Allocation Per Line

- At Setup > Accounting > Accounting Preferences, enable the preference called Landed Cost Allocation per Line.

Note: This preference is available if the Landed Costs feature is enabled in your account.

- Under Landed Cost Allocation per Line, select the default setting.

Note: When enabled, all new item receipts, credit card charges, vendor bills, and checks default to use landed cost per line. You can still change this setting when adding a new transaction.

Landed Cost Allocation Per Line on Transactions

- On a purchase order, the item list includes a new check box called Landed Cost per Line.

- Check this box to enter a landed cost for individual lines.

Note: When this preference is enabled, the landed cost subtab is disabled and allocation is not used.

- Clear this box if you are not entering a landed cost for individual lines.

Note: When this preference is disabled, the landed cost subtab is used.

Important: This setting cannot be changed once an item receipt is recorded against the purchase order.

- On item receipts, credit card charges, vendor bills, and checks, the item list includes a new check box called Landed Cost per Line. This check box is available in Add mode.

When you enable Landed Cost per Line on a transaction, you will see an icon for items that tracks landed cost. Clicking the icon opens a landed cost entry window. Click the add multiple button to enter multiple landed costs at one time. Landed cost data you enter is saved as a subrecord of the transaction.

Projects

Version 2013 Release 2 includes the following enhancements to Projects features:

- [Expanded Task Scheduling for Advanced Projects](#)
- [Milestone Records for Advanced Projects](#)
- [Gantt Chart Enhancements](#)
- [Billing Class Enhancements for Advanced Projects](#)
- [Project Center](#)
- [Project Task Manager \(formerly Resource Manager\) Enhancements](#)

Expanded Task Scheduling for Advanced Projects

Advanced Projects now includes additional task scheduling options. You can create project tasks with multiple predecessors and finish-to-finish (FF) and start-to-finish (SF) relationships. In addition, project tasks can now be created with lag time between tasks and a Finish No Later Than (FNLT) constraint.

Prior to Version 2013 Release 2, Advanced Projects only offered finish-to-start (FS) and start-to-start (SS) predecessor relationships. This required manual manipulation of task dates in order to include additional days and limited the relationship between adjacent tasks.

The following predecessor relationships are now offered when creating project tasks:

- finish-to-start – Task starts when preceding task finishes. Start date is adjusted based on the preceding task's finish date.
- start-to-start – Task starts after preceding task starts. Start date is adjusted based on the preceding task's start date.
- start-to-finish – Task finishes after the preceding task starts. Start date is adjusted based on the preceding task's start date.
- finish-to-finish – Task finishes after the preceding task finishes. Start date is adjusted based on the preceding task's finish date.

When you are creating a project task, on the Predecessor subtab, you can add one of these new predecessor types by selecting it in the Type field.

Lag time is a delay between tasks that have a dependency. You can enter lag time to adjust your project schedule. To enter lag time, enter the number of days in the Lag Days field on the Predecessor subtab.

Adding a Finish No Later Than (FNLTL) constraint to a project task ensures that the task cannot finish after the selected date. To set a FNLTL constraint, on the project task record, under Project Task Dates, select the desired date in the Finish No Later Than field.

Note: A constraint takes precedence over task relationships, and start dates are adjusted according to the fixed end date of a task with a FNLTL constraint.

Milestone Records for Advanced Projects

With Advanced Projects, you can now create project milestone tasks in your project plans. Previously, you could create a project task with no hours to serve as a milestone but there was no way to differentiate the task as a milestone task.

Project milestones are used to mark a point in your project, usually completion of a set of tasks or as a project health check to determine if you're on schedule.

Project milestones cannot have estimated hours, assignees, or a Finish No Later Than (FNLTL) constraint. On a project Gantt chart or schedule, milestone tasks are differentiated from regular project tasks.

To create a project milestone:

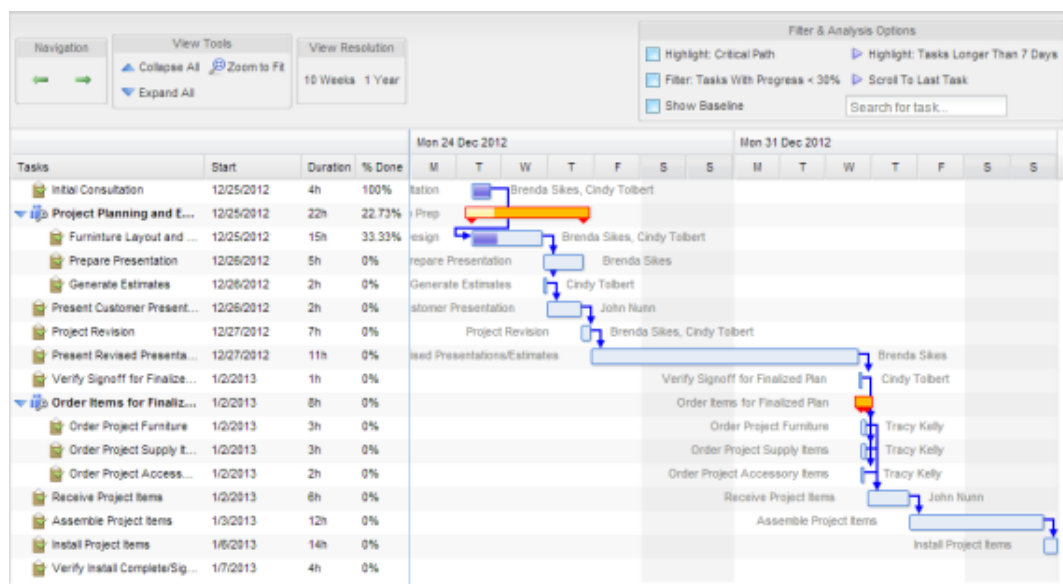
1. Go to Lists > Relationships > Projects.
2. Click the name of the project you want to edit.

3. On the Schedule tab, click New Milestone.
4. Enter a name for your milestone.
5. Click the Predecessors tab.
6. Select a predecessor task and type.
7. Click Add.
8. Continue adding any additional predecessors, and when you have finished, click Save.

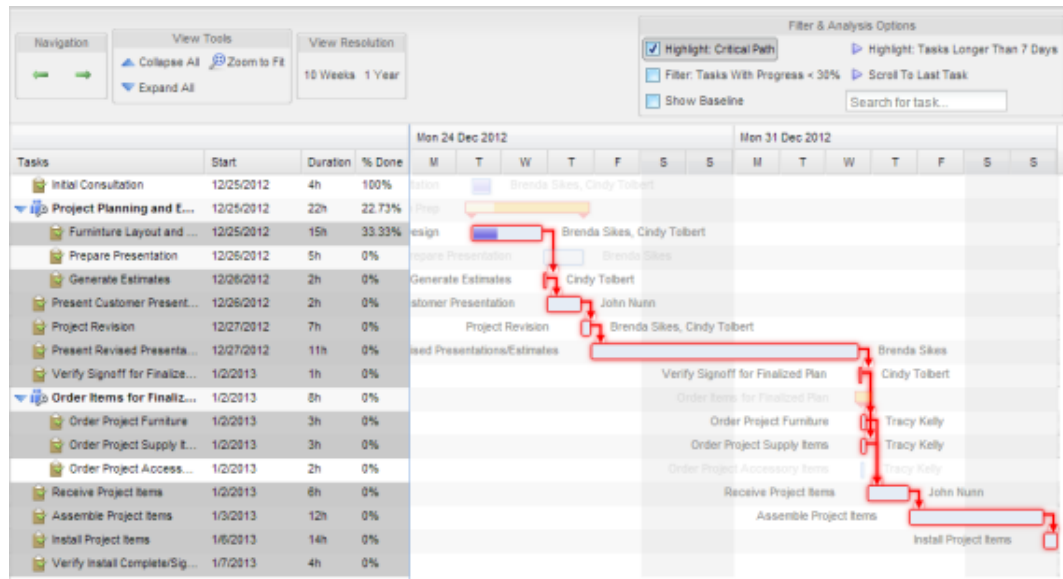
You can convert a milestone to a project task by adding hours or tracking time against the milestone. A Project task can also be converted to a milestone by removing estimated hours, assignees, and any FNLTL constraint.

Gantt Chart Enhancements

With Version 2013 Release 2, NetSuite offers a fully redone graphical Gantt chart. This chart uses a new visualization library that gives project managers more visibility into their project schedules. Previously, a static Gantt chart when you clicked Gantt Chart from the project record.



The improved Gantt chart offers a new interface that allows zooming and scrolling through the project schedule, and highlighting important project metrics, such as the critical path.



Billing Class Enhancements for Advanced Projects

Billing classes are now available to use with project resources. Previously, billing classes were not available to be associated with a resource on a project task.

Using billing classes with projects allows you to create different roles that resources play on a project, for example, *Consultant*, *Analyst*, or *Project Manager*. Billing classes can then be used to create different rates for each of these roles, depending on the service or item that is being offered.

To create a billing class:

1. Go to Setup > Accounting > Billing Classes > New.
2. Enter a name and description for the billing class.
3. Enter a price for this billing class.

If you use multiple currencies, you can enter a different default price for each currency.

4. When you have finished, click Save.

Once you have created your billing classes, you can assign each employee a default class in the Classification section of the employee record. You can also assign a default billing class to vendors that serve as project resources on the Financial tab of the vendor record. When a resource with a billing class is assigned on a project task, the price for that resource is calculated from the billing class price.

Billing classes can also be assigned to service items for sale and resale. When you assign a billing class to an item, you can define a different price structure that applies only to that service item. When a service item is also used on a project task, the price is calculated based on the service item billing class and will override the default billing class price.

Note: Billing Class Enhancements for Advanced Projects requires both Advanced Projects and Per-Employee Billing Rates features to be enabled.

Project Center

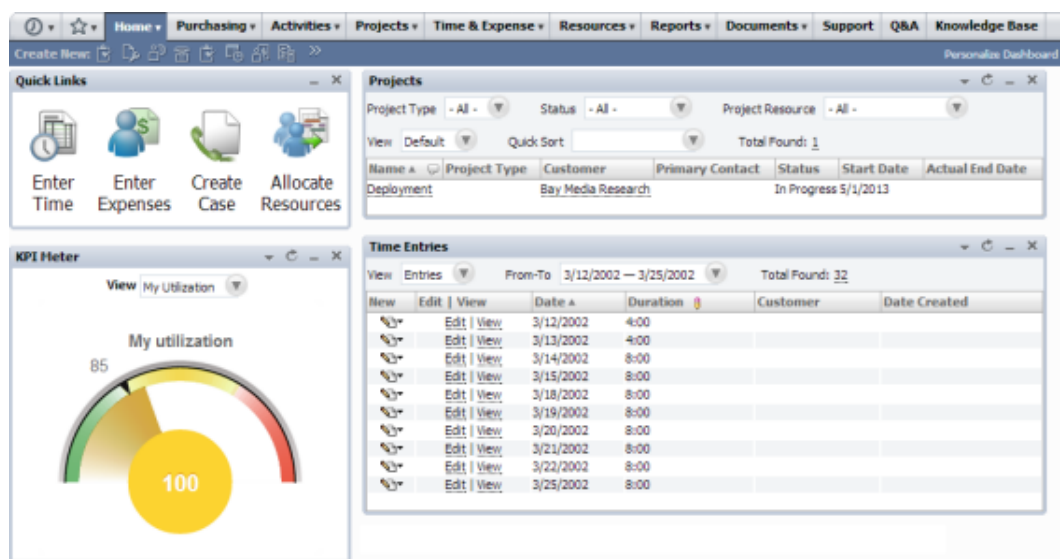
NetSuite now offers a standard Consultant role with access to a new Project Center. The Project Center allows access to NetSuite through a specialized interface that routes the most important functions of project work directly to the home page.

Prior to Version 2013 Release 2, there was no standard center that offered a project specific interface. A custom center was required to give project consultants and managers specialized access to project information.

Important: Advanced Projects is required to use the Consultant role and the Project Center. For more information, see the help topic Advanced Projects.

The Project Center has the following tabs by default: Home, Activities, Projects, Time & Expenses, Reports, Documents, and Support. Each tab offers access to links and information that deal directly with project management in NetSuite.

The Home tab contains a variety of at-a-glance portlets for easy access to the most used tasks and information for projects. Included on the Home page are Quick Links for entering time and expenses, creating a case, and allocating resources.



The Projects portlet contains a list of assigned projects with links to view the project record and additional information. The My Timesheets portlet offers a list of timesheets for a given time period including the status of each sheet. A specialized Utilization KPI Meter offers a visual representation of current utilization.

To give employees access to the Project Center, add the Consultant role to their employee record. For information on giving access to employees, see the help topic [Giving an Employee Access to NetSuite](#).

Site Builder

Version 2013 Release 2 includes the following changes:

- [Google Checkout Retirement Impact on NetSuite Web Stores](#)
- [Alternate Payment Methods for Web Store Orders](#)

Google Checkout Retirement Impact on NetSuite Web Stores

Google Checkout is scheduled for retirement by Google in November 2013. As a result, in September, 2013, NetSuite will begin a phased approach to removing Google Checkout integration. Read the [official announcement from Google](#), which includes end-dates for charging, shipping, and refunding orders taken via Google Checkout.

Timeline for Changes in NetSuite

You will notice a series of changes in NetSuite leading up to the end of support for Google Checkout. Details are listed below:

September 2013

- The Google Checkout button will be removed from all web sites powered by NetSuite. Note that orders in NetSuite which have been authorized by Google Checkout will still be able to be processed after the Google Checkout button is removed.
- It will not be possible to create new Google Checkout transactions.
- Google Checkout will be removed from the list of features on the Enable Features page.

November 2013

- All tasks related to Google Checkout will be removed from the NetSuite UI.

May 2014

- Google Checkout online help topics will be removed.

Action Required - Updating your Web Store

NetSuite recommends that you remove the Google Checkout button from your web store by August 30, 2013. Note that the Google Checkout button will be removed in September, 2013, even if you have taken no action.

You can remove the Google Checkout button by disabling the feature in your NetSuite account.

To disable the Google Checkout feature in NetSuite:

1. Log in to NetSuite.
2. Go to Setup > Company > Enable Features.
3. Click the Web Presence Tab.
4. Clear the Google Checkout box.
5. Click Save.

When the Google Checkout button is no longer displayed in your web store, shoppers will not be able to create new transactions with Google Checkout. NetSuite recommends that you also remove any customizations to your web store email messages, or to scriptable cart, that may involve Google Checkout.

If you need assistance in locating or removing web site or SuiteScript customizations, please contact NetSuite Customer Support.

Enhancement Voting for Payment Products

In NetSuite, Google Checkout will not be immediately replaced by another Google or virtual wallet product. Since there are many new, competing payment products on the market, submit an enhancement request for the product integration you would like to see, or vote for one that has already been submitted, such as:

- Google Wallet Instant Buy Integration – Enhancement #254715
- V.me Wallet Integration – Enhancement #238912
- MasterCard Wallet Integration – Enhancement #238913
- Amazon Payments – Enhancement #215818

To vote for an enhancement, log in to your NetSuite Account Center, click the SuiteIdeas tab, and then click the link to launch the SuiteIdeas site. You can enter the Enhancement number in the search box on the SuiteIdeas site, and cast your vote.

Alternate Payment Methods for Web Store Orders

In Version 2013 Release 2, you can display a non-credit card payment method on your web store. The payment method you create in NetSuite is available on your Site Builder and your SuiteCommerce Advanced site. Online shoppers can select the payment method that you set up as an alternative to credit card payment, and then complete checkout without entering credit card information. When the order is submitted on the web site, the sales order created in NetSuite has the corresponding payment method selected. For more information, see the help topic [Creating an Alternative to Credit Card Payment Methods for Web Store Orders](#).

Warning: Before you expose non-credit card payment methods on your web site, make sure you have set up connections to other appropriate payment gateways, or you have set up a sales order processing workflow to collect payment for inventory shipped.

SuiteCommerce Advanced

Version 2013 Release 2 includes the following enhancements to SuiteCommerce Advanced features:

- [Sitemap Generator for SuiteCommerce Advanced](#)
- [Reference Implementation Enhancements](#)

Sitemap Generator for SuiteCommerce Advanced

In Version 2013 Release 2, the Sitemap Generator has been enhanced for SuiteCommerce Advanced sites. Now, the Sitemap Generator spiders all facets and views.

Reference Implementation Enhancements

For release notes describing recent versions of the ShopFlow, Checkout, and My Account Reference bundles see the help topic [Reference Implementation Release Notes](#).

SuiteCommerce Platform

SuiteCommerce Platform includes APIs and other platform capabilities you can use to build and deploy a better and more interactive and engaging shopping experience for customers. Version 2013 Release 2 includes the following enhancements to SuiteCommerce Platform:

- [New Item Search API Input Parameter for Bypassing the CDN Cache](#)
- [Field Set Fields for Accessing Item Quantity On Hand Per Location](#)
- [New Commerce API Methods](#)

New Item Search API Input Parameter for Bypassing the CDN Cache

Note: The Item Search API is only available with SuiteCommerce Advanced. SuiteCommerce Advanced is currently in limited availability. For more information, contact your account representative.

In Version 2013 Release 2 a new input parameter is available for the Item Search API. You can use this API to get the latest data updates while bypassing the CDN cache. This new parameter, `nocache`, affects the edge-control header. When the parameter is set with a value of "T," the content volatility of every asset served is set to UNIQUE. Note the sample Item Search API request: `http://www.mystore.com/api/items?nocache=T`. For more information, see the help topic Item Search API Input Parameters.

Field Set Fields for Accessing Item Quantity On Hand Per Location

Note: This feature is only available with SuiteCommerce Advanced. SuiteCommerce Advanced is currently in limited availability. For more information, contact your account representative.

If you use multiple locations, and operate a SuiteCommerce Advanced web site, you can use Field Sets to access item quantity and availability per location. Additional Field Sets are available to enable you to use the Item Search API to query the field set field, and then get location information in the JSON response. The following table shows the inventory location fields on the item record, along with the corresponding field set field you can use to return location details.

Field on Locations Subtab of Item Record	Field Set Field for Returning Location Details
Quantity Available	quantityavailable_detail
Quantity Backordered	quantitybackordered_detail
Quantity Committed	quantitycommitted_detail
Quantity On Hand	quantityonhand_detail
Quantity On Order	quantityonorder_detail

The following example shows a query for the quantity back ordered:

`http://www.mystore.com/api/items?quantitybackordered_detail`

Note the JSON response to this query:

```
result = { qtybackordered_detail:{ qtybackordered:10 locations:[ {  
  internalid:1, qtybackordered:6 }, { internalid:2, qtybackordered:4 } ] } }
```

New Commerce API Methods

The following table summarizes updates to existing Commerce APIs and new methods in Version 2013 Release 2.

Method	Note
Updates to Existing Methods	
addItem(item)	The quantity parameter used with this method can be a negative number to allow a web developer to add negative quantities to the cart.
setPayment()	Previously, only credit card payment methods were allowed for web orders. As of Version 2013 Release 2, non-credit card payment methods, such as cash and check, can be made available on an e-commerce site. For non-credit card payment methods, this method does not require the following fields: ccname, ccnumber, expmonth, expyear.
New Methods	
order.setItemOptions(itemlineid, optionMap);	The Commerce API has been enhanced to update the item options for a particular cart row. The item ID is an itemlineid, a combination of item ID and set. This value is included to support multiple quantities of the same item with different item options: for example, shirt size small, and shirt size large. The optionmap can contain any of the valid options for the given item id and does not have to include all of them. For example, if only "size" is included, the option "color" would be unaffected.
estimateShippingCost (address)	<p>Previously, the Commerce API accepted country and zip code, and then the state was later determined from the zip code. This approach worked well for addresses in the U.S. and Canada. In Version 2013 Release 2, this method has been extended so that shipping information can be determined by passing either the state or the zip code.</p> <p>Passing either the state parameter or the zip code is required. You can pass either one or both. State is used for shipping estimation when the zip code is not entered.</p> <p>For addresses in the U.S. and Canada, if only the zip code is specified, then the state is determined from the zip code. When both state and zip code are specified, and the zip code does not match the state, the correct state is determined from the zip code.</p>
setItemExcludedFromShipping(String cartItemId, boolean excluded)	<p>The Commerce API has been enhanced with a new method used to identify items that are excluded from the shipping charge on an order.</p> <p>The new field Excluded From Shipping will be added to the nlobjWebStoreOrderItem. This boolean field contains a True value only when the line item on the sales order is excluded from the shipping charge.</p>

General SuiteCloud Platform

Version 2013 Release 2 includes the following general SuiteCloud platform change:

- NetSuite to Retire smbXML

NetSuite to Retire smbXML

As of Version 2013 Release 2, NetSuite no longer supports data imports using smbXML. Any existing smbXML import jobs will stop functioning as of the Release Preview phase of Version 2013 Release 2 in July 2013. The exact time line depends on your upgrade schedule, which will be communicated to you prior to the release.

Important: NetSuite strongly encourages customers using smbXML to begin transitioning away from smbXML now. There are two methods that can replace your existing smbXML method: CSV Import or SuiteTalk Web Services. Many smbXML customers can choose the CSV Import Assistant to import their data into NetSuite. The CSV Import Assistant has an easy-to-use UI and requires no programming knowledge. smbXML customers can also import data using SuiteTalk (Web services); however, programming knowledge and Web services development are required. This method is more beneficial to customers who seek an automated system-to-system integration model.

NetSuite is discontinuing its support of smbXML for several reasons. smbXML is considered to be “legacy” technology. As such, NetSuite has placed its development resources into enhancing CSV and SuiteTalk data integration capabilities. Both the CSV Import Assistant and SuiteTalk are supported in NetSuite OneWorld accounts; smbXML is not supported in OneWorld accounts. Additionally, the CSV Import Assistant and SuiteTalk now support most of the record import types offered through smbXML.

Note: SuiteTalk currently supports import of all record types supported for smbXML import. For CSV import, any previously unsupported record types are supported in Version 2013 Release 2.

To begin your smbXML to CSV transition process, see the help topic [CSV Imports Overview](#) help topic to learn about the CSV Import Assistant .

To learn more about SuiteTalk, see the help topic [SuiteTalk Platform Overview](#).

If you have issues during your transition process, please contact NetSuite Technical Support.

SuiteBundler

Version 2013 Release 2 SuiteBundler features are designed to enhance bundle development in sandbox accounts. These features aim to make the release and update of bundles from sandbox accounts into their related production accounts a smoother process. They include the following

- [Selective Update of Sandbox Bundle Objects](#)
- [Restoring Bundle Definitions after Sandbox Refresh](#)

- Dissolving Sandbox Bundles

Important: Bundles are also known as SuiteApps.

Bundle Support During Release Phasing

Bundle support during phasing of Version 2013 Release 2 is consistent with the support provided during previous releases. Review the description below for a reminder of how bundles from different versions are handled during release phasing.

During phasing of Version 2013 Release 2, some accounts are still using Version 2013 Release 1, while other accounts have been upgraded to Version 2013 Release 2, so you may be using a different version than some of the accounts where your bundles are installed.

- Bundles developed with Version 2013 Release 1 can be installed into accounts that are already using Version 2013 Release 2.
- Bundles developed with Version 2013 Release 2 can be installed into accounts that are still using Version 2013 Release 1, but note the following limitations:
 - When a user in a Version 2013 Release 1 account installs a bundle from a Version 2013 Release 2 account, any object types in the bundle that are not supported in Version 2013 Release 1 are not installed, resulting in potentially serious impact to bundle function.
 - A bundle with a SuiteScript that uses a new API available only in Version 2013 Release 2 can be installed into a Version 2013 Release 1 account, but the script may not function correctly because Version 2013 Release 1 does not support the API.

Warning: The only bundles that can be installed into a Release Preview account are those created in another Release Preview account. A bundle originally installed from a production account or the production repository can be updated in a Release Preview account, as long as the original account or the repository is also available in Release Preview. A bundle originally installed from a sandbox account cannot be accessed or updated in a Release Preview account. Also, any bundle created in a Release Preview account cannot be installed into a production or sandbox environment, even if it has been copied to the bundle repository.

Selective Update of Sandbox Bundle Objects

Version 2013 Release 2 simplifies the task of updating bundles installed into production from sandbox. This release supports a new update process for this type of bundle, in order to meet the specialized requirements of bundles installed from a sandbox account into a production account.

The normal bundle update process updates all bundle objects in the target account, overwriting target account objects with source bundle objects. The Preview Bundle Update page lists bundle objects to be updated. This page also highlights in red any objects in the target account that conflict with bundle objects, allowing the user to choose whether to add and rename each bundle object, leaving the target account object intact, or to overwrite the existing target account object with the bundle object.

This bundle update process could be time-consuming, confusing, and error-prone for bundles installed into production from sandbox. The update of all production objects was unnecessary and had the potential to unintentionally overwrite production object changes, with no ability to recover these changes after the update. The Preview Bundle Update page displayed almost all production bundle objects as conflicting with sandbox bundle objects, making it difficult to sort out which production objects should not be overwritten. The preview page did not evaluate which bundle objects had changed in sandbox since the last bundle update, and therefore genuinely needed to be updated in production. The preview page also did not identify production objects which had been changed since the last bundle update, and therefore should not be overwritten.

As of Version 2013 Release 2, a new update process for bundles installed into production from sandbox allows selective updates to production objects. When you start the update process for this type of bundle, the system evaluates whether objects have changed in production and/or sandbox since the last bundle update, and makes intelligent decisions about whether the update should overwrite each production object. This limitation of updates to a subset of the production objects can improve performance and can prevent unintentional overwrites of production object changes.

The following table shows how the update process for bundles installed into production from sandbox works for different cases:

Production object changed since last bundle update?	Sandbox object changed since last bundle update?	Bundle Update Action	Action Text on Preview Bundle Update Page
No	No	No change to production object	No Change
No	Yes	Production object overwritten by sandbox object	Update
Yes	No	No change to production object	No Change
Yes	Yes	Production object overwritten by sandbox object (Note that this may overwrite production object changes.)	Overwrite Production Objects (with warning)

Production object changed since last bundle update?	Sandbox object changed since last bundle update?	Bundle Update Action	Action Text on Preview Bundle Update Page
Does not yet exist in production	Yes (added to bundle)	Sandbox object is added to production	Add
N/A	Yes (removed from bundle)	Production object is removed	Delete (with warning)

The Preview Bundle Update page for bundles installed into production from sandbox now clearly displays the actions to be taken during the update. Instead of providing a confusing list of false conflicts between production and sandbox objects, this page allows users to understand exactly what will happen to each production object during the update. Each object displays action text indicating what the update will do to the production object. The preview page highlights in red the Overwrite Production Changes and Delete actions, and provides hover text warnings for each of these actions. Also, when the user clicks the Update Bundle button on this page, a list of the production objects that will be deleted, or that will have changes overwritten, is displayed. These alerts allow you to review these objects in the production account before the update. You can verify that overwriting the changes will not be a problem, or keep a record of the changes so the objects can be manually fixed after the update. You also can decide not to proceed with the update.

Preview Bundle Update					
Name: ERPBundle		Current Version:	New Version: 1.1		
Bundle Contents	Existing Object in Conflict	Script ID	Action	Preference	Lock on Install
Custom Fields					
Entity Fields					
Emma Cust Field Entity pref		customfield_emma_cust_field_entity_pref	No Change		
Emma Cust Field Entity		customfield_emma_cust_field_entity	⚠ Overwrite Production Objects		
Item Fields					
Emma Cust Field - Item		customfield_emma_cust_field_item	Update		
Custom Lists/Records					
Lists					
Emma Cust List		customlist_emma_cust_list	⚠ Delete		
Emma Cust List Computer		customlist_emma_cust_list_computer	Add		
Records					
RawMaterial	RawMaterial	customrecord_rawmaterial	⚠ Add and Rename		
Emma CRT child01		customrecord_emma_crt_child01	No Change		
Forms					
Entry Forms					
Emma Form - Inventory Part			Update		
Emma Form - Non-Inventory Part			No Change		
Emma Form - Project			No Change		
Emma Form - Customer			⚠ Overwrite Production Objects		
Online Forms					
Case Forms					
Emma Form - Online Case			⚠ Delete		
Roles					
Custom Roles					
Emma Role		customrole_emma_role	No Change		
Saved Searches					
<div>Export - CSV</div> <div>Cancel</div> <div>Update Bundle</div>					

Restoring Bundle Definitions after Sandbox Refresh

Before this release, if a bundle creator installed a bundle created in a sandbox account into a production account, and that sandbox account was later refreshed with data from that same

production account, the definition of the bundle in the sandbox account was lost after the refresh.

This issue occurred because bundles are handled differently in source and target accounts. In a source account, the bundle is listed under Saved Bundles. Users with the SuiteBundler permission can access the bundle definition to make edits to its contents and to take actions such as copying, installing, and updating the bundle in other accounts. In a target account, the bundle is listed under Installed Bundles. Users cannot edit the bundle or take these actions. During a refresh of sandbox from production, the bundle is copied from the production account, which is a bundle target account. After the refresh, the bundle in the sandbox account is a copy of the bundle in the production account, so it is treated as a target account bundle and listed under Installed Bundles.

This issue made it difficult to do repeated updates of a bundle installed from sandbox into production. After a sandbox refresh, the bundle definition would no longer exist in the sandbox account, and there would be no link between the production account bundle and its original source bundle in sandbox. If further work was needed on the bundle, it would need to be recreated manually in the sandbox account, and this new bundle definition would not be directly linked to the existing bundle installed in production.

As of Version 2013 Release 2, the sandbox refresh process automatically restores bundle definitions in the sandbox account, completely eliminating this issue. Now, the source bundle in sandbox continues to exist after sandbox refreshes, and the link between the source bundle and the target bundle in production is maintained. As a result of this change, a bundle installed into production from sandbox can be continually updated with changes made to the source bundle in sandbox.

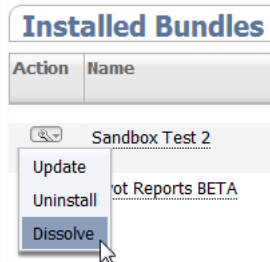
Dissolving Sandbox Bundles

The issue with sandbox bundle definitions being lost after sandbox refresh, described in [Restoring Bundle Definitions after Sandbox Refresh](#), resulted in production accounts having installed bundles that no longer had any link to their source bundles in sandbox. These bundles cannot be updated, so grouping the customization objects into a bundle no longer has any purpose.

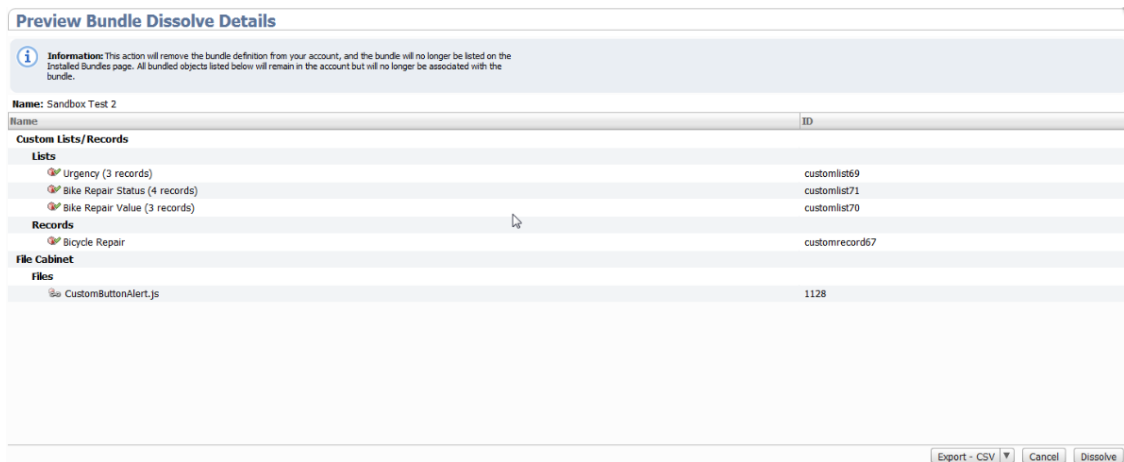
To allow the removal of these “orphaned” bundles, Version 2013 Release 2 supports a new action called “Dissolve”. This action removes the bundle from the production account. The customization objects installed as part of the bundle remain in the production account and do not lose any functionality. They are simply no longer considered part of a bundle.

The Dissolve action is available for any bundle installed into a production account from sandbox, so it can be used for other bundles in addition to orphaned bundles. For example, when the development cycle of a bundle is complete, you can dissolve a bundle in a production account where it has been installed, if you want the bundle objects to be treated as native objects rather than bundled objects.

Account administrators can find the Dissolve action at Setup > Customization > Search & Install Bundles > List, as a new option in the Action dropdown.



When a user selects the Dissolve action, a preview page displays. This page lists the objects in the bundle to be dissolved, and allows the user to either cancel or proceed with the action.



SuiteBuilder - Customization

Version 2013 Release 2 includes the following enhancements to SuiteBuilder features:

- Advanced PDF/HTML Templates (Beta)
- Translation of Custom Record Instance Names
- Custom Fields with Values Derived from Summary Search Results
- Change to Custom Field IDs

Advanced PDF/HTML Templates (Beta)

The Version 2013 Release 1 feature called Advanced Printing (Beta) has been renamed to Advanced PDF/HTML Templates (Beta) in Version 2013 Release 2. This feature provides

expanded customization capabilities over the previously supported transaction form layouts, also known as legacy layouts. The beta version of this feature allows you to get familiar with this new HTML-template based model for print customization.

When this feature is enabled, you can associate advanced templates with custom forms, so that these templates are used to format printed and emailed versions of these forms' transactions. You can use advanced PDF/HTML templates to produce either PDF or HTML output, depending upon the settings of your print and email preferences. Standard templates are provided for each supported print type. You can create your own customized templates in a Template Editor that supports current industry standards for HTML-based editing, including rich text editing and HTML source editing. You also can use SuiteScript to produce HTML and PDF printed forms that take advantage of advanced template customization capabilities.

For information about this feature, see the following:

- [Setting a Custom Form to Use an Advanced Template](#)
- [Using the Template Editor](#)
- [Enhancements to Advanced Templates for this Release](#)
- [Current Limitations for the Advanced PDF/HTML Templates Feature](#)

For more details, see the help topic [Advanced PDF/HTML Templates \(Beta\)](#).

Setting a Custom Form to Use an Advanced Template

The following screenshot shows how you can set a custom form to use an advanced PDF/HTML template:

The screenshot displays the 'Edit Custom Transaction Form' window. At the top, there are buttons for 'Save', 'Cancel', 'Reset', and 'Save & Move Elements'. The form fields include: 'Name' (Custom Purchase Order), 'Type' (Purchase Order), 'Printing Type' (Advanced selected, Legacy unselected), 'Advanced PDF/HTML Template' (Custom Purchase PDF/HTML Template selected from a dropdown), 'Disclaimer' (empty text area), 'Address' (empty text area), 'Logo' (dropdown menu), 'Columns Width' (16.75), 'Layout Space' (7.5), 'Allow Add Multiple' (checked), 'Inactive' (unchecked), and 'Form is Preferred' (unchecked). At the bottom, there is a tab bar with the following tabs: 'Tabs', 'Field Groups', 'Screen Fields', 'Actions', 'Lists', 'Custom Code', 'Roles', and 'Linked Forms'.

Using the Template Editor

The following screenshot shows the template editor where you can create advanced PDF/HTML templates:

Advanced PDF/HTML Template

Save Cancel Reset

*Name: Custom Purchase PDF/HTML Template

Type: Purchase Order

Layout is Preferred ☒ Inactive ☐

Source

Styles Format Font Size A₁ A₂ A₃ A₄ A₅ A₆ A₇ A₈ A₉ A₁₀ A₁₁ A₁₂ A₁₃ A₁₄ A₁₅ A₁₆ A₁₇ A₁₈ A₁₉ A₂₀ A₂₁ A₂₂ A₂₃ A₂₄ A₂₅ A₂₆ A₂₇ A₂₈ A₂₉ A₃₀ A₃₁ A₃₂ A₃₃ A₃₄ A₃₅ A₃₆ A₃₇ A₃₈ A₃₉ A₄₀ A₄₁ A₄₂ A₄₃ A₄₄ A₄₅ A₄₆ A₄₇ A₄₈ A₄₉ A₅₀ A₅₁ A₅₂ A₅₃ A₅₄ A₅₅ A₅₆ A₅₇ A₅₈ A₅₉ A₆₀ A₆₁ A₆₂ A₆₃ A₆₄ A₆₅ A₆₆ A₆₇ A₆₈ A₆₉ A₇₀ A₇₁ A₇₂ A₇₃ A₇₄ A₇₅ A₇₆ A₇₇ A₇₈ A₇₉ A₈₀ A₈₁ A₈₂ A₈₃ A₈₄ A₈₅ A₈₆ A₈₇ A₈₈ A₈₉ A₉₀ A₉₁ A₉₂ A₉₃ A₉₄ A₉₅ A₉₆ A₉₇ A₉₈ A₉₉ A₁₀₀ A₁₀₁ A₁₀₂ A₁₀₃ A₁₀₄ A₁₀₅ A₁₀₆ A₁₀₇ A₁₀₈ A₁₀₉ A₁₁₀ A₁₁₁ A₁₁₂ A₁₁₃ A₁₁₄ A₁₁₅ A₁₁₆ A₁₁₇ A₁₁₈ A₁₁₉ A₁₂₀ A₁₂₁ A₁₂₂ A₁₂₃ A₁₂₄ A₁₂₅ A₁₂₆ A₁₂₇ A₁₂₈ A₁₂₉ A₁₃₀ A₁₃₁ A₁₃₂ A₁₃₃ A₁₃₄ A₁₃₅ A₁₃₆ A₁₃₇ A₁₃₈ A₁₃₉ A₁₄₀ A₁₄₁ A₁₄₂ A₁₄₃ A₁₄₄ A₁₄₅ A₁₄₆ A₁₄₇ A₁₄₈ A₁₄₉ A₁₅₀ A₁₅₁ A₁₅₂ A₁₅₃ A₁₅₄ A₁₅₅ A₁₅₆ A₁₅₇ A₁₅₈ A₁₅₉ A₁₆₀ A₁₆₁ A₁₆₂ A₁₆₃ A₁₆₄ A₁₆₅ A₁₆₆ A₁₆₇ A₁₆₈ A₁₆₉ A₁₇₀ A₁₇₁ A₁₇₂ A₁₇₃ A₁₇₄ A₁₇₅ A₁₇₆ A₁₇₇ A₁₇₈ A₁₇₉ A₁₈₀ A₁₈₁ A₁₈₂ A₁₈₃ A₁₈₄ A₁₈₅ A₁₈₆ A₁₈₇ A₁₈₈ A₁₈₉ A₁₉₀ A₁₉₁ A₁₉₂ A₁₉₃ A₁₉₄ A₁₉₅ A₁₉₆ A₁₉₇ A₁₉₈ A₁₉₉ A₂₀₀ A₂₀₁ A₂₀₂ A₂₀₃ A₂₀₄ A₂₀₅ A₂₀₆ A₂₀₇ A₂₀₈ A₂₀₉ A₂₁₀ A₂₁₁ A₂₁₂ A₂₁₃ A₂₁₄ A₂₁₅ A₂₁₆ A₂₁₇ A₂₁₈ A₂₁₉ A₂₂₀ A₂₂₁ A₂₂₂ A₂₂₃ A₂₂₄ A₂₂₅ A₂₂₆ A₂₂₇ A₂₂₈ A₂₂₉ A₂₃₀ A₂₃₁ A₂₃₂ A₂₃₃ A₂₃₄ A₂₃₅ A₂₃₆ A₂₃₇ A₂₃₈ A₂₃₉ A₂₄₀ A₂₄₁ A₂₄₂ A₂₄₃ A₂₄₄ A₂₄₅ A₂₄₆ A₂₄₇ A₂₄₈ A₂₄₉ A₂₅₀ A₂₅₁ A₂₅₂ A₂₅₃ A₂₅₄ A₂₅₅ A₂₅₆ A₂₅₇ A₂₅₈ A₂₅₉ A₂₆₀ A₂₆₁ A₂₆₂ A₂₆₃ A₂₆₄ A₂₆₅ A₂₆₆ A₂₆₇ A₂₆₈ A₂₆₉ A₂₇₀ A₂₇₁ A₂₇₂ A₂₇₃ A₂₇₄ A₂₇₅ A₂₇₆ A₂₇₇ A₂₇₈ A₂₇₉ A₂₈₀ A₂₈₁ A₂₈₂ A₂₈₃ A₂₈₄ A₂₈₅ A₂₈₆ A₂₈₇ A₂₈₈ A₂₈₉ A₂₉₀ A₂₉₁ A₂₉₂ A₂₉₃ A₂₉₄ A₂₉₅ A₂₉₆ A₂₉₇ A₂₉₈ A₂₉₉ A₃₀₀ A₃₀₁ A₃₀₂ A₃₀₃ A₃₀₄ A₃₀₅ A₃₀₆ A₃₀₇ A₃₀₈ A₃₀₉ A₃₁₀ A₃₁₁ A₃₁₂ A₃₁₃ A₃₁₄ A₃₁₅ A₃₁₆ A₃₁₇ A₃₁₈ A₃₁₉ A₃₂₀ A₃₂₁ A₃₂₂ A₃₂₃ A₃₂₄ A₃₂₅ A₃₂₆ A₃₂₇ A₃₂₈ A₃₂₉ A₃₃₀ A₃₃₁ A₃₃₂ A₃₃₃ A₃₃₄ A₃₃₅ A₃₃₆ A₃₃₇ A₃₃₈ A₃₃₉ A₃₄₀ A₃₄₁ A₃₄₂ A₃₄₃ A₃₄₄ A₃₄₅ A₃₄₆ A₃₄₇ A₃₄₈ A₃₄₉ A₃₅₀ A₃₅₁ A₃₅₂ A₃₅₃ A₃₅₄ A₃₅₅ A₃₅₆ A₃₅₇ A₃₅₈ A₃₅₉ A₃₆₀ A₃₆₁ A₃₆₂ A₃₆₃ A₃₆₄ A₃₆₅ A₃₆₆ A₃₆₇ A₃₆₈ A₃₆₉ A₃₇₀ A₃₇₁ A₃₇₂ A₃₇₃ A₃₇₄ A₃₇₅ A₃₇₆ A₃₇₇ A₃₇₈ A₃₇₉ A₃₈₀ A₃₈₁ A₃₈₂ A₃₈₃ A₃₈₄ A₃₈₅ A₃₈₆ A₃₈₇ A₃₈₈ A₃₈₉ A₃₉₀ A₃₉₁ A₃₉₂ A₃₉₃ A₃₉₄ A₃₉₅ A₃₉₆ A₃₉₇ A₃₉₈ A₃₉₉ A₄₀₀ A₄₀₁ A₄₀₂ A₄₀₃ A₄₀₄ A₄₀₅ A₄₀₆ A₄₀₇ A₄₀₈ A₄₀₉ A₄₁₀ A₄₁₁ A₄₁₂ A₄₁₃ A₄₁₄ A₄₁₅ A₄₁₆ A₄₁₇ A₄₁₈ A₄₁₉ A₄₂₀ A₄₂₁ A₄₂₂ A₄₂₃ A₄₂₄ A₄₂₅ A₄₂₆ A₄₂₇ A₄₂₈ A₄₂₉ A₄₃₀ A₄₃₁ A₄₃₂ A₄₃₃ A₄₃₄ A₄₃₅ A₄₃₆ A₄₃₇ A₄₃₈ A₄₃₉ A₄₄₀ A₄₄₁ A₄₄₂ A₄₄₃ A₄₄₄ A₄₄₅ A₄₄₆ A₄₄₇ A₄₄₈ A₄₄₉ A₄₅₀ A₄₅₁ A₄₅₂ A₄₅₃ A₄₅₄ A₄₅₅ A₄₅₆ A₄₅₇ A₄₅₈ A₄₅₉ A₄₆₀ A₄₆₁ A₄₆₂ A₄₆₃ A₄₆₄ A₄₆₅ A₄₆₆ A₄₆₇ A₄₆₈ A₄₆₉ A₄₇₀ A₄₇₁ A₄₇₂ A₄₇₃ A₄₇₄ A₄₇₅ A₄₇₆ A₄₇₇ A₄₇₈ A₄₇₉ A₄₈₀ A₄₈₁ A₄₈₂ A₄₈₃ A₄₈₄ A₄₈₅ A₄₈₆ A₄₈₇ A₄₈₈ A₄₈₉ A₄₉₀ A₄₉₁ A₄₉₂ A₄₉₃ A₄₉₄ A₄₉₅ A₄₉₆ A₄₉₇ A₄₉₈ A₄₉₉ A₅₀₀ A₅₀₁ A₅₀₂ A₅₀₃ A₅₀₄ A₅₀₅ A₅₀₆ A₅₀₇ A₅₀₈ A₅₀₉ A₅₁₀ A₅₁₁ A₅₁₂ A₅₁₃ A₅₁₄ A₅₁₅ A₅₁₆ A₅₁₇ A₅₁₈ A₅₁₉ A₅₂₀ A₅₂₁ A₅₂₂ A₅₂₃ A₅₂₄ A₅₂₅ A₅₂₆ A₅₂₇ A₅₂₈ A₅₂₉ A₅₃₀ A₅₃₁ A₅₃₂ A₅₃₃ A₅₃₄ A₅₃₅ A₅₃₆ A₅₃₇ A₅₃₈ A₅₃₉ A₅₄₀ A₅₄₁ A₅₄₂ A₅₄₃ A₅₄₄ A₅₄₅ A₅₄₆ A₅₄₇ A₅₄₈ A₅₄₉ A₅₅₀ A₅₅₁ A₅₅₂ A₅₅₃ A₅₅₄ A₅₅₅ A₅₅₆ A₅₅₇ A₅₅₈ A₅₅₉ A₅₆₀ A₅₆₁ A₅₆₂ A₅₆₃ A₅₆₄ A₅₆₅ A₅₆₆ A₅₆₇ A₅₆₈ A₅₆₉ A₅₇₀ A₅₇₁ A₅₇₂ A₅₇₃ A₅₇₄ A₅₇₅ A₅₇₆ A₅₇₇ A₅₇₈ A₅₇₉ A₅₈₀ A₅₈₁ A₅₈₂ A₅₈₃ A₅₈₄ A₅₈₅ A₅₈₆ A₅₈₇ A₅₈₈ A₅₈₉ A₅₉₀ A₅₉₁ A₅₉₂ A₅₉₃ A₅₉₄ A₅₉₅ A₅₉₆ A₅₉₇ A₅₉₈ A₅₉₉ A₆₀₀ A₆₀₁ A₆₀₂ A₆₀₃ A₆₀₄ A₆₀₅ A₆₀₆ A₆₀₇ A₆₀₈ A₆₀₉ A₆₁₀ A₆₁₁ A₆₁₂ A₆₁₃ A₆₁₄ A₆₁₅ A₆₁₆ A₆₁₇ A₆₁₈ A₆₁₉ A₆₂₀ A₆₂₁ A₆₂₂ A₆₂₃ A₆₂₄ A₆₂₅ A₆₂₆ A₆₂₇ A₆₂₈ A₆₂₉ A₆₃₀ A₆₃₁ A₆₃₂ A₆₃₃ A₆₃₄ A₆₃₅ A₆₃₆ A₆₃₇ A₆₃₈ A₆₃₉ A₆₄₀ A₆₄₁ A₆₄₂ A₆₄₃ A₆₄₄ A₆₄₅ A₆₄₆ A₆₄₇ A₆₄₈ A₆₄₉ A₆₅₀ A₆₅₁ A₆₅₂ A₆₅₃ A₆₅₄ A₆₅₅ A₆₅₆ A₆₅₇ A₆₅₈ A₆₅₉ A₆₆₀ A₆₆₁ A₆₆₂ A₆₆₃ A₆₆₄ A₆₆₅ A₆₆₆ A₆₆₇ A₆₆₈ A₆₆₉ A₆₇₀ A₆₇₁ A₆₇₂ A₆₇₃ A₆₇₄ A₆₇₅ A₆₇₆ A₆₇₇ A₆₇₈ A₆₇₉ A₆₈₀ A₆₈₁ A₆₈₂ A₆₈₃ A₆₈₄ A₆₈₅ A₆₈₆ A₆₈₇ A₆₈₈ A₆₈₉ A₆₉₀ A₆₉₁ A₆₉₂ A₆₉₃ A₆₉₄ A₆₉₅ A₆₉₆ A₆₉₇ A₆₉₈ A₆₉₉ A₇₀₀ A₇₀₁ A₇₀₂ A₇₀₃ A₇₀₄ A₇₀₅ A₇₀₆ A₇₀₇ A₇₀₈ A₇₀₉ A₇₁₀ A₇₁₁ A₇₁₂ A₇₁₃ A₇₁₄ A₇₁₅ A₇₁₆ A₇₁₇ A₇₁₈ A₇₁₉ A₇₂₀ A₇₂₁ A₇₂₂ A₇₂₃ A₇₂₄ A₇₂₅ A₇₂₆ A₇₂₇ A₇₂₈ A₇₂₉ A₇₃₀ A₇₃₁ A₇₃₂ A₇₃₃ A₇₃₄ A₇₃₅ A₇₃₆ A₇₃₇ A₇₃₈ A₇₃₉ A₇₄₀ A₇₄₁ A₇₄₂ A₇₄₃ A₇₄₄ A₇₄₅ A₇₄₆ A₇₄₇ A₇₄₈ A₇₄₉ A₇₅₀ A₇₅₁ A₇₅₂ A₇₅₃ A₇₅₄ A₇₅₅ A₇₅₆ A₇₅₇ A₇₅₈ A₇₅₉ A₇₆₀ A₇₆₁ A₇₆₂ A₇₆₃ A₇₆₄ A₇₆₅ A₇₆₆ A₇₆₇ A₇₆₈ A₇₆₉ A₇₇₀ A₇₇₁ A₇₇₂ A₇₇₃ A₇₇₄ A₇₇₅ A₇₇₆ A₇₇₇ A₇₇₈ A₇₇₉ A₇₈₀ A₇₈₁ A₇₈₂ A₇₈₃ A₇₈₄ A₇₈₅ A₇₈₆ A₇₈₇ A₇₈₈ A₇₈₉ A₇₉₀ A₇₉₁ A₇₉₂ A₇₉₃ A₇₉₄ A₇₉₅ A₇₉₆ A₇₉₇ A₇₉₈ A₇₉₉ A₈₀₀ A₈₀₁ A₈₀₂ A₈₀₃ A₈₀₄ A₈₀₅ A₈₀₆ A₈₀₇ A₈₀₈ A₈₀₉ A₈₁₀ A₈₁₁ A₈₁₂ A₈₁₃ A₈₁₄ A₈₁₅ A₈₁₆ A₈₁₇ A₈₁₈ A₈₁₉ A₈₂₀ A₈₂₁ A₈₂₂ A₈₂₃ A₈₂₄ A₈₂₅ A₈₂₆ A₈₂₇ A₈₂₈ A₈₂₉ A₈₃₀ A₈₃₁ A₈₃₂ A₈₃₃ A₈₃₄ A₈₃₅ A₈₃₆ A₈₃₇ A₈₃₈ A₈₃₉ A₈₄₀ A₈₄₁ A₈₄₂ A₈₄₃ A₈₄₄ A₈₄₅ A₈₄₆ A₈₄₇ A₈₄₈ A₈₄₉ A₈₅₀ A₈₅₁ A₈₅₂ A₈₅₃ A₈₅₄ A₈₅₅ A₈₅₆ A₈₅₇ A₈₅₈ A₈₅₉ A₈₆₀ A₈₆₁ A₈₆₂ A₈₆₃ A₈₆₄ A₈₆₅ A₈₆₆ A₈₆₇ A₈₆₈ A₈₆₉ A₈₇₀ A₈₇₁ A₈₇₂ A₈₇₃ A₈₇₄ A₈₇₅ A₈₇₆ A₈₇₇ A₈₇₈ A₈₇₉ A₈₈₀ A₈₈₁ A₈₈₂ A₈₈₃ A₈₈₄ A₈₈₅ A₈₈₆ A₈₈₇ A₈₈₈ A₈₈₉ A₈₉₀ A₈₉₁ A₈₉₂ A₈₉₃ A₈₉₄ A₈₉₅ A₈₉₆ A₈₉₇ A₈₉₈ A₈₉₉ A₉₀₀ A₉₀₁ A₉₀₂ A₉₀₃ A₉₀₄ A₉₀₅ A₉₀₆ A₉₀₇ A₉₀₈ A₉₀₉ A₉₁₀ A₉₁₁ A₉₁₂ A₉₁₃ A₉₁₄ A₉₁₅ A₉₁₆ A₉₁₇ A₉₁₈ A₉₁₉ A₉₂₀ A₉₂₁ A₉₂₂ A₉₂₃ A₉₂₄ A₉₂₅ A₉₂₆ A₉₂₇ A₉₂₈ A₉₂₉ A₉₃₀ A₉₃₁ A₉₃₂ A₉₃₃ A₉₃₄ A₉₃₅ A₉₃₆ A₉₃₇ A₉₃₈ A₉₃₉ A₉₄₀ A₉₄₁ A₉₄₂ A₉₄₃ A₉₄₄ A₉₄₅ A₉₄₆ A₉₄₇ A₉₄₈ A₉₄₉ A₉₅₀ A₉₅₁ A₉₅₂ A₉₅₃ A₉₅₄ A₉₅₅ A₉₅₆ A₉₅₇ A₉₅₈ A₉₅₉ A₉₆₀ A₉₆₁ A₉₆₂ A₉₆₃ A₉₆₄ A₉₆₅ A₉₆₆ A₉₆₇ A₉₆₈ A₉₆₉ A₉₇₀ A₉₇₁ A₉₇₂ A₉₇₃ A₉₇₄ A₉₇₅ A₉₇₆ A₉₇₇ A₉₇₈ A₉₇₉ A₉₈₀ A₉₈₁ A₉₈₂ A₉₈₃ A₉₈₄ A₉₈₅ A₉₈₆ A₉₈₇ A₉₈₈ A₉₈₉ A₉₉₀ A₉₉₁ A₉₉₂ A₉₉₃ A₉₉₄ A₉₉₅ A₉₉₆ A₉₉₇ A₉₉₈ A₉₉₉ A₁₀₀₀ A₁₀₀₁ A₁₀₀₂ A₁₀₀₃ A₁₀₀₄ A₁₀₀₅ A₁₀₀₆ A₁₀₀₇ A₁₀₀₈ A₁₀₀₉ A₁₀₁₀ A₁₀₁₁ A₁₀₁₂ A₁₀₁₃ A₁₀₁₄ A₁₀₁₅ A₁₀₁₆ A₁₀₁₇ A₁₀₁₈ A₁₀₁₉ A₁₀₂₀ A₁₀₂₁ A₁₀₂₂ A₁₀₂₃ A₁₀₂₄ A₁₀₂₅ A₁₀₂₆ A₁₀₂₇ A₁₀₂₈ A₁₀₂₉ A₁₀₃₀ A₁₀₃₁ A₁₀₃₂ A₁₀₃₃ A₁₀₃₄ A₁₀₃₅ A₁₀₃₆ A₁₀₃₇ A₁₀₃₈ A₁₀₃₉ A₁₀₄₀ A₁₀₄₁ A₁₀₄₂ A₁₀₄₃ A₁₀₄₄ A₁₀₄₅ A₁₀₄₆ A₁₀₄₇ A₁₀₄₈ A₁₀₄₉ A₁₀₅₀ A₁₀₅₁ A₁₀₅₂ A₁₀₅₃ A₁₀₅₄ A₁₀₅₅ A₁₀₅₆ A₁₀₅₇ A₁₀₅₈ A₁₀₅₉ A₁₀₆₀ A₁₀₆₁ A₁₀₆₂ A₁₀₆₃ A₁₀₆₄ A₁₀₆₅ A₁₀₆₆ A₁₀₆₇ A₁₀₆₈ A₁₀₆₉ A₁₀₇₀ A₁₀₇₁ A₁₀₇₂ A₁₀₇₃ A₁₀₇₄ A₁₀₇₅ A₁₀₇₆ A₁₀₇₇ A₁₀₇₈ A₁₀₇₉ A₁₀₈₀ A₁₀₈₁ A₁₀₈₂ A₁₀₈₃ A₁₀₈₄ A₁₀₈₅ A₁₀₈₆ A₁₀₈₇ A₁₀₈₈ A₁₀₈₉ A₁₀₉₀ A₁₀₉₁ A₁₀₉₂ A₁₀₉₃ A₁₀₉₄ A₁₀₉₅ A₁₀₉₆ A₁₀₉₇ A₁₀₉₈ A₁₀₉₉ A₁₁₀₀ A₁₁₀₁ A₁₁₀₂ A₁₁₀₃ A₁₁₀₄ A₁₁₀₅ A₁₁₀₆ A₁₁₀₇ A₁₁₀₈ A₁₁₀₉ A₁₁₁₀ A₁₁₁₁ A₁₁₁₂ A₁₁₁₃ A₁₁₁₄ A₁₁₁₅ A₁₁₁₆ A₁₁₁₇ A₁₁₁₈ A₁₁₁₉ A₁₁₂₀ A₁₁₂₁ A₁₁₂₂ A₁₁₂₃ A₁₁₂₄ A₁₁₂₅ A₁₁₂₆ A₁₁₂₇ A₁₁₂₈ A₁₁₂₉ A₁₁₃₀ A₁₁₃₁ A₁₁₃₂ A₁₁₃₃ A₁₁₃₄ A₁₁₃₅ A₁₁₃₆ A₁₁₃₇ A₁₁₃₈ A₁₁₃₉ A₁₁₄₀ A₁₁₄₁ A₁₁₄₂ A₁₁₄₃ A₁₁₄₄ A₁₁₄₅ A₁₁₄₆ A₁₁₄₇ A₁₁₄₈ A₁₁₄₉ A₁₁₅₀ A₁₁₅₁ A₁₁₅₂ A₁₁₅₃ A₁₁₅₄ A₁₁₅₅ A₁₁₅₆ A₁₁₅₇ A₁₁₅₈ A₁₁₅₉ A₁₁₆₀ A₁₁₆₁ A₁₁₆₂ A₁₁₆₃ A₁₁₆₄ A₁₁₆₅ A₁₁₆₆ A₁₁₆₇ A₁₁₆₈ A₁₁₆₉ A₁₁₇₀ A₁₁₇₁ A₁₁₇₂ A₁₁₇₃ A₁₁₇₄ A₁₁₇₅ A₁₁₇₆ A₁₁₇₇ A₁₁₇₈ A₁₁₇₉ A₁₁₈₀ A₁₁₈₁ A₁₁₈₂ A₁₁₈₃ A₁₁₈₄ A₁₁₈₅ A₁₁₈₆ A₁₁₈₇ A₁₁₈₈ A₁₁₈₉ A₁₁₉₀ A₁₁₉₁ A₁₁₉₂ A₁₁₉₃ A₁₁₉₄ A₁₁₉₅ A₁₁₉₆ A₁₁₉₇ A₁₁₉₈ A₁₁₉₉ A₁₂₀₀ A₁₂₀₁ A₁₂₀₂ A₁₂₀₃ A₁₂₀₄ A₁₂₀₅ A₁₂₀₆ A₁₂₀₇ A₁₂₀₈ A₁₂₀₉ A₁₂₁₀ A₁₂₁₁

Advanced PDF/HTML Templates			
Edit	Name	Type ▲	Preferred
Customize	Standard Payment Voucher PDF/HTML Template	Bill Payment	✓
Customize	Standard Cash Refund PDF/HTML Template	Cash Refund	✓
Customize	Standard Cash Sale PDF/HTML Template	Cash Sale	✓
Customize	Standard Credit Memo PDF/HTML Template	Credit Memo	✓
Customize	Standard Customer Deposit PDF/HTML Template	Customer Deposit	✓
Customize	Standard Invoice PDF/HTML Template	Invoice	✓
Customize	Standard Picking Ticket PDF/HTML Template	Picking Ticket	✓
Customize	Standard Packing Slip PDF/HTML Template	Packing Slip	✓
Customize	Standard Sales Order PDF/HTML Template	Order	✓
Customize	Standard Purchase PDF/HTML Template	Purchase Order	✓
Customize	Standard Quote PDF/HTML Template	Quote	✓
Customize	Standard Return Authorization PDF/HTML Template	Return Authorization	✓

Show Inactives ☐

Submit Print Export - CSV ▼

- Ability to modify templates in the template editor to include fields from records other than the current transaction record. For example, if you are editing a sales order template, you can add fields from related records as well as fields from the current sales order record. Supported syntax is `${record.<related record>.<field ID>}`. For example, to include Sales Description field values from related item records in a sales order template, use the syntax: `${record.item.salesdescription}`.
- Ability to modify templates in the template editor to include search results fields.
- Automatic inclusion of the company logo on printed forms, based on the image file defined at Setup > Company > Company Information.
- Ability to print in the locale of the current user, set as the PDF Language at Home > Set Preferences. (Note that there is a current known limitation with East Asian languages that use the CJK font. See [Current Limitations for the Advanced PDF/HTML Templates Feature](#).)
- Ability to print in the locale of the customer associated with the current record. (The Print in Customer's Locale option is available for the printing of some transactions accessed through Transactions > Management > Print Checks and Forms.)
- Various issue fixes, including usability enhancements to the template editor.
- Template redesign, aimed at improving the look and feel of printed transactions. This redesign process is an ongoing effort.

Important: Customized PDF/HTML templates created prior to Version 2013 Release 2 have been migrated to be compatible with the new release's changes to this feature.

Current Limitations for the Advanced PDF/HTML Templates Feature

- Advanced PDF/HTML templates are not supported for all record types.
- The template editor does not provide a prepopulated selector for field names; users must obtain field IDs themselves. (See the help topic Adding a Field to an Advanced Template for tips for obtaining field IDs.)
- Advanced PDF/HTML templates cannot be included in bundles.
- The Download PDF Files user preference, set at Home > Preferences on the General subtab's Defaults area, is not supported for transactions using advanced PDF/HTML templates.
- Advanced PDF/HTML templates do not support printing in Chinese, Japanese, or Korean by default. In order to print a transaction using one of these templates in one of these languages, you need to edit the template's HTML source in the template editor, to change the CSS definition for the table attribute from *font-family: sans-serif;* to *font-family: hygothic, sans-serif;*. Also note that the user who is printing needs to have the CJK font pack installed.

Translation of Custom Record Instance Names

Version 2013 Release 2 supports the translation of custom record instance names so that they match the language of the NetSuite user interface. Before this release, you could provide translations for the name of the custom record type itself, but not for the names of the individual instances of that record type. For example, before you could provide the Spanish translation "Muebles" for a custom record type named Furniture, but you could not provide translations for the individual instances of that record type, such as chair, table, bed. As of this release, you can provide translations for chair, table, bed.

To allow translation of custom record instance names, check the new Enable Name Translation option for the custom record type. This option is disabled by default.

Edit Custom Record Type: Furniture

Save Cancel Reset Change ID More Actions ▾

*Name Furniture

ID customrecord810

Internal ID 810

Owner Sloan, Cindy

Description

Include Name Field ☒

Show ID ☐

Show Creation Date On Record ☐ On List ☐

Show Last Modified On Record ☐ On List ☐

Show Owner On Record ☐ On List ☐ Allow Change ☐

Access Type Require Custom Re...tries Permission ▾

Allow UI Access ☒

Allow Attachments ☒

Show Notes ☒

Enable Mail Merge ☐

Records are Ordered ☐

Allow Child Record Editing ☐ Allow Delete ☐

Available Offline ☐

Allow Quick Search ☐

Allow Quick Add ☒

Enable System Notes ☒

Include In Global Search ☒

Include In Search Menu ☒

Enable Optimistic Locking ☐

Enable Inline Editing ☒

Enable Name Translation ☒

Inactive ☐

Fields Subtabs Sublists Icon Numbering Forms Online Forms Permissions Links Managers Translation Child Records Parent Records History System Notes

You can enable the Enable Name Translation option for a custom record type if ALL of the following are true:

- The Multi-Language feature is enabled for the account.
- The Include Name Field option is enabled for the custom record type.
- Numbering is not enabled for the custom record type. (The Enable box on the Numbering subtab is not checked.)

When the Enable Name Translation box is checked for a custom record type, each custom record instance has a Translation subtab.

Edit Furniture

Save Cancel Reset More Actions ▾

*Name Chair Inactive ☐

Notes Files Translation Workflow

Language ▲	Name
Czech	
Spanish	Silla

Important: Before you can add these translations, you need to select translation languages at Setup > Company > General Preferences, on the Languages subtab. This subtab lists both system-supported languages that can be used for the NetSuite user interface (and are available at Home > Set Preferences), and additional languages that can be used for Web site translations only (and are not available at Home > Set Preferences). You should only enter translations for system-supported languages, because these are the only languages that can be displayed in the user interface. For details, see the help topic Setting Up Multiple Languages.

Custom Fields with Values Derived from Summary Search Results

Version 2013 Release 2 supports a new kind of custom field that can display values derived from summary saved search results. Each summary search field displays a rollup value for a selected search results field; this value is dynamically calculated each time a form containing the summary search field is displayed. For example, you can use a summary search field to display the total quantity of all line items on a transaction. The summary search field provides an alternative to using SuiteScript to calculate a custom field's values. Note that this functionality is very similar to that of custom sublists, except the search results are displayed in a field rather than a sublist.

To create this kind of field, you need to do the following:

1. Create or edit a summary saved search that rolls up to the result you want to display in the custom field.
 - Add search criteria to filter out any records/lines you do not want included in the rollup value.
 - Define a search results field for which values will be rolled up, and select a summary type. (Count, Sum, Minimum, Maximum, and Average are supported; Group is not supported.) For example, you could set Quantity as the results field and set a summary type of Sum.
 - Define an available filter field. This field is used to filter search results to include only those records with available filter field values that match the available filter field value of the current record. The value displayed for the summary search custom field is a rollup of search results field values for the records that have matching available filter field values.
2. Create or edit the custom field.
 - Select this search on the custom field's Validation and Defaulting subtab.
 - Optionally, you can select a "Compare To" field. You can use this field in cases where you want to put the custom field on a form for a record type that is different from the summary search record type. For example, if you want to put a custom

entity field showing the result of a customer record summary search on an employee custom form, you could select an employee record field whose values could be matched to the values for the search's Available Filter field. Search results field values for all records with matching values for the Compare To field and Available Filter field would be used to calculate the value of the summary search custom field.

- Do not enable the Store Value option, as values for this kind of custom field are not stored.

The following example shows the creation of a custom field to be displayed on custom purchase order forms, with the value of the field calculated by a purchase order summary search that sums the values for the purchase order line items' Amount field. No Compare To field is needed, because the Internal ID field set as the Available Filter for the search can simply be matched to the Internal ID of the purchase order record displayed on the form.

Edit Transaction Body Field

Save Cancel Reset Change ID Apply to Forms More Actions

*Label: Total Amount ID: custbody3 Owner: Jonathan Wolfe

Description: Type: Currency List/Record:

Store Value: ☒ Show In List: ☐ Global Search: ☐ Record is Parent: ☐

Applies To **Display** **Validation & Defaulting** **Sourcing & Filtering** **Access** **Translation** **History**

Mandatory: ☐ Check Spelling: ☐ Default Value: Formula: ☐

Search: Total Purchase Order Amount Field:

Supported Custom Field Types and Data Types

You can select a summary search to provide rollup values for the following custom field types: Entity Fields, Item Fields, CRM Fields, Transaction Body Fields, Other Custom Fields, and Custom Record Custom Fields.

This option is supported for the following types: Currency, Date, Date/Time, Decimal Number, Email Address, Free-Form Text, Hyperlink, Integer Number, Long Text, Percent, Phone Number, Rich Text, Text Area, and Time of Day.

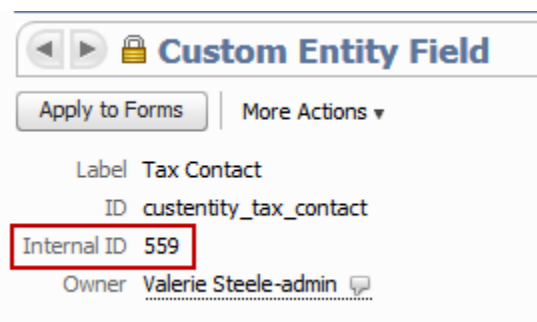
Current Limitations for Summary Search Custom Fields

- Custom field values from summary search results are never stored. Field values are always calculated dynamically at run time. A user may be able to edit values, depending on the display options set for the custom field, but the edited values will not be stored. Also, because its values are not stored, this kind of field is not available in search results, including lists based on saved searches.
- Calculated values for this kind of field may be different for users with different permissions. Summary search results are rolled up for the records to which the current user has access. Because users with different permissions may have access to different sets of records, the calculated value of this field may vary per user.

Change to Custom Field IDs

Version 2013 Release 2 has changed the ID values available for custom fields. Before this release, each custom field record had only one ID value, a unique string that could be defined by users. Now, in addition to this field, which is labeled **ID**, each custom field also has an additional value for a field labeled **Internal ID**, a read-only unique numerical value generated by the system when a custom field record is saved.

Now when the Show Internal IDs preference has been enabled at Home > Set Preferences, an Internal ID value displays on each custom field record and an Internal ID column displays these values on custom field list pages.



Custom Entity Fields

#	Description	From Bundle	ID	Internal ID	Type
22	VAT Registration No 	14864	custentity_vat_reg_no	507	Free-Form Text
23	Tax Contact Last Name 	14864	custentity_tax_contact_last	556	Free-Form Text
24	Tax Contact First Name 	14864	custentity_tax_contact_first	557	Free-Form Text
25	Tax Contact Middle Name 	14864	custentity_tax_contact_middle	558	Free-Form Text
26	Tax Contact 	14864	custentity_tax_contact	559	List/Record
1	Residential Address		custentity4	38	Check Box

The URL for a custom field record in NetSuite previously included the ID value. Now it includes the Internal ID value instead.

- The following custom field URL is before Version 2013 Release 2:

https://system.netsuite.com/app/common/custom/entitycustfield.nl?id=CUSTENTITY_TAX_CONTACT&e=T

- The following custom field URL is after Version 2013 Release 2:

<https://system. netsuite.com/app/common/custom/entitycustfield.nl?id=559&e=T>

Note that as a result of this change, some internal ID fields are being renamed in the Web services 2013.2 endpoint. For details, see [With New Endpoint, Integrations Referencing Custom Fields Must Be Updated](#).

SuiteAnalytics (Dashboards, Reporting and Search)

Advance Notice – NetSuite to Retire ODBC Driver 5.5

In order to place more resources into strategic SuiteAnalytics projects, NetSuite ODBC 5.5 driver will be retired on December 31, 2013. After this date, you will no longer be able to run queries using the NetSuite ODBC 5.5 driver. Note that support for this driver was officially withdrawn in July, 2010.

Important: NetSuite strongly encourages customers using this driver to upgrade to NetSuite ODBC driver 6.0. Instructions for getting started with this driver are available in the NetSuite Help Center.

See the help topics [Upgrading from NetSuite ODBC Driver 5.5 and Downloading and Installing NetSuite ODBC Driver 6.0](#) to get started with the new driver.

CSV Import

Version 2013 Release 2 includes the following changes to CSV Import:

- [Support for New Record Type Imports](#)
- [New Option Prevents the Import of Duplicate Records](#)
- [In Japan Edition, New Customer Field Available for Mapping](#)

Support for New Record Type Imports

The following table lists newly supported record types for CSV import as of Version 2013 Release 2.

Newly Supported Record Type	Category	Notes
Events	Activities Import Type	<ul style="list-style-type: none">• An event is an activity, such as an appointment or meeting. You can use the event record to invite participants, track employee time, and reserve resources needed for the activity.• In the UI, you can view the event record by choosing Activities > Scheduling > Events > New.• When Reserve Time is set to Yes, the system reserves times on participants' calendars for the event.• The behavior of the Date field varies depending on whether Reserve Time is enabled. When Reserve Time is set to Yes, the Date field must include both a date and a start time (and in these cases, the record must also have an end date). When Reserve Time is set to No, the Date field is used for a date alone, and no End Date is used.• Exercise care with both the Date and End Date fields. The formatting used in your CSV file must match your preferences for Time Format and Date Format in NetSuite. You can view these preferences at Home > Set Preferences.• Recurrence is not supported.
Expense Categories	Accounting Import Type	<ul style="list-style-type: none">• An expense category record represents a type of expense that can be referenced on an employee's expense report. Every expense must belong to an expense category.

Newly Supported Record Type	Category	Notes
		<ul style="list-style-type: none"> The expense category record is available when the Expense Reports feature is enabled at Setup > Company > Enable Features, on the Employees subtab. In the UI, you access the expense category record by choosing Setup > Accounting > Expense Categories > New. The Translations sublist is not supported for import.
Expense Reports	Employees Import	<ul style="list-style-type: none"> An expense report transaction is used to record an employee's expenses for approval. The expense report record is available when the Expense Reports feature is enabled at Setup > Company > Enable Features, on the Employees subtab. In the UI, you can view the expense report record by choosing Transactions > Employees > Enter Expense Reports. The Expenses sublist can be selectively updated by either the Ref No or Line ID field. Of these, only Ref No is viewable in the UI. In the Expenses sublist, certain fields are available only when the Multiple Currencies feature is enabled at Setup > Company > Enable Features, on the Company tab. These fields include Currency, Exchange Rate, and Foreign Amount.
Inventory Cost Revaluation	Transactions Imports	<ul style="list-style-type: none"> The inventory cost revaluation record is used to activate a standard cost version in production, and also to recalculate the value of inventory items. The inventory cost revaluation record is available for import when the Standard Costing feature is enabled at Setup > Company > Enable Features, on the Items & Inventory subtab. In the UI, you access the inventory cost revaluation record by choosing Transactions > Inventory > Revalue Inventory Cost.
Track Time	Employees Import	<ul style="list-style-type: none"> The track time record is used to keep track of an employee's hours worked. The track time record is available when the Time Tracking feature is enabled at Setup > Company > Enable Features, on the Employees subtab. In the UI, you access the track time record by choosing Transactions > Employees > Track Time.

New Option Prevents the Import of Duplicate Records

Often in the course of doing business, it is possible to mistakenly create multiple records representing the same person or organization. NetSuite includes a feature (Duplicate Detection & Merge) to help manage this problem. In this release, the Import Assistant has been enhanced to support the Duplicate Detection & Merge feature. Specifically, the Import Assistant now

includes an option that will prevent the import of records that appear to be duplicates. This option can be used with the import of contacts, customers, partners, and vendors.

This new import option is available when the Duplicate Detection & Merge feature is enabled at Setup > Company > Enable Features, and when you have enabled the feature for specific record types on the Setup > Company > Duplicate Detection page. Also on this page, you can set criteria that determine when a record should be flagged as a duplicate. For example, you might choose to specify that if two contact records have the same phone number, one is probably a duplicate. When this rule is in place, if a user opens the New Contact form and types in a phone number identical to one used on another contact record, the system displays a warning. At that point, the user entering the record can cancel out of the process, or continue and save the record. If the record is saved, it is marked as a duplicate and can be managed at Lists > Mass Update > Entity Duplicate Resolution.

The CSV feature provides an extra layer of functionality. Specifically, you have the ability to specify that duplicate records should be blocked from import. To identify duplicates, the Import Assistant uses the same criteria that apply to the manual creation of records in the UI (for example, matching phone numbers, email addresses, and so on). These criteria can be managed at Setup > Company > Duplicate Detection.

The new import option is located on the Import Options page of the Import Assistant. To see it, expand the Advanced Options menu, and look for the box labeled Prevent Duplicate Records.

The screenshot shows the 'Import Options' window. At the top, under 'Data Handling', there are three radio buttons: 'Add' (selected), 'Update', and 'Add or Update'. Below this is the 'Advanced Options' section, which is expanded. It contains several checkboxes: 'Log System Notes for Custom Fields' (unchecked), 'Overwrite Missing Fields' (unchecked), 'Validate Mandatory Custom Fields' (checked), 'Overwrite Sublists' (unchecked), 'Ignore Read-Only Fields' (checked), 'Prevent Duplicate Records' (checked and highlighted with a yellow box), and 'Custom Multi-Select Value Delimiter' (with an empty text input field).

If you select this option and then try to import a record that appears to be a duplicate, the import of that record fails with an error reading “Duplicate record was detected.” If you do *not* select the Prevent Duplicate Records box, the duplicate record will be created and marked as a duplicate. For this reason, if you prefer for potential duplicates to be entered and reviewed on a case-by-case basis, you should turn the Prevent Duplicate Records option off. As long as the Duplicate Detection & Merge feature is enabled, records identified as duplicates according to your criteria are marked as such and flagged for manual review.

By default, the Prevent Duplicate Records box is not selected, and it is not possible to set a general preference for it. However, you can create a saved import that selects or de-selects this option as appropriate.

Note that even when the Prevent Duplicate Records option is used, it is possible for a duplicate record to be imported in the following situations:

- If two users are importing records at the same time, and both users are adding two similar records.
- If multi-threading is being used and two similar records are being processed in different queues at or around the same time.

In Japan Edition, New Customer Field Available for Mapping

For accounts using the Japan Edition of NetSuite, an additional field, Monthly Closing Date, has been exposed for mapping during customer imports. In the UI, this field is located on the Financial tab, in the Account Information area. This field is used only in the Japan Edition.

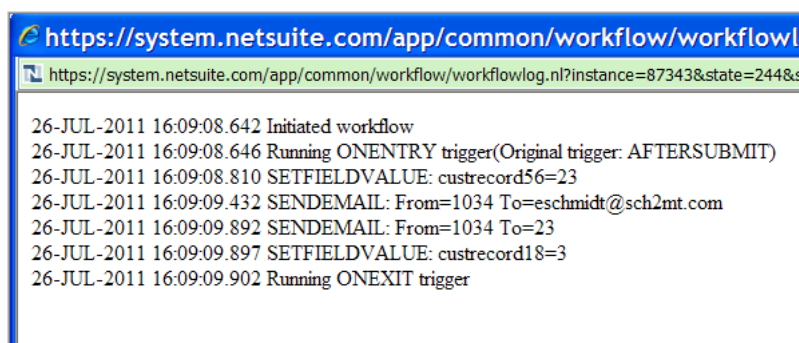
SuiteFlow (Workflow Manager)

Version 2013 Release 2 includes the following enhancements to SuiteFlow features:

- Enhanced Workflow Logging
- Enhanced Workflow Scheduling
- Enhancements for Add Button Actions in Workflows
- Deleting Workflow States

Enhanced Workflow Logging

Version 2013 Release 2 includes significant improvements to workflow logging. Before this release, workflow logs provided limited notes on executed triggers and their related actions, in a very basic interface.



The new workflow logs available with this release provide much more information than the old logs, and have a more understandable user interface. This interface is interactive, so that you can modify its display to more easily find the details you require.

Workflow Log		
Workflow Employee Demo		Start Time 7/22/2013 - 11:54:04.267
State State 1		
<input type="checkbox"/> Show Rejected Actions/Transitions		
Entry	Result	Date/Time
Field 'ALTEMAIL' not found in the record.	Warning	7/22/2013 - 11:54:04.685
▷ SETDISPLAYLABEL: Job Title -> Title [Field = TITLE]	Executed	7/22/2013 - 11:54:04.717
▲ Running BEFORELOAD trigger (Event: EDIT; Context: USERINTERFACE)		7/22/2013 - 11:54:48.356
▷ ADDBUTTON: Approve	Executed	7/22/2013 - 11:54:48.434
▷ ADDBUTTON: Reject	Executed	7/22/2013 - 11:54:48.496
REMOVEBUTTON: Delete	Executed	7/22/2013 - 11:54:48.556
▷ SETDISPLAYLABEL: Alt. Email -> Another Email [Warning: Field 'ALTEMAIL' not found in the re...	Skipped	7/22/2013 - 11:54:48.585
▷ SETDISPLAYLABEL: Job Title -> Title [Field = TITLE]	Executed	7/22/2013 - 11:54:48.656
▲ Running BEFORESUBMIT trigger (Event: EDIT; Context: USERINTERFACE)		7/22/2013 - 11:55:00.847
▷ SETFIELDVALUE [assistant = null]	Executed	7/22/2013 - 11:55:00.880
▲ Running AFTERSUBMIT trigger (Event: EDIT; Context: USERINTERFACE)		7/22/2013 - 11:55:01.737
▲ CREATE RECORD: Case (SUPPORTCASE)	Failed	7/22/2013 - 11:55:01.777
Record type: Case (SUPPORTCASE)		7/22/2013 - 11:55:01.791
basecost = 100.00		7/22/2013 - 11:55:02.040
status = 1		7/22/2013 - 11:55:02.089
assigned = -5		7/22/2013 - 11:55:02.102
title = New Case		7/22/2013 - 11:55:02.116
Action execution FAILED: Invalid assigned reference key -5.	Error	7/22/2013 - 11:55:02.375
▲ Running BEFORELOAD trigger (Event: EDIT; Context: USERINTERFACE)		7/22/2013 - 11:55:32.866
▷ ADDBUTTON: Approve	Executed	7/22/2013 - 11:55:32.973

The new logs include the following information that was not provided in the old logs: workflow name, current state, event and context for executed triggers, parameters and conditions for executed actions, and more in-depth error details.

Note: Errors are now logged for every workflow, even those that do not have the Enable Logging option enabled.

The increased level of detail in the new logs helps you to better understand what happened during workflow execution and improves your troubleshooting ability. For example, in the case where a workflow action fails, resulting in the rollback of many previous actions, the new logs fully reflect these details, while the old logs could not capture them.

In addition to increased content, the new logs offer the benefits of a more usable, flexible interface.

- New log entries have a hierarchical structure that you can expand and collapse to display the level of detail you prefer.

- New logs clearly indicate the status for each executed workflow action in the Result column: Executed, Skipped, or Failed. A Skipped result means that a non-critical error occurred to prevent an action from executing, but this error did not stop the execution of subsequent workflow actions.
- New logs offer the option of showing actions and transitions that were rejected, for example, actions with conditions that evaluated to false.

Workflow Log		
Workflow: Employee Demo		Start Time: 7/22/2013 - 11:54:04.267
State: State 1		
<input checked="" type="checkbox"/> Show Rejected Actions/Transitions		
Entry	Result	Date/Time
Initiated workflow		7/22/2013 - 11:54:04.332
▶ Running ONENTRY trigger under BEFORELOAD (Event: VIEW; Context: USERINTERFACE)		7/22/2013 - 11:54:04.363
▲ Running BEFORELOAD trigger (Event: EDIT; Context: USERINTERFACE)		7/22/2013 - 11:54:48.356
▲ ADDBUTTON: Approve	✓ Executed	7/22/2013 - 11:54:48.434
Condition: Supervisor Is Not Empty	✓ True	7/22/2013 - 11:54:48.464
▲ ADDBUTTON: Reject	✓ Executed	7/22/2013 - 11:54:48.496
Condition: Supervisor Is Not Empty	✓ True	7/22/2013 - 11:54:48.523
REMOVEBUTTON: Delete	✓ Executed	7/22/2013 - 11:54:48.556
▶ SETDISPLAYLABEL: Alt. Email -> Another Email [Warning: Field 'ALTEMAIL' not found in the re...	⚠ Skipped	7/22/2013 - 11:54:48.585
▶ SETDISPLAYLABEL: Job Title -> Title [Field = TITLE]	✓ Executed	7/22/2013 - 11:54:48.656
▲ Running BEFORESUBMIT trigger (Event: EDIT; Context: USERINTERFACE)		7/22/2013 - 11:55:00.847
▶ SETFIELDVALUE [assistant = null]	✓ Executed	7/22/2013 - 11:55:00.880
▲ RETURNUSERERROR: Please enter Job Title	⚪ Considered	7/22/2013 - 11:55:00.924
Condition: Job Title Is Empty	✗ False	7/22/2013 - 11:55:00.949
▲ RETURNUSERERROR: Please enter Supervisor	⚪ Considered	7/22/2013 - 11:55:00.963
Condition: Supervisor Is Empty	✗ False	7/22/2013 - 11:55:00.987
▲ Running AFTERSUBMIT trigger (Event: EDIT; Context: USERINTERFACE)		7/22/2013 - 11:55:01.737
▶ CREATERECORD: Case (SUPPORTCASE) [Error: Action execution FAILED: Invalid assigned refer...	⚠ Failed	7/22/2013 - 11:55:01.777
▶ Running BEFORELOAD trigger (Event: EDIT; Context: USERINTERFACE)		7/22/2013 - 11:55:32.866

Enhanced Workflow Scheduling

Before this release, when you set a workflow to be executed on a scheduled basis rather than triggered by events, the workflow was executed every 30 minutes, and you had no ability to alter this schedule. As of Version 2013 Release 2, many more options are available for scheduling workflows. These new options greatly increase the degree of control you have over when workflows execute.

As in previous releases, you still need to define a saved search for each scheduled workflow, and all of the records returned as search results enter the workflow according to the schedule. However, now you can set up a workflow to be executed on a recurring basis at defined

intervals. Or, you can schedule a workflow to be executed on a one-time basis at a specific time on a specific day.

You define workflow scheduling on the Workflow Summary page. This page has been modified to include a separate area for either Event Definition details or Schedule details, depending on whether the Event Based or Scheduled option is selected for Initiation.

If the Repeat check box is checked in the Schedule area of the page, you can enter the Scheduled From Date and Scheduled Until Date, to indicate the range of time during which the workflow should recur. You can choose a value for Frequency, to indicate the interval between workflow executions, with options for every 30 minutes, daily, weekly, monthly, or yearly. The default is every 30 minutes. Note that available execution times are at 30-minute intervals and are formatted according to each user's Time Format preference.

Workflow

Save Cancel Reset Change ID More Actions

Basic Information

*Name: Test
ID: customworkflow357
Record Type: Transaction
*Sub Types: Expense Report, Inventory Adjustment, **Invoice**, Item Fulfillment, Item Receipt
Description:
Owner: Employee One
Execute as Admin: ☐
Release Status: Not Running
Enable Logging: ☐
Inactive: ☐

Initiation

☐ Event Based ☒ Scheduled

Schedule

*Saved Search: Open Invoices by Due Date
Repeat: ☒
Frequency: Every 30 minutes
*Scheduled From Date: 7/1/2013
Scheduled Until Date: 7/31/2013

If you choose any value other than every 30 minutes, you can set the Execution Time, as well as other scheduling options.

Schedule

* Saved Search: Open Invoices by Due Date

Repeat: ☒

Frequency: Daily

* Scheduled From Date: 7/1/2013

Scheduled Until Date: 7/31/2013

* Execution Time: 10:00 pm

Daily Schedule

☒ Repeat every 1 day(s)

☐ Repeat every weekday

Schedule

* Saved Search: Open Invoices by Due Date

Repeat: ☒

Frequency: Weekly

* Scheduled From Date: 7/1/2013

Scheduled Until Date: 7/31/2013

* Execution Time: 10:00 pm

Weekly Schedule

Repeat every 1 week(s)

☐ Sunday ☒ Monday ☐ Tuesday ☐ Wednesday ☐ Thursday ☐ Friday ☐ Saturday

Schedule

* Saved Search: Open Invoices by Due Date

Repeat: ☒

Frequency: Monthly

* Scheduled From Date: 7/1/2013

Scheduled Until Date: 7/31/2013

* Execution Time: 10:00 pm

Monthly Schedule

☒ Day 1 of every 1 month(s)

☐ The first of every 1 month(s)

Schedule

* Saved Search: Open Invoices by Due Date

Repeat: ☒

Frequency: Yearly

* Scheduled From Date: 7/1/2013

Scheduled Until Date: 7/31/2013

* Execution Time: 10:00 pm

Yearly Schedule

☒ July 1 of July

☐ The first Monday of July

You can clear the Repeat check box in the Schedule area of the page, to indicate that a workflow should be scheduled for a single one-time execution. You can then enter the Execution Date and the Execution Time.

Workflow

Save Cancel Reset Change ID More Actions

Basic Information

* Name: Test

ID: customworkflow357

Record Type: Transaction

* Sub Types: Expense Report, Inventory Adjustment, **Invoice**, Item Fulfillment, Item Request

Description:

Owner: Employee One

Execute as Admin: ☐

Release Status: Not Running

Enable Logging: ☐

Inactive: ☐

Initiation

☐ Event Based ☒ Scheduled

Schedule

* Saved Search: Open Invoices by Due Date

Repeat: ☐

* Execution Date: 7/1/2013

* Execution Time: 10:00 pm

Enhancements for Add Button Actions in Workflows

Version 2013 Release 2 supports two new parameters for Add Button actions in workflows:

- Save record first
- Check condition before execution

Workflow: Approve Sales Training

Diagram

New State New Transition

Workflow Summary

```

graph TD
    A[Enter New Employee] --> B[Pending Training Approval]
    B --> C[Approved]
    B --> D[Rejected]
    D --> E[New State]

```

Workflow Action

Save Reset Cancel Change ID More Actions

Basic Information

Workflow: Approve Sales Training

State: Pending Training Approval

Type: Add Button

Insert Before: - Unchanged -

Trigger On: Before Record Load

Event Type: Copy, Create, Edit

Context: CSV Import, Custom Mass Update, Offline Client

Inactive: ☐

Script ID: workflowaction811

Condition

Use: ☒ Visual Builder ☐ Custom Formula

Condition: [Empty Box]

Saved Search: [Empty Box]

Parameters

*Label: Approve Training

Save record first: ☐

Check condition before execution: ☐

Language	Label
English (International)	[Empty Box]
French (France)	[Empty Box]
German	[Empty Box]
Thai	[Empty Box]

Save Reset Cancel Change ID More Actions

Save record first

The Save record first parameter indicates whether the record should always be saved before the action is executed. This option, which is available in Edit mode only, is disabled by default.

- When this option is disabled, the action has its own separate button on the page. For example, for the action shown in the screenshot above, a button labeled “Approve Training” would be added. When the Approve Training button is clicked, the record is not saved.
- When this option is enabled, the button for the action is added as an expanded option for the Save button, with a label of “Save & <Action Label>”. For example, if the Save record first box was checked for the action shown in the screenshot, an option labeled “Save & Approve Training” would be added. When the Save & Approve option is selected, the record is saved before the action is executed. If the save of the record fails, the action is not executed.

Check condition before execution

The Check condition before execution parameter indicates whether, when a user clicks the button, the condition associated with the button's action should be evaluated before the action is executed. This option is enabled by default for all new Add Button actions. (For backwards compatibility, this option is disabled for existing Add Button actions.) When this option is enabled, the system evaluates the condition and verifies that the record exists in the saved search results before executing the Add Button action. If the condition evaluates to false, or if it cannot be evaluated, the workflow does not execute the action.

Note the following:

- The check condition before execution parameter has no effect on an Add Button action that does not have a condition defined. In this case, the check simply evaluates to true and the action is executed.
- In cases where the Save record first and Check condition before execution options are both enabled, when the "Save &" button is clicked, the record is first saved, then the condition is evaluated based on the record's value after being saved. If the condition evaluates to false, the action is not executed.

Deleting Workflow States

Version 2013 Release 2 allows you to delete more workflow states than previously supported. Before this release, you could only delete states in workflows that had not yet executed. This restriction is no longer in place. Note that you still cannot delete a currently active workflow state, meaning a state that has been entered but not yet exited. Also, you cannot delete a workflow state that has transitions assigned to it; you must first delete the transitions. Delete is available in the More Actions menu for each workflow state.

For states that have been entered and exited, and later deleted, details remain accessible on the Workflow History subtab of the records that entered the workflow. On this subtab, the state name is appended with "(Deleted)". Also, the associated logs are available for 60 days from links on this subtab.

Note: In addition to showing when a state is deleted, the Workflow History subtab also clearly indicates renamed states, as shown in the following screenshot.

The screenshot shows the 'Edit Furniture' form with the 'Workflow History' subtab selected. The 'State Name Info' column is highlighted with a red box, showing a state that has been renamed: 'State 3: Update Record (deleted)'. The table below shows the workflow history.

Workflow	State Name Info	Date Entered State	Date Exited State	Options	Log	Notes
Furniture Save	State 3: Navigate to Customer	7/9/2013 14:42			Log	
Furniture Save	State 3: Update Record (deleted)	7/9/2013 14:42	7/9/2013 14:42		Log	
Furniture Save	State 2: Notify Customer	7/9/2013 14:42	7/9/2013 14:42		Log	
Furniture Save	State 1: Create Phone Call (now: State 1: Create Phone Calls)	7/9/2013 14:42	7/9/2013 14:42		Log	

SuiteScript

Refer to the following sections for details on all SuiteScript updates for Version 2013 Release 2:

- Enhancement to Tax Group Record Exposure
- Renaming of Duplicate Record Permission
- Component Yield Preferences — SuiteScript Impact
- Script Field ID Change
- Recalc Event Change
- HTTP Request Validation Change
- Script and Script Deployment Exposure
- Advanced Scheduled Script Management Enhancement
- nlapiYieldScript and nlapiSetRecoveryPoint Error Handling
- Alternate Time Zone Support
- Credential Builder API
- Void Transaction Support

- Script Performance Monitor SuiteApp
- Updated Version of SuiteCloud IDE
- Records Exposed to SuiteScript
- API Additions and Enhancements
- Changed Behavior when Scripting Project Tasks
- 2013.2 SuiteScript Records Browser

Enhancement to Tax Group Record Exposure

Version 2013 Release 2 includes an enhancement to the tax group record exposure in SuiteScript. This change makes it simpler to create a tax group for a nexus in a different country.

Specifically, in this release, you can specify the country by entering it as a parameter of `nlapiCreateRecord` (and doing nothing further). In previous releases, it was also necessary to include a separate line of code reiterating the nexus country.

For example, the following code creates a new tax group for Canada. In previous releases, the commented-out second line was required. In this release, it can be omitted:

```
var newgroup = nlapiCreateRecord('taxgroup', {nexuscountry: 'CA'}); //initial
value has to be specified //newgroup.setFieldValue('nexuscountry', 'CA'); //not necessary
anymore newgroup.setFieldValue('itemid', 'CA-T5'); newgroup.setFieldValue('piggyback', 'T');
newgroup.setFieldValue('taxitem1', 21); //Canadian tax code newgroup.setFieldValue('taxitem2',
24); //Canadian tax code newgroup.setFieldValue('subsidiary', 2); //Canadian subsidiary
newgroup.setFieldValue('state', 'AB'); //Canadian subsidiary
newgroup.setFieldValue('description', 'New Canada Tax Group');
nlapiSubmitRecord(newgroup);
```

Renaming of Duplicate Record Permission

In this release, the `LIST_DUPLICATES` permission has been renamed to `LIST_ENTITY_DUPLICATES`. Note also that the permission name in the UI has changed from Duplicate Record Management to Duplicate Entity Management. This permission is related to the Duplicate Detection & Merge feature. For more on this feature, see the help topic Duplicate Record Detection. For more details on SuiteScript permission IDs, see the help topic Permission Names and IDs.

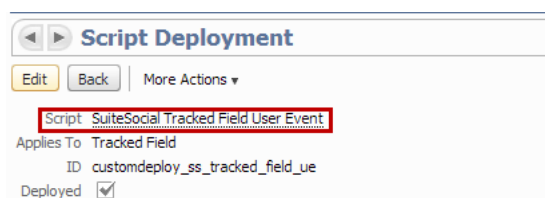
Component Yield Preferences — SuiteScript Impact

Version 2013 Release 2 includes new [Component Yield Preferences](#) for ERP. One of these preferences, the [Use Component Yield Check Box](#), may impact scripts deployed on the

assembly record. If you enable this option on the Components sublist, the Quantity field is no longer editable. Existing scripts that manipulate this field will no longer work.

Script Field ID Change

Version 2013 Release 2 includes a field ID change on the script deployment record. The field ID for the Script field is now 'script'. Prior to Version 2013 Release 2, the field ID for this field was 'scripttype'. NetSuite changed this field ID to enable the development of the [Script and Script Deployment Exposure](#) feature.



Recalc Event Change

As of Version 2013 Release 2, the recalc event is no longer triggered on the web store sales order form when an individual line item is changed. Recalc is now only triggered once, when the total changes at the end of the transaction. This change greatly improves performance, but may require updates to existing scripts.

Client scripts for the web store utilizing `getCurrentLineItem` and `setCurrentLineItem` functions no longer work on the recalc event. Developers who wish to update the current line item on the sales order form should script on the `validateLine` event instead.

Important: In general, the recalc event is designed for updating a global total, not for manipulating the current line item value. As of Version 2013 Release 2, non web store client scripts using `getCurrentLineItem` and `setCurrentLineItem` functions may technically work on the recalc event. Regardless, NetSuite recommends that all customers use the `validateLine` event to update current line item values. If needed, non web store customers should make every effort to update their client scripts.

HTTP Request Validation Change

We are making a configuration change to our infrastructure to tighten HTTP request validation. This change is targeted for release September 13, 2013. Customers and partners who have built integrations with SuiteScript may need to change their code to comply with the HTTP 1.1 specification. Users accessing the basic NetSuite service through a web browser are not likely to notice any changes.

One important result of this change is that HTTP requests containing white space in the URL will be rejected; the HTTP 1.1 specification does not allow white space in the URL. This policy may differ from the URL validation rules enforced by more lenient systems. We expect some impact to existing customers who are not properly encoding the URL for incoming calls to NetSuite resources. These calls are typically made using the `nlapiRequestURL` API and external programs making calls to RESTlets. There may be other incoming calls to NetSuite that are affected but we are not aware of specific customer impact.

Modifications to `nlapiRequestURL` API

To assist developers with troubleshooting scripts affected by this change, we are adding the following error codes and messages to the `nlapiRequestURL` API.

- When a connection is closed due to an invalid URL (including those containing white space), the following is thrown:
 - Error code: `SSS_INVALID_URL`
 - Error message: Invalid URL – Connection Closed
- When a connection is closed because the server associated with the URL is unresponsive, the following is thrown:
 - Error code: `SSS_CONNECTION_CLOSED`
 - Error message: Connection Closed

Recommended Code Fix

To fix invalid URLs affected by the configuration change, NetSuite recommends using the standard JavaScript function `encodeURIComponent(uri)` to encode the URL argument. For additional information, see http://www.w3schools.com/jsref/jsref_encodeuricomponent.asp. An example is provided below:

```
//Use encodeURIComponent when you want to encode a URL argument
var orderId = getOrderID(); //a dynamic generated value that could have space in it.
var customerId = getCustomerId(); //a dynamic generated value that could have space in it.
var url = 'www.netsuite.com/app/site/hosting/scriptlet.nl?script=187&deploy=1';
url += '&orderId=' + encodeURIComponent(orderId);
url += '&customerId=' + encodeURIComponent(customerId);
var response = nlapiRequestURL(url, null, null, null);
```

Script and Script Deployment Exposure

Version 2013 Release 2 exposes script and script deployment records to SuiteScript. With programmatic access to these records, you can more efficiently manage the deployment and execution of customizations and SuiteApps. For example, you can now write bundle installation scripts to detect if a target account has SuiteCloud Plus licenses. With that information, the

script can tailor the scheduled script deployment's queue assignment to match the target account's queue capacity. Additionally, scripts and their deployments can now be dynamically adjusted based on factors such as system load.

Script and script deployment records are accessed through the standard record APIs. These APIs can be used to read and update script records, and create, read, update, and delete script deployment records.

To access script and script deployment within a script, the script owner must log in with a role that has the SuiteScript permission.

For more details about this feature, see the following:

- [Items Not Exposed or Not Fully Exposed](#)
- [Bundle Restrictions](#)
- [Reading and Editing Scripts](#)
- [Copying, Creating, Reading, Deleting, and Editing Script Deployments](#)
- [Performing Searches on Script and Script Deployment Records](#)

Items Not Exposed or Not Fully Exposed

The following items are not exposed, or are not fully exposed, on the script record for this release:

- Change ID button (in Edit view of UI)
- Unhandled Errors subtab
- The Parameters subtab is exposed as read only; editing is not supported

The following items are not exposed on the script deployment record for this release:

- Change ID button (in Edit view of UI)
- Execution Log subtab
- Scheduling subtab (scheduled script deployments only)

Bundle Restrictions

Scripts can only access script and script deployment records when the records are part of the same bundle.

Scripts that are not part of a bundle cannot access script and script deployment records that are part of a bundle.

Reading and Editing Scripts

Note: Script records cannot be created, copied or deleted programmatically.

To load a script record, `nlapiLoadRecord` takes in one of the nine, case-insensitive, script types listed below:

- 'bundleinstallationscript'
- 'clientscript'
- 'massupdatescript'
- 'portlet'
- 'restlet'
- 'scheduledscript'
- 'suitelet'
- 'usereventscript'
- 'workflowactionscript'

For the second argument, you must use the script record's internal ID. Use of the script ID is not supported. You can find a script record's internal ID at Setup > Customization > Scripts (when the Show Internal IDs option is enabled at Home > Set Preferences).

Scripts							
Edit View	Name ^	From Bundle	ID	Type	Deployments	Script	Library Script
Edit View	142683_JT		customscript_csalert	Client	Deployments	csalert.js	
Edit View	142683_SS_JT		customscript_loadsuitelet	Suitelet	Deployments	load_suitelet.js	

```
//edit a user event script record var rec = nlapiLoadRecord('usereventscript',
302); rec.setFieldValue('name', 'userevent_001'); rec.setFieldValue('scriptfile', '227');
rec.setFieldValue('notifyadmins', 'T'); rec.setFieldValue('aftersubmitfunction',
'afterSubmitFunction'); nlapiSubmitRecord(rec);
```

Copying, Creating, Reading, Deleting, and Editing Script Deployments

To copy, create, read, delete and edit a script deployment record, use 'scriptdeployment' for the type argument. Note that 'scriptdeployment' is case-insensitive. For the id argument, you must use the script deployment record's internal ID. Previously, a script deployment's internal ID could be shared across multiple deployments. To support script deployment exposure, the script deployment's internal ID is now unique across all script deployments.

Script Deployments							
Internal ID	Edit View	ID	Script	Status	Type	Last Modified	
88	Edit View	customdeploy_newsfeed_email	SuiteSocial Email	In Queue	Scheduled	10.12.2012 2:01 am	
87	Edit View	customdeploy_update_profile_name_sked	Update Profile Name Sked	Not Scheduled	Scheduled	22.11.2012 6:19 am	

```
//copy and edit a script deployment record var rec =
nlobjCopyRecord('scriptdeployment', 41); rec.setFieldValue('isdeployed', 'F');
rec.setFieldValue('title', 'Feature 4141 - Suitelet Scratch 4');
nlobjSubmitRecord(rec);
```

Performing Searches on Script and Script Deployment Records

You can use `nlobjSearchRecord`, `nlobjLoadSearch`, and `nlobjCreateSearch` to perform searches on script and script deployment records. Note that when you use one of these APIs to search for script records, you must pass in 'script' as the type parameter. This differs from how you access script records with the record APIs; in those instances you pass in the individual script type.

```
//search a script record var filters = [new
nlobjSearchFilter('defaultfunction', null, 'is', 'myfunctionname')]; var columns = [new
nlobjSearchColumn('name')]; var results = nlobjSearchRecord('script', null, filters, columns);

//search a script deployment record var filters = [new nlobjSearchFilter('internalidnumber',
null, 'anyof', '43')]; var columns = [new nlobjSearchColumn('title')]; var results =
nlobjSearchRecord('scriptdeployment', null, filters, columns);
```

See [Advanced Scheduled Script Management Enhancement](#) for additional information on performing searches on records in the scheduled script queue.

Advanced Scheduled Script Management Enhancement

SuiteScript developers can now programmatically perform scheduled script queue searches with server-side scripts. To enable this feature, Version 2013 Release 2 introduces a new record type, scheduled script instance. This new record type is viewed in the UI as the Scheduled Script Status page. The Scheduled Script Status page shows the scheduled script queue. Each record listed on this page is an instance of scheduled script instance. To access this record in the UI, go to Setup > Customization > Script Deployments > Status.

Note: The Cancel column is not scriptable.

Scheduled Script Status								
Script	Deployment ID	Date Created v	Status	Start	End	Queue	% Complete	Cancel
SWV_SB_SS_Termination	customdeploy_sb_ss_termination	6/28/2013 10:01:18 am	Complete	10:02.14 am	10:02.14 am	1	100.0%	
SWV_SB_SS_Termination	customdeploy_sb_ss_termination	6/28/2013 8:01:00 am	Complete	8:02.12 am	8:02.13 am	1	100.0%	
SWV_SB_SS_Termination	customdeploy_sb_ss_termination	6/28/2013 6:01:01 am	Complete	6:02.09 am	6:02.10 am	1	100.0%	

Scheduled script instance is only scriptable through three existing APIs: `nlobjSearchRecord`, `nlobjLoadSearch`, and `nlobjCreateSearch`. In other words, developers can only perform searches

on this record type. The use of `nlapiCreateRecord`, `nlapiLoadRecord`, and `nlapiLookUpField` is not supported.

Note: To access scheduled script instance data, the script owner must log in with a role that has the SuiteScript permission.

nlapiYieldScript and nlapiSetRecoveryPoint Error Handling

NetSuite introduced the `nlapiSetRecoveryPoint` and `nlapiYieldScript` APIs in Version 2012 Release 1. These APIs were created to assist developers with scheduled script stabilization. With `nlapiSetRecoveryPoint`, developers could recover a failed script at a set recovery point. With `nlapiYieldScript`, developers could set a recovery point in a failed script and then place the script back into the queue to resume at the recovery point.

One limitation of these new APIs was that developers could not use `nlapiSetRecoveryPoint` or `nlapiYieldScript` in scripts after catching an `nlobjError` object. Attempting to do so would throw an `SS_DISALLOWED_OBJECT_REFERENCE` error. Version 2013 Release 2 supports the use of `nlapiSetRecoveryPoint` and `nlapiYieldScript` in scripts with an `nlobjError` object in their stack trace. The example script below will no longer throw `SS_DISALLOWED_OBJECT_REFERENCE` after catching an error.

```
function sampler() { try { var rec = nlapiLoadRecord('customer',1234); }  
  catch(err) { nlapiLogExecution('DEBUG','Error!'); err = null; } var state = nlapiYieldScript()  
  ;  
  // or nlapiSetRecovery... if(state.status == 'FAILURE'){ nlapiLogExecution("ERROR","Failed to  
  yield script, exiting: Reason = "+state.reason + " / Size = "+ state.size); } else  
  if(state.status == 'RESUME'){ nlapiLogExecution('DEBUG','Resuming script'); }  
}
```

Alternate Time Zone Support

Prior to Version 2013 Release 2, NetSuite was limited to storing and displaying the date/time of the user currently logged in. If a user was logged into NetSuite in New York, and needed to see an event time displayed from the perspective of a customer in Hong Kong, the user would have to do the time zone conversion manually. In addition, to do the time zone conversion correctly, the user would need to know the daylight savings time rules for both locations. Manual conversion was also required when the user was the system user (in other words, a script). In that case, the only date/time available to that script was based on where the NetSuite server was located. Developers would need to programmatically convert that date/time to get the value actually needed.

Version 2013 Release 2 includes support for alternate time zones. Developers can now manipulate the default time zone set on datetime fields.

For details, see the following:

- [Olson Values](#)

- [DateTime Time Zone API — User Event Scripts](#)
- [DateTime Time Zone Conversion Methods for nlobjRecord](#)

Olson Values

The newly supported alternate time zones are what are known as Olson Values. These values are maintained by the International Assigned Numbers Authority (IANA) in an international standard time zone database. This database defines all valid time zone names in a format similar to “America/Denver” and includes daylight savings time rules for each time zone. The values used to populated the Time Zone dropdown found at Home > Set Preferences are also based on these values.

DateTime Time Zone API — User Event Scripts

The new DateTime Time Zone APIs allow developers to manipulate default time zones with user event scripts:

- `nlapiSetDateTimeValue(fieldId, dateTime, timeZone)`
- `nlapiGetDateTimeValue(fieldId, timeZone)`
- `nlapiSetCurrentLineItemDateTimeValue(type, fieldId, dateTime, timeZone)`
- `nlapiGetCurrentLineItemDateTimeValue(type, fieldId, timeZone)`
- `nlapiSetLineItemDateTimeValue(type, fieldId, lineNum, dateTime, timeZone)`
- `nlapiGetLineItemDateTimeValue(type, fieldId, lineNum, timeZone)`

The `fieldId` passed in must point to a datetime field.

The `timeZone` argument is optional. If this argument is not supplied, the time zone will default to the time zone set in User Preferences.

The `timeZone` argument passed in can either be a string value or an integer. If it is a string value, it must exactly match one of the Olson values listed in Olson Valuestable (values are case-insensitive). If an integer is passed in as the `timeZone` argument, it must match one of the key values listed in the Olson Valuestable.

```
nlapiSetDateTimeValue('custrecord_datetimetz', '09/25/2013 06:00:01 am',
  'Asia/Manila'); var tz = nlapiGetDateTimeValue('custrecord_datetimetz',
  'America/Los_Angeles');
```

```
nlapiSetLineItemDateTimeValue('recmachcustrecord_childdatetime',
  'custrecord_datetimetzcol', 1, '09/25/2013 06:01:01 AM', 'Asia/Hong_Kong');
nlapiSelectNewLineItem('recmachcustrecord_childdatetime');
nlapiSetCurrentLineItemDateTimeValue('recmachcustrecord_childdatetime',
  'custrecord_datetimetzcol', '07/10/2013 06:00:01 am');
```

```
nlapiCommitLineItem('recmachcustrecord_childdatetime');
```

Date Time Zone Conversion Methods for nlobjRecord

New methods for nlobjRecord allow developers to manipulate default time zones with both remote client and server side script types.

- setDateTimeValue(fieldId, dateTime, timeZone)
- getDateTimeValue(fieldId, timeZone)
- setCurrentLineItemDateTimeValue(type, fieldId, dateTime, timeZone)
- getCurrentLineItemDateTimeValue(type, fieldId, timeZone)
- setLineItemDateTimeValue(type, fieldId, lineNum, dateTime, timeZone)
- getLineItemDateTimeValue(type, fieldId, lineNum, timeZone)

The fieldId passed in must point to a datetime field.

The timeZone argument is optional. If this argument is not supplied, the time zone will default to the time zone set in User Preferences.

The timeZone argument passed in can either be a string value or an integer. If it is a string value, it must exactly match one of the Olson values listed in Olson Values table (values are case-insensitive). If an integer is passed in as the timeZone argument, it must match one of the key values listed in the Olson Values table.

```
var a = nlapiLoadRecord('customrecord_parentdatetime', 1);
a.selectLineItem('recmachcustrecord_childdatetime', 1);
a.setCurrentLineItemDateTimeValue('recmachcustrecord_childdatetime',
'custrecord_datetimetzcol', '01/01/2013 06:00:01 am', 'America/Phoenix');
a.commitLineItem('recmachcustrecord_childdatetime'); nlapiSubmitRecord(a); var a =
nlapiLoadRecord('customrecord_parentdatetime', 1);
a.selectLineItem('recmachcustrecord_childdatetime', 1); var tz =
a.getCurrentLineItemDateTimeValue('recmachcustrecord_childdatetime',
'custrecord_datetimetzcol', 'America/Regina');
```

```
var a = nlapiLoadRecord('customrecord_parentdatetime', 1);
a.setLineItemDateTimeValue('recmachcustrecord_childdatetime', 'custrecord_datetimetzcol', 1,
'01/01/2013 06:00:01 am', 'America/Phoenix'); nlapiSubmitRecord(a); var a =
nlapiLoadRecord('customrecord_parentdatetime', 1); var tz =
a.getLineItemDateTimeValue('recmachcustrecord_childdatetime', 'custrecord_datetimetzcol', 1,
'America/Regina');
```

Credential Builder API

Before making secure requests to an external URL, some customers require additional encryption or encoding be performed on the request string. To satisfy these additional security

requirements, Version 2013 Release 2 introduces a new object, `nlobjCredentialBuilder(string, domainString)`, that developers can use with `nlapiRequestURLWithCredentials`. Six methods are included that perform various string transformations: two encryption methods for SHA-1 and SHA-256 encryption, two encoding methods for Base64 and UTF8 encoding, a character replacement method, and a string appending method.

- `sha1()`
- `sha256()`
- `base64()`
- `utf8()`
- `replace(string1, string2)`
- `append(string)`

The `nlobjCredentialBuilder` object takes in a user defined string, that can include an embedded GUID (global unique ID), as the first argument. It takes in your URL's host name as the second argument. Note that the host name passed in must exactly match the host name in your URL. For example, if your URL is `https://payment.ns.com/process.money?passwd={GUID}`, the host name passed in must be `'payment.ns.com'`.

The `nlobjCredentialBuilder` object is instantiated with the “new” keyword.

```
var builder = new nlobjCredentialBuilder("rawtext{GUID}Hash:",  
    'www.netsuite.com'); builder = builder.sha1().base64(); //builder now contains SHA-1 encrypted  
  
and Base64 encoded content builder = builder.replace('#', '-'); //replace all instances of "#" with "-" inside builder
```

Modifications to `nlapiRequestURLWithCredentials`

Developers can pass in the `nlobjCredentialBuilder` object as an argument to `nlapiRequestURLWithCredentials`. To accommodate for this usage, Version 2013 Release 2 includes several changes to the `nlapiRequestURLWithCredentials` API.

- `nlapiRequestURLWithCredentials(creds, url, postdata, headers, method)` now accepts the following as arguments:
 - `url` parameter: `nlobjCredentialBuilder` object
 - `postdata` parameter: `nlobjCredentialBuilder` object; a map of `<string, builder>` (in addition to the previously supported map of `<string, string>`)
 - `headers` parameter: `nlobjCredentialBuilder` object; a map of `<string, builder>` (in addition to the previously supported map of `<string, string>`)

- The creds argument can be null if the nlobjCredentialBuilder object is passed in as an argument.

Important: If the nlobjCredentialBuilder object is passed into nlapiRequestURLWithCredentials as the url argument, it must be passed in its original state (pre-encryption and pre-encoding). Otherwise, nlapiRequestURLWithCredentials will be unable to validate the url.

```
var _uname="user@company.com"; var _pwd = 'PASSWORD'; var creds = [_uname,
_pwd]; var url = 'https://cert.merchante-solutions.com/mes-api/tridentApi?'; url = url +
'profile_id=' + '{'+_uname+'}'; url = url + '&profile_key=' + '{'+_pwd+'}'; var cbSha256 =
new nlobjCredentialBuilder(url, 'cert.merchante-solutions.com').sha256(); var cbUtf8 =
cbSha256.utf8(); var cbBase64 = cbUtf8.base64(); var cbReplace = cbBase64.replace('-', '*');
var a = nlapiRequestURLWithCredentials(creds ,url,{ 'CredentialBuilder' : cbReplace
});
```

Void Transaction Support

When a transaction is voided, its total and all its line items are set to zero, but the transaction is not removed from the system. NetSuite supports two types of voids: direct voids and voids by reversing journals. See the help topic Voiding, Deleting, or Closing Transactions for additional information.

Developers can now void transactions programmatically with a new API that takes in the internal ID of the record type to be voided and internal ID of the specific record to be voided. The new API, nlapiVoidTransaction(transactionType, recordId), voids a transaction and then returns an id that indicates the type of void performed. If a direct void is performed, nlapiVoidTransaction returns the ID of the record voided. If a void by reversing journals is performed, nlapiVoidTransaction returns the ID of the newly created voiding journal.

The type of void performed depends on the targeted account's preference settings:

- If the Using Reversing Journals preference is disabled, nlapiVoidTransaction performs a direct void. See the help topic Supported Transaction Types - Direct Void for a list of transactions that support direct voids.
- If the Using Reversing Journals preference is enabled, nlapiVoidTransaction performs a void by reversing journals. See the help topic Supported Transaction Types — Void by Reversing Journals for a list of transactions that support voids by reversing journals.

The use of nlapiVoidTransaction is supported in client scripts, user events, scheduled scripts, suitelets, RESTlets and workflow action scripts. It has a governance of 10.

Supported Transaction Types

See the help topics Supported Transaction Types - Direct Void and Supported Transaction Types — Void by Reversing Journals.

Example

The following code creates a new Cash Refund transaction and then voids the transaction. Note that if the Using Reversing Journals preference is enabled, the void will not be successful since Cash Refund is not a supported transaction type for voids by reversing journals.

```
var recordtype = 'cashrefund'; var rec = nlapiCreateRecord(recordtype);
rec.setFieldValue('entity', 48); rec.setFieldValue('location', 1);
rec.setFieldValue('paymentmethod', 1); rec.setLineItemValue('item', 'item', 1, 233);
rec.setLineItemValue('item', 'amount', 1, 150); var id = nlapiSubmitRecord(rec); var voidingId
;
var errorMsg; try { voidingId = nlapiVoidTransaction(recordtype, id); } catch(e) { errorMsg =
e; }
```

Script Performance Monitor SuiteApp

The new Script Performance Monitor SuiteApp measures the performance of user event scripts, and allows administrators to quickly identify those scripts that take an unusually long time to execute. Script data is kept for a 30 day period. Within this period, administrators can search for scripts by record type, operation and user.

The Script Performance Monitor displays data by record type. For each record instance, it displays the overall time it took for all scripts deployed on that record type to execute. Administrators can also drill down into each instance to view the execution time of each individual script.

See the help topic Using the Script Performance Monitor for additional information.

Note: Script performance is only measured on a per record and per script basis; the Script Performance Monitor cannot identify the performance of individual API calls.

Updated Version of SuiteCloud IDE

An updated version of SuiteCloud IDE is available as part of Version 2013 Release 2. SuiteCloud IDE is an Eclipse-based IDE that is packaged for NetSuite platform development. The new version of SuiteCloud IDE includes the following enhancements:

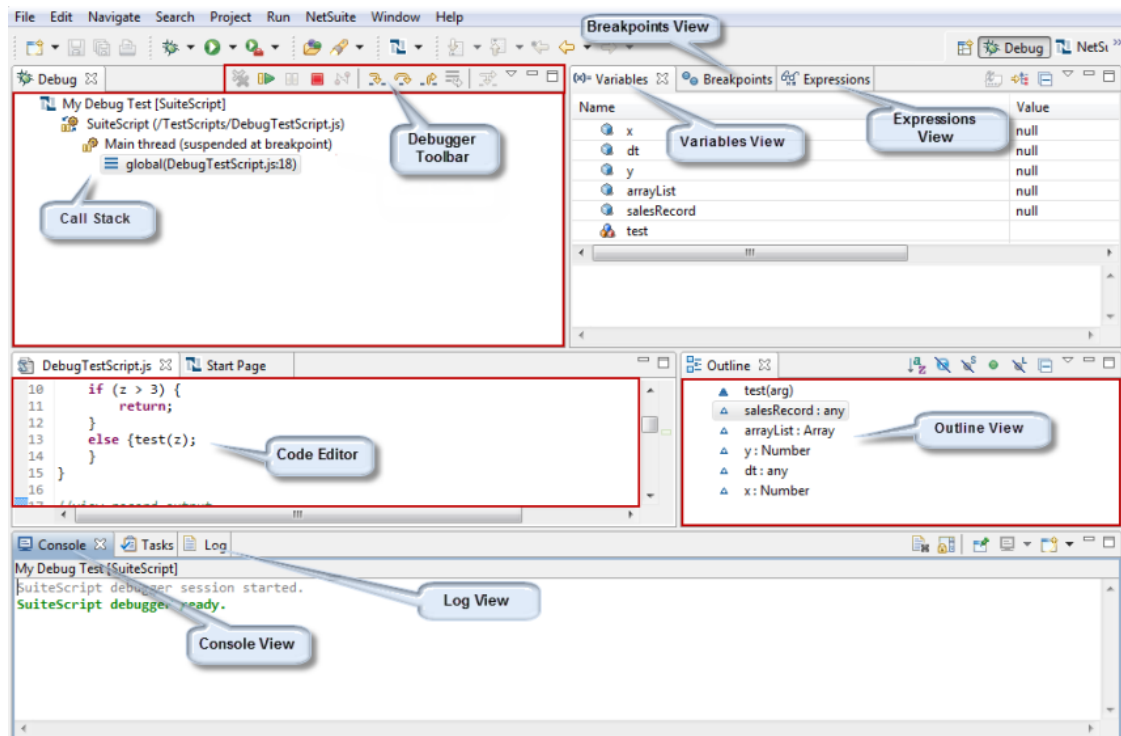
- [Multi-file Cloud Application Debugger](#)
- [Search ID Code Completion](#)
- [File Encoding During File Upload to NetSuite](#)
- [Manage Accounts/Add Accounts/Manage Environments Dialog Column Sorting](#)

Multi-file Cloud Application Debugger

A debugger allows you to interactively control the execution of your code so that you can monitor the progress of its variables and output. In previous releases, SuiteCloud IDE users

had to utilize external applications to debug their SuiteScript code. With NetSuite Version 2013 Release 2, the SuiteCloud IDE includes a new integrated multi-breakpoint cloud-based debugger.

Developers can now debug server-side SuiteScripts within the same Eclipse-based environment they develop in. The debugger utilizes the same keyboard shortcuts and menu options found in the standard version of Eclipse. Developers familiar with the Eclipse Debugger will find the SuiteCloud IDE Debugger easy to use.



The SuiteCloud IDE Debugger provides developers with two debugging modes: Single File Mode (Ad Hoc Debugging) and Project Mode (Deployed Debugging). Both modes support the use of multiple breakpoints and the standard debugging functionality (for example: step through, inspect, watch expression, variable evaluation and log output). Multi-file debugging is only supported with Project Mode. Project Mode also incorporates an integrated browser session for the project's NetSuite account and a client for both Suitelets and RESTlets.

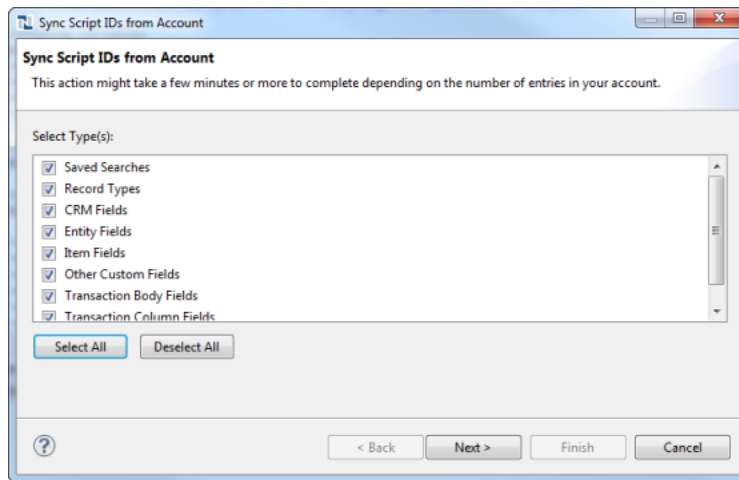
See the help topic [Using the SuiteCloud IDE Debugger](#) for additional information.

Search ID Code Completion

For NetSuite Version 2013 Release 2, the SuiteCloud IDE includes code completion for saved search IDs. This new functionality greatly increases productivity; developers can

now quickly insert search IDs when scripting on a saved search with `nlapiLoadSearch` and `nlapiSearchRecord`.

To support this feature, the **Sync Custom Field and Record ID(s) from Account** menu item is now replaced with the **Sync Script IDs from Account** menu item. When users click **Sync Script IDs from Account** from a project in the NS Explorer View, a new wizard appears that walks them through the synchronization process for record IDs, field IDs and saved search IDs. See the help topic [Synchronizing Internal IDs from a Project Account](#) for additional information.



File Encoding During File Upload to NetSuite

With Version 2013 Release 2, the SuiteCloud IDE supports file encoding upon upload to NetSuite. To facilitate this feature, the Set File Properties dialog has a new Character Encoding property. During upload, the value of this property populates the file's `textFileEncoding` attribute.

Possible values for the Character Encoding property are:

- `_big5`
- `_gb2312`
- `_gb18030`
- `_iso88591`
- `_macRoman`
- `_shiftJis`
- `_utf8`
- `_windows1252`

Note: If left blank, the Character Encoding property is set to _utf8 by default.

See the help topic [Defining SuiteScript File/Folder Properties Before Upload](#) for additional information.

Manage Accounts/Add Accounts/Manage Environments Dialog Column Sorting

For Version 2013 Release 2, SuiteCloud IDE adds column sorting to all columns in the Manage Accounts, Add Accounts, and Manage Environments dialogs. This addition enables users to quickly locate the account or environment they need to manage. See the help topics [Setting Up an Account](#) and [Setting Up an Environment](#) for additional information

Records Exposed to SuiteScript

The following table lists newly supported record types for SuiteScript as of Version 2013 Release 2.

Newly Supported Record Types	Category	Notes
Billing Class	Lists	<ul style="list-style-type: none">• The billing class record allows you to create different rates that you can use when calculating the cost of a resource's time on a project. This record is useful if any given employee might play one of several roles on a project. This record has been enhanced in this release. For more details, see Billing Class Enhancements for Advanced Projects.• The billing class record is available when the Per-Employee Billing Rates feature is enabled at Setup > Company > Enable Features, on the Employees tab.• In the UI, you access the billing class record by choosing Setup > Accounting > Billing Classes > New.• You can interact with the Currency sublist only if the Multiple Currencies feature is enabled, at Setup> Company > Enable Features. If the Multiple Currencies feature is not enabled, then the sublist is replaced by a single body field named Price.• The internal ID for this record is billingclass.
Campaign Template	Marketing	<ul style="list-style-type: none">• A campaign template record allows you to identify a standard format for marketing campaign mail.• To use this record, the Customer Relationship Management must be enabled at Setup > Company > Enable Features, on the CRM tab.• In the UI, you access this record by choosing Documents > Templates > Marketing Templates, then selecting Campaign from the Select Type page.• The internal ID for this record is campaigntemplate.

Newly Supported Record Types	Category	Notes
Charge	Transactions	<ul style="list-style-type: none"> The charge record is used to represent a single billable amount that a client must pay. This feature, which is new in this release, is described further in Charge-Based Billing. The charge record is available only when the Charge-Based Billing feature is enabled at Setup > Company > Enable Features, on the Transactions subtab. In the UI, you access the charge record by choosing Transactions > Customers > Create Charges > List. The internal ID for this record is charge.
Email Template	Marketing	<ul style="list-style-type: none"> An email template record allows you to identify a standard format to be used for automated emails, bulk emails, and other communications sent from NetSuite. To use this record, the Customer Relationship Management must be enabled at Setup > Company > Enable Features, on the CRM tab. In the UI, you access this record by choosing Documents > Templates > Email Templates. The internal ID for this record is emailtemplate.
Inventory Cost Revaluation	Transactions	<ul style="list-style-type: none"> The inventory cost revaluation record is used to activate a standard cost version in production, and also to recalculate the value of inventory items. This record is available when the Standard Costing feature is enabled at Setup > Company > Enable Features, on the Items & Inventory subtab. In the UI, you access the inventory cost revaluation record by choosing Transactions > Inventory > Revalue Inventory Cost. The internal ID for this record is inventorycostrevaluation.
Scheduled Script Instance	Customization	See Advanced Scheduled Script Management Enhancement for additional information.
Script	Customization	See Script and Script Deployment Exposure for additional information.
Script Deployment	Customization	See Script and Script Deployment Exposure for additional information.
Tax Control Account	Lists	<ul style="list-style-type: none"> A tax control account is an account to which the amounts computed for indirect taxes such as sales tax and VAT are posted. In the UI, you can create a tax control account or view existing accounts by choosing Setup > Accounting > Tax Control Accounts. An alternative way of viewing existing tax control accounts is to choose Lists > Accounting > Accounts.

Newly Supported Record Types	Category	Notes
		<ul style="list-style-type: none"> If you are using the Advanced Taxes feature, you must reference an existing nexus when creating a tax control account. Once a tax control account is created, you cannot change the nexus. You can set tax account type only in certain nexuses. A tax control account cannot be deleted. The internal ID for this record is taxacct.

API Additions and Enhancements

The following table summarizes updates to existing APIs. This table also summarizes new objects, methods, and functions that have been added to the SuiteScript API.

API	Note
Updates to Existing APIs	
<code>nlapiCreateRecord(type, initializeValues)</code>	New types defined to support script and script deployment record exposure. See Script and Script Deployment Exposure for additional information.
<code>nlapiDeleteRecord(type, id)</code>	New types defined to support script and script deployment record exposure. See Script and Script Deployment Exposure for additional information.
<code>nlapiLoadRecord(type, id, initializeValues)</code>	New types defined to support script and script deployment record exposure. See Script and Script Deployment Exposure for additional information.
<code>nlapiRequestURLWithCredentials(creds, url, postdata, headers, method)</code>	See Credential Builder API for additional information.
nlobjRecord <ul style="list-style-type: none"> <code>setDateTimeValue(fieldId, dateTime, timeZone)</code> <code>getDateTimeValue(fieldId, timeZone)</code> <code>setCurrentLineItemDateTimeValue(type, fieldId, dateTime, timeZone)</code> <code>getCurrentLineItemDateTimeValue(type, fieldId, timeZone)</code> <code>setLineItemDateTimeValue(type, fieldId, lineNum, dateTime, timeZone)</code> <code>getLineItemDateTimeValue(type, fieldId, lineNum, timeZone)</code> 	See Alternate Time Zone Support for additional information.
New APIs	

API	Note
<code>nlapiGetCurrentLineItemDateTimeValue(type, fieldId, timeZone)</code>	See Alternate Time Zone Support for additional information.
<code>nlapiGetDateTimeValue(fieldId, timeZone)</code>	See Alternate Time Zone Support for additional information.
<code>nlapiGetLineItemDateTimeValue(type, fieldId, lineNum, timeZone)</code>	See Alternate Time Zone Support for additional information.
<code>nlapiSetCurrentLineItemDateTimeValue(type, fieldId, dateTime, timeZone)</code>	See Alternate Time Zone Support for additional information.
<code>nlapiSetDateTimeValue(fieldId, dateTime, timeZone)</code>	See Alternate Time Zone Support for additional information.
<code>nlapiSetLineItemDateTimeValue(type, fieldId, lineNum, dateTime, timeZone)</code>	See Alternate Time Zone Support for additional information.
<code>nlapiVoidTransaction(transactionType, recordId)</code>	See Void Transaction Support for additional information.
<code>nlobjCredentialBuilder(string, domainString)</code> <ul style="list-style-type: none"> • <code>append()</code> • <code>prepend()</code> • <code>base65()</code> • <code>utf8()</code> • <code>sha256()</code> • <code>sha1()</code> 	See Credential Builder API for additional information.

Changed Behavior when Scripting Project Tasks

In Version 2013 Release 2, SuiteScript includes support for milestone tasks. Essentially, a milestone task is a project task record that has a value of zero in the Estimated Work field. Project milestones are used to mark a point in your project, usually completion of a set of tasks or as a project health check to determine if you're on schedule. This feature, which is new in this release, is described more fully in [Milestone Records for Advanced Projects](#).

If you are using SuiteScript to interact with project task records, the behavior mimics what you see in the UI. That is, if you add a project task record using SuiteScript, and you have included no value for the Estimated Work field, the record is saved as a milestone task. Similarly, if you edit a project task record to eliminate all Estimated Work, it is saved as a milestone task. Note that both project tasks and milestone tasks are the same type of record.

See also Project Task in the SuiteScript section of the Help Center.

2013.2 SuiteScript Records Browser

The 2013.2 version of the SuiteScript Records Browser is now available. You can access the browser by clicking [here](#) or by doing one of the following:

- In the NetSuite Help Center and in the *SuiteScript Developer and Reference Guide*, click the links to the Records Browser that are provided in the description of each supported record.
- Open the Help Center and navigate to *SuiteCloud (Customization, Scripting, and Web Services) > SuiteScript > SuiteScript Reference > SuiteScript Reference > Using the SuiteScript Records Browser*. This topic includes a link to the Records Browser.

SuiteTalk (Web Services) Integration

Refer to the following sections for details on all SuiteTalk Web services updates for Version 2013 Release 2:

- [NetSuite to End Support for SuiteTalk 2009.2 and Earlier Endpoints](#)
- [Newly Supported Record Types in SuiteTalk](#)
- [Search Enhancement for Customized Records](#)
- [Enhancements to Tax Group Record Exposure](#)
- [Improved Result Reporting for Failed Asynchronous Operations](#)
- [Enhancement to AsyncInitializeList](#)
- [Renaming of Duplicates Task ID](#)
- [Improved Error Handling for Budget Record](#)
- [Enhancement for Campaign Response Integrations](#)
- [New Endpoint Exposes ScriptId of Saved Searches](#)
- [With New Endpoint, Integrations Referencing Custom Fields Must Be Updated](#)
- [Changed Behavior when Using SuiteTalk with Project Tasks](#)
- [In New Endpoint, LandedCost Complex Type Is Renamed](#)
- [New Abstract Type for Custom Fields](#)
- [In Japan Edition, New Field Exposed on the Customer Record](#)
- [Fields Added to 2013.2 Endpoint to Support New Features](#)
- [Additional Values for Enumerations](#)
- [New and Modified Search Objects](#)

- [2013.2 WSDL Now Generally Available](#)
- [Araxis Merge Diff File for the 2013.2 Endpoint](#)
- [2013.2 Schema Browser](#)

NetSuite to End Support for SuiteTalk 2009.2 and Earlier Endpoints

As of Version 2014 Release 1, NetSuite will no longer support SuiteTalk 2009.2 or earlier endpoints. NetSuite strongly encourages customers and partners using 2009.2 and earlier versions of the SuiteTalk WSDL to begin upgrading to later endpoints.

The end of support for these endpoints will enable NetSuite to focus maintenance on fewer versions, and will enable customers to benefit from more consistent record processing behavior in 2010.1 and later endpoints.

See the following for some tips:

- [Determining Endpoints that are Currently in Use](#)
- [Resources for Endpoint Upgrades](#)
- [Note About Record Processing Behavior in 2010.1 and Later Endpoints](#)

Determining Endpoints that are Currently in Use

Use the following guidelines to determine whether any of your SuiteTalk integrations use the 2009.2 or earlier endpoints:

- For each partner application, work with the partner directly to find out the endpoint in use.
- For your own integrations:
 - You can check SOAP requests to see which endpoints they reference. You can find links to these requests at Setup > Integration > Web Services Usage Log, in the Request column.
 - You can contact NetSuite Support to ask for help in determining the endpoints used in your account.

Resources for Endpoint Upgrades

To begin the process of upgrading your endpoint, refer to the following resources:

- For the schema of the latest endpoint, see:

https://webservices.netsuite.com/wSDL/v2013_2_0/netsuite.wsdl.

- For details about the records supported in the latest endpoint, see the help topic SuiteTalk Records Overview.
- For general information about using SuiteTalk, see the help topic SuiteTalk Platform Overview.
- For upgrade tips, see the help topic Upgrading WSDL Versions.
- If you have issues during your transition process, please contact NetSuite Technical Support.

Note About Record Processing Behavior in 2010.1 and Later Endpoints

Be aware of the following change: When using the 2009.2 and earlier endpoints, most, but not all, of the NetSuite core business logic is executed when you modify a record. For integrations that use an endpoint later than 2009.2, *all* core business logic executes when the record is submitted. So you may notice some changes in SuiteTalk behavior for integrations that are upgraded to use an endpoint later than 2009.2. Following your upgrade, the behavior will more closely match the behavior when records are modified through the NetSuite user interface.

Newly Supported Record Types in SuiteTalk

The following table lists the newly supported record types in the 2013.2 endpoint. Click the links to see help topics with more detail about working with specific record types in Web services.

Newly Supported Record Types	Category	Notes
Charge	Transactions	<ul style="list-style-type: none">• The charge record is used to represent a single billable amount that a client must pay.• This feature, which is new in this release, is described further in Charge-Based Billing.• The charge record is available when the Charge-Based Billing feature is enabled at Setup > Company > Enable Features, on the Transactions subtab.• In the UI, you access the charge record by choosing Transactions > Customers > Create Charges > List.• This record is defined in the tranCust (customers) XSD, where it is called Charge.
Inventory Cost Revaluation	Transactions	<ul style="list-style-type: none">• The inventory cost revaluation record is used to activate a standard cost version in production, and also to recalculate the value of inventory items.• This record is available when the Standard Costing feature is enabled at Setup > Company > Enable Features, on the Items & Inventory subtab.

Newly Supported Record Types	Category	Notes
		<ul style="list-style-type: none"> In the UI, you access the inventory cost revaluation record by choosing Transactions > Inventory > Revalue Inventory Cost. This record is defined in the tranInvt (inventory) XSD, where it is called InventoryCostRevaluation.
Tax Control Account	Other Lists	<ul style="list-style-type: none"> A tax control account is an account to which the amounts computed for indirect taxes such as sales tax and VAT are posted. In the UI, you access the tax control account by choosing Setup > Accounting > Tax Control Accounts. An alternative way of viewing existing tax control accounts is to choose Lists > Accounting > Accounts. If you are using the Advanced Taxes feature, you must reference an existing nexus when creating a tax control account. Once a tax control account is created, you cannot change the nexus. You can set tax account type only in certain nexuses. A tax control account cannot be deleted. The tax control account record is defined in the listAcct (accounting) XSD, where it is called taxAcct.

Search Enhancement for Customized Records

The 2013.2 WSDL includes an enhancement related to searchable records for which customFieldList is exposed to SuiteTalk. In other words, these are records that can be customized and which include, among their exposed fields, a custom field that you have added.

For these records, if the customized field is a field of type list/record or multiple select, and if the field references another record, the 2013.2 WSDL allows your search of the customized record to include a join to the SearchBasic of the referenced record. Further, with an advanced search of the customized record, your search results can include columns from the referenced record.

For example, suppose your employee record has been altered to include a custom field that points to the account record. With the 2013.2 WSDL, you can search for all employees associated with an account that has a balance over a particular amount. You can do this in either of the following two ways:

- Using EmployeeSearch — Results returned would include the entire employee record for each employee that meets your criteria.
- Using EmployeeSearchAdvanced — Results returned could include any columns you specify from the employee record or the account record.

Additionally, note that you can join to and from custom record types. For example, suppose your organization used custom record types called Customer Survey and Customer Suggestion.

If Customer Survey and Customer Suggestion both included fields that referenced each other, a search of Customer Survey could include a join to Customer Suggestion and vice versa.

Similarly you can join between the following:

- A customized record type, such as Customer Survey, if it includes a reference to a standard record type, such as the employee record.
- A standard record, such as the employee record, if it has been altered to include a field referencing a custom record type.

With this enhancement, searching is more efficient, as you can gather more data with a single Web services call.

Note: This change affects only customers who upgrade to the 2013.2 WSDL.

For more details, see:

- [Schema Changes to Support New Search Feature](#)
- [Example – Using CustomSearchJoin and CustomSearchRowBasic](#)
- [Limitations of New Search Feature](#)

Schema Changes to Support New Search Feature

To support this enhancement, the 2013.2 WSDL includes the following changes:

- [Changes in Common.xsd](#)
- [Changes to Core.xsd, Changed Inheritance](#)

Changes in Common.xsd

Common.xsd includes two new elements: CustomSearchJoin and CustomSearchRowBasic.

CustomSearchJoin adds the ability to include search criteria from the record that is referenced by the custom field. This join is defined as follows:

```
<complexType name="CustomSearchJoin">
  <sequence>
    <element name="customizationRef" type="platformCore:CustomizationRef"/>
    <element name="searchRecordBasic" type="platformCore:SearchRecordBasic"/>
  </sequence>
</complexType>
```

CustomSearchRowBasic adds the ability for advanced searches to include, in the search results, columns from a record referenced by a custom field. This element is defined as follows:

```
<complexType name="CustomSearchRowBasic">
```

```
<sequence>
<element name="customizationRef" type="platformCore:CustomizationRef"/>
<element name="searchRowBasic" type="platformCore:SearchRowBasic"/>
</sequence>
</complexType>
```

Notice that both of these elements use CustomizationRef, which you use to identify the custom field. CustomizationRef includes five possible attributes. Of these, you can use only internal ID or script ID to identify a custom field. The other attributes (external ID, type, and name) are not applicable to custom fields.

The following example shows how you can use script ID to reference a custom field and join to accountSearchBasic:

```
CustomizationRef field_account = new CustomizationRef(null,null,null,null,"custentity_account")
;
CustomSearchJoin customSearchJoin = new CustomSearchJoin(field_account,accountSearchBasic);
```

Changes to Core.xsd, Changed Inheritance

Two new abstract types have been added to Core.xsd. These changes have no effect on pre-2013.2 WSDLs and are described here only because they might be of interest to advanced users with in-depth knowledge of the schema.

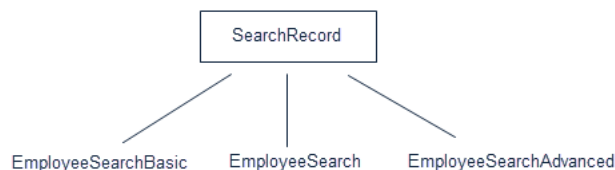
The new abstract types have been added to prevent the use of nested joins. These types are:

- [SearchRecordBasic](#)
- [SearchRowBasic](#)

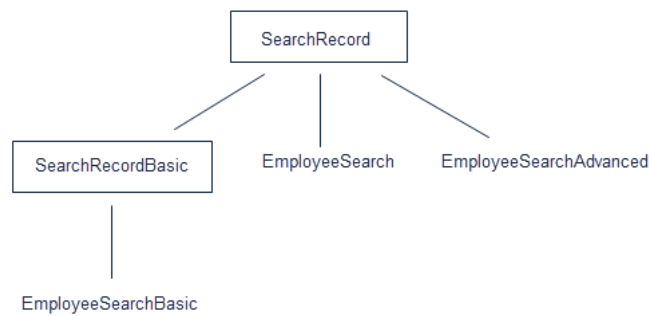
SearchRecordBasic

SearchRecordBasic extends SearchBasic. By contrast, in older versions of the schema, each record's SearchBasic (for example, EmployeeSearchBasic) was an extension of SearchRecord.

The following illustration shows how the Employee search records were inherited in 2013.1 and previous endpoints:



In 2013.2, the inheritance is as follows:



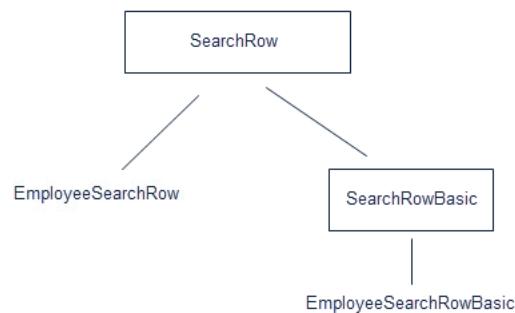
SearchRowBasic

SearchRowBasic extends **SearchRow**. By contract, in older versions of the schema, each record's **SearchRowBasic** (for example, **EmployeeSearchRowBasic**) was not inherited from any other element.

The follow illustration shows the inheritance structure in 2013.1 and previous endpoints:



In 2013.2, the inheritance is as follows:



Example – Using CustomSearchJoin and CustomSearchRowBasic

Suppose you customize the employee record to include a field that identifies an account record. The following example shows how you might use EmployeeSearchAdvanced to find employee records associated with an account of a certain type and value:

```
public void testSearch1() throws Exception{

    // We are searching for employees associated with a certain type of account. Here we specify that account.

    AccountSearchBasic accountSearchBasic = new AccountSearchBasic();
    accountSearchBasic.setType(new SearchEnumMultiSelectField(new String[]{AccountType.__bank}, SearchEnumMultiSelectFieldOperator.anyOf));

    // Here we specify the field on the employee record that references the account record.

    CustomizationRef field_account = new CustomizationRef();
    field_account.setScriptId("custentity_account");

    // Here we create a custom search join object.
    CustomSearchJoin customSearchJoin = CustomSearchJoin(field_account, accountSearchBasic);

    // Next, we pass the custom search join that we just created into an EmployeeSearch instance.

    EmployeeSearch employeeSearch = new EmployeeSearch();
    employeeSearch.setCustomSearchJoin(new CustomSearchJoin[] {customSearchJoin});

    // Next, we specify that we want the results to include the balance of the account.

    AccountSearchRowBasic accountSearchRowBasic = new AccountSearchRowBasic();
    accountSearchRowBasic.setBalance(new SearchColumnDoubleField[]{new SearchColumnDoubleField()});

    CustomSearchRowBasic customSearchRowBasic = new CustomSearchRowBasic(field_account, accountSearchRowBasic);

    // Here, we specify that we want the employee's internal ID included in the results.

    EmployeeSearchRowBasic employeeSearchRowBasic = new EmployeeSearchRowBasic();
    employeeSearchRowBasic.setInternalId(new SearchColumnSelectField[]{new SearchColumnSelectField()});

    // In the following snippet, we pass the specified search columns into an EmployeeSearchRow instance.

    EmployeeSearchRow employeeSearchRow = new EmployeeSearchRow();
    employeeSearchRow.setCustomSearchJoin(new CustomSearchRowBasic[]{customSearchRowBasic});
    employeeSearchRow.setBasic(employeeSearchRowBasic);

    // Create the advanced search.

    EmployeeSearchAdvanced employeeSearchAdvanced = new EmployeeSearchAdvanced();
    employeeSearchAdvanced.setCriteria(employeeSearch);
    employeeSearchAdvanced.setColumns(employeeSearchRow);

    c.search(employeeSearchAdvanced);
}
```

```
}
```

Limitations of New Search Feature

Refer to the following sections for details on limitations of this feature.

- [Use CustomSearchJoin Only With Custom Fields](#)
- [CustomSearchJoin Connects Only With SearchBasics](#)
- [Nested Joins Are Not Allowed](#)

Use CustomSearchJoin Only With Custom Fields

You cannot use CustomSearchJoin in conjunction with a standard field that references another record. For example, the standard employee record includes the department field. If you decide to use CustomSearchJoin, and you identify the department field as your custom field, the system returns an error reading “Invalid custom field reference.” You should use the standard departmentJoin that is already provided as part of EmployeeSearch and EmployeeSearchAdvanced.

CustomSearchJoin Connects Only With SearchBasics

CustomSearchJoin allows only for a basic search of the referenced record. For example, if the employee record has been altered to include a custom field that points to the account record, you can join to AccountSearchBasic – but not AccountSearch or AccountSearchAdvanced. This limitation is consistent with the capabilities of searching in the UI.

Nested Joins Are Not Allowed

As noted in [Changes to Core.xsd](#), [Changed Inheritance](#), you cannot use nested joins. For example, suppose your contact record includes a custom field referencing the employee record, which includes a custom field that references the account record. A search of the contact record could include a join to the employee record, but the same search could not *also* join to the account record.

Enhancements to Tax Group Record Exposure

Version 2013 Release 2 includes the following enhancements to tax group record exposure in Web services:

- In the 2013.2 endpoint only, the Subsidiary field is now exposed.
- For all endpoints, the Version 2013 Release 2 changes the result of a get operation for the tax group record. Specifically, a get response will now include nexusCountry.

Improved Result Reporting for Failed Asynchronous Operations

Version 2013 Release 2 includes an enhancement related to asynchronous operations that fail during processing because of an unexpected internal error. It is now possible to retrieve detailed results that show which records were processed and which were not attempted.

Important: All accounts will have this change after the upgrade to Version 2013 Release 2, regardless of the endpoint in use.

For details, see:

- [Viewing the Status of an Asynchronous Job](#)
- [Viewing Detailed Results of Asynchronous Operations](#)
- [Operations that Use Improved Result Reporting](#)

Viewing the Status of an Asynchronous Job

To start, use `checkAsyncStatus` to find out the status of the job. If the status is `FINISHEDWITHERRORS`, that indicates that at least one record was processed, but some records failed.

One difference between Version 2013 Release 2 and prior releases is that, previously, if an unexpected internal error occurred during the processing of the job, the operation was marked as failed. The response could not be retrieved, and it was impossible to determine which records (if any) had been processed. Now, for these cases, the system uses the `FINISHEDWITHERRORS` status, and the response can be retrieved.

As an alternative to using `checkAsyncStatus`, you can view the status in the UI at Setup > Integration > Web Services Process Status.

Viewing Detailed Results of Asynchronous Operations

When a job finishes with errors, you can find out which records failed using `getAsyncResult`. If any record was not processed, it is identified in the `writeResponse`. The response might include details like the following. Note that records (for example, “record number 3” below) are numbered according to their order in the SOAP request:

```
<writeResponse>
  <platformCore:status isSuccess="false" xmlns:platformCore="urn:core_2013_2.platform.webservice.s.netsuite.com">
    <platformCore:statusDetail>
      <platformCore:code>UNEXPECTED_ERROR</platformCore:code>
      <platformCore:message>Unexpected error occurred. The record number 3 has not been processed. Please resubmit the record.</platformCore:message>
    </platformCore:statusDetail>
```

```
</platformCore:status>  
</writeResponse>
```

To retrieve the response, you can use code like the following:

```
AsyncAddListResult result = (AsyncAddListResult)port.getAsyncResult("job ID", 1);  
WriteResponseList responses = result.getWriteResponseList();  
for (WriteResponse response : responses.getWriteResponse())  
{  
    if (!response.getStatus().isIsSuccess())  
    {  
        if (response.getStatus().getStatusDetail()[0].getCode() == StatusDetailCodeType.UNEXPECTED_ER  
ROR)  
            // add for resubmit  
        }  
    }  
}
```

Alternatively, you can see some details about specific record failures by navigating to Setup > Integration > Web Services Process Status and viewing the Response.xml file.

Operations that Use Improved Result Reporting

This enhancement affects all asynchronous operations that modify the database. These are:

- asyncAddList
- asyncUpdateList
- asyncUpsertList
- asyncDeleteList

Enhancement to AsyncInitializeList

Because of a change in the 2013.2 endpoint, you can now use GetAsyncResult to view details about AsyncInitializeList jobs. In previous endpoints, when used with AsyncInitializeList, GetAsyncResult failed with an exception.

Renaming of Duplicates Task ID

In this release, the LIST_DUPLICATES task ID has been renamed to LIST_ENTITY_DUPLICATES. If you have an integration that uses this task ID, make sure you update this name. Note also that the menu item name has changed from Mass Duplicate Resolution to Entity Duplicate Resolution. (The full path is Lists > Mass Update > Entity Duplicate Resolution.) The URL remains the same: /app/common/entity/manageduplicates.nl. This page is related to the Duplicate Detection & Merge feature. For more on this feature, see the help topic Duplicate Record Detection. For more details on task IDs, see the help topic Task IDs.

Improved Error Handling for Budget Record

For accounts using the 2013.2 endpoint, the system returns a more descriptive error code when a user makes a mistake while attempting to work with a budget record.

With previous endpoints, if the user failed to enter an amount for at least one of the periodAmount fields (such as periodAmount1, periodAmount2, and so on), the system returned an error code of USER_ERROR. If you upgrade to the new endpoint, the system returns a code of NO_UPSERT. The text of the error message remains the same ("No budget inserted or updated").

Enhancement for Campaign Response Integrations

In Version 2013 Release 2, changes have been made to SuiteTalk to reflect enhancements to the campaign response record. These changes affects all integrations, regardless of which endpoint is being used.

This enhancement allows an integration to set a specific time indicating when a campaign response occurred. For more details on this change, see the help topic Campaign Response.

New Endpoint Exposes ScriptId of Saved Searches

When using the 2013.2 WSDL, the getSavedSearch operation now returns the scriptId of a saved search. In previous endpoints, this operation returned only the internalId and the name of the search.

The following example shows a SOAP response to the getSavedSearch operation when the new WSDL is used:

```
<soapenv:Body>
  <getSavedSearchResponse xmlns="urn:messages_2013_2.platform.webservices.netsuite.com">
    <platformCore:getSavedSearchResult xmlns:platformCore="urn:core_2013_2.platform.webservices.netsuite.com">
      <platformCore:status isSuccess="true"/>
      <platformCore:totalRecords>1</platformCore:totalRecords>
      <platformCore:recordRefList>
        <platformCore:recordRef scriptId="customsearch_txn_search_salesorder" internalId="14" xsi:type="platformCore:CustomizationRef">
          <platformCore:name>Feature 2432 - Sales Order Search</platformCore:name>
        </platformCore:recordRef>
      </platformCore:recordRefList>
    </platformCore:getSavedSearchResult>
  </getSavedSearchResponse>
</soapenv:Body>
```

The specific change that allows this enhancement involves the CustomizationRef complex type. In the 2013.1 and earlier endpoints, CustomizationRef was inherited from BaseRef. Now, CustomizationRef is inherited from RecordRef. Because the getSavedSearch API returns RecordRef, the new endpoint allows for the use of scriptId, which is part of CustomizationRef.

With New Endpoint, Integrations Referencing Custom Fields Must Be Updated

If your SuiteTalk integration includes interaction with custom fields, be aware of the following changes.

InternalId Is Renamed

In the 2013.2 endpoint, the internalId element of certain custom field types is being renamed. Specifically, “internalId” is being renamed “scriptId.” Another way of describing the change is that the value previously called internalId is now called scriptId.

Therefore, if your SuiteTalk integration references any of the affected custom field types and you upgrade to the 2013.2 WSDL, **you must edit all references to the “internalId” of these fields in order for your integration to continue working.** The simplest solution is to rename “internalId” to “scriptId.” If you are not upgrading to the new WSDL, you do not have to do anything.

Note that the value in question is labeled “ID” in the UI (on the record representing the custom field). The actual values are strings in the following style: “custentity_field,” “custitem_field,” and so on.

This change applies to custom fields, custom field references, search custom fields, and search column custom fields. The exact field types affected are described in the table below. Fields not listed in this table are not changing.

Category	Field
Custom fields	<ul style="list-style-type: none">• CrmCustomField• CustomRecordCustomField• EntityCustomField• ItemCustomField• ItemNumberCustomField• ItemOptionCustomField• OtherCustomField• ScriptCustomField• TransactionBodyCustomField• TransactionColumnCustomField
Custom field references	<ul style="list-style-type: none">• BooleanCustomFieldRef• CustomFieldRef (abstract)

Category	Field
	<ul style="list-style-type: none"> • DateCustomFieldRef • DoubleCustomFieldRef • LongCustomFieldRef • MultiSelectCustomFieldRef • SelectCustomFieldRef • StringCustomFieldRef
Search custom fields	<ul style="list-style-type: none"> • SearchBooleanCustomField • SearchCustomField (abstract) • SearchDateCustomField • SearchDoubleCustomField • SearchEnumMultiSelectCustomField • SearchLongCustomField • SearchMultiSelectCustomField • SearchSelectCustomField • SearchStringCustomField
Search column custom fields	<ul style="list-style-type: none"> • SearchColumnBooleanCustomField • SearchColumnCustomField (abstract) • SearchColumnDateCustomField • SearchColumnDoubleCustomField • SearchColumnEnumMultiSelectCustomField • SearchColumnLongCustomField • SearchColumnMultiSelectCustomField • SearchColumnSelectCustomField • SearchColumnStringCustomField

A related change is that these fields will now support a new “Internal ID” field. This value is numeric. In the 2013.2 WSDL, this value is identified as the `internalId` element. For more details on this enhancement, see [Change to Custom Field IDs](#).

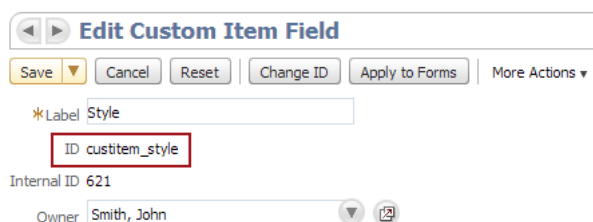
Both of these changes – the renaming of `internalId` to `scriptId` and the addition of the new `internalId` element – are intended to make the structure of custom fields more consistent with the structure of custom records.

Again, if you are not upgrading the 2013.2 WSDL, or if your integration does not reference these field types, you do not have to do anything. However, if you *are* upgrading to the new WSDL, you must edit any references in your integration to these fields. NetSuite recommends renaming “internalId” to “scriptId.” Alternatively, you could keep references to internalId, but edit the values for each of these references to reflect the newly created internal ID values, which will be numeric. **If you fail to make any adjustment, references to these fields in your integration will stop working.**

In SOAP Responses, Prefix Is Dropped

With the 2013.2 endpoint, the way the system identifies custom fields in certain SOAP responses is different from the behavior associated with prior endpoints. This change applies to scenarios in which you execute an advanced search, join to a record that has a custom field, and include the custom field in your search results. With 2013.1 and earlier endpoints, the system in its response added a prefix to the string used to identify the custom field. In 2013.2, this prefix is not added.

For example, suppose you had a custom item field on your inventory item record, and this custom field had an ID of `custitem_style`.



First, note that this value is referred to as `internalId` when using 2013.1 endpoint or an earlier endpoint. With the 2013.2 endpoint, the value is called `scriptId`. (The renaming of `internalId` to `scriptId` is described in [InternalId Is Renamed](#).)

Second, in some cases, the system modifies this value slightly. For example: Suppose you executed a `TransactionSearchAdvanced` and joined to the item record. If you included this custom field in your search results, the 2013.1 endpoint would identify the field’s `internalId` as having a value of “`item_custitem_style`.” That is, in this particular scenario, the system would add the “`item_`” prefix. With a different type of join, a different prefix would be added.

If you are using the 2013.2 endpoint, the system does **not** add a prefix. The `scriptId` returned in the response is the same as in the request (“`custitem_style`,” in this example). Therefore, if you upgrade to the 2013.2 WSDL and if you have integrations that depend on this value, you must modify these integrations.

To see sample SOAP responses, read the help topic [How Custom Fields Are Identified in Search Results](#).

Changed Behavior when Using SuiteTalk with Project Tasks

Version 2013 Release 2 of SuiteTalk includes support for milestone tasks. While a project task is used to represent an activity, a milestone task is used to represent a checkpoint in the overall progress of the project. From a technical perspective, a milestone task is a project task record that has a value of zero in the estimatedWork body field. This feature, which is new in this release, is described more fully in [Milestone Records for Advanced Projects](#).

If you are using SuiteTalk to interact with project task records, the behavior mimics what you see in the UI. That is, if you add a project task record with no estimated work, the record is saved as a milestone task. Note that both project tasks and milestone tasks are the same type of record, so you can get and delete all tasks in the same way, regardless of whether they are project tasks or milestone tasks.

This new behavior is the same for all accounts, regardless of which endpoint you are using.

If you are upgrading to the 2013.2 WSDL, see also [Fields Added to 2013.2 Endpoint to Support New Features](#), for additional details on other enhancements to the exposure of this record.

See also Project Task in the SuiteTalk section of the Help Center.

In New Endpoint, LandedCost Complex Type Is Renamed

In the 2013.2 WSDL, the LandedCost complex type has been renamed LandedCostSummary. If you upgrade to the new WSDL, and your integration references this element, your integration must be updated to use the new name. Note that this element remains part of [common.xsd](#).

Version 2013 Release 2 also introduces a new feature called [Landed Cost Allocation Per Line](#), which includes the addition of a new Landed Cost subrecord. This subrecord is also exposed to SuiteTalk in the 2013.2 WSDL. For more details, see [Fields Added to 2013.2 Endpoint to Support New Features](#).

New Abstract Type for Custom Fields

In the 2013.2 WSDL, a new abstract type has been added called CustomFieldType. This element contains the fieldType and scriptId attributes. The various custom fields types – such as EntityCustomField, ItemCustomField, and so on – all inherit from the new CustomFieldType. This change affects integrations that invoke any of the custom field classes using a constructor that takes arguments, as opposed to an empty constructor. Specifically, if you use a constructor that takes arguments, you should be aware that the arguments are expected in a different order than was the case with 2013.1 and previous endpoints.

CustomFieldType is defined in [setup.customization.xsd](#).

In Japan Edition, New Field Exposed on the Customer Record

In the 2013.2 WSDL, an additional field has been exposed on the customer record to support the Japan Edition of NetSuite. In the UI, this field is labeled Monthly Closing Date, and it is located on the Financial tab, in the Account Information area. This field is referred to as monthlyClosing in [lists.relationships.xsd](#). This field is used only in the Japan Edition.

Fields Added to 2013.2 Endpoint to Support New Features

The section lists additions to the 2013.2 endpoint to support features introduced in this release. This list is not intended to be a comprehensive reference of fields added to the new endpoint. For full details, refer to the 2013.2 WSDL, the 2013.2 Schema Browser, and the 2013.2 Araxis Merge Diff File.

Customer Relationship Management

The following sublist has been added to support the [Issue Relationship Management](#) feature:

Record	Fields Added to 2013.2 Endpoint	Label in UI	Location of Field	Notes	XSD
Issue	relatedIssuesList	Issues	The Related Records subtab	In the UI, you can find this sublist by choosing the Related Records tab.	lists.support.xsd

Enterprise Resource Planning

The following fields have been added to support ERP features:

For [Charge-Based Billing](#):

Record	Fields Added to 2013.2 Endpoint	Label in UI	Location of Field	XSD
Cash refund	chargeType	Charge Type	Items sublist	transactions.customers.xsd
	chargesList	Charges		
Cash sale	chargeType	Charge Type	Items sublist	transactions.sales.xsd
	chargesList	Charges		
Credit memo	chargeType	Charge Type	Items sublist	transactions.customers.xsd
	chargesList	Charges		
Estimate	chargeType	Charge Type	Items sublist	transactions.sales.xsd
Invoice	chargeType	Charge Type	Items sublist	transactions.sales.xsd
	chargesList	Charges		

Record	Fields Added to 2013.2 Endpoint	Label in UI	Location of Field	XSD
Sales order	chargeType	Charge Type	Items sublist	transactions.sales.xsd

For [Attaching Line Item Receipts to Expense Reports](#):

Record	Fields Added to 2013.2 Endpoint	Label in UI	Location of Field	XSD
Expense report	expMediaItem	Attach File	Expenses sublist	transactions.employees.xsd

For [Expense Report Line Item Reference Numbers](#):

Record	Fields Added to 2013.2 Endpoint	Label in UI	Location of Field	XSD
Expense report	refNumber	Ref No.	Expenses sublist	transactions.employees.xsd

For [Quantities and Rates for Expense Categories](#):

Record	Fields Added to 2013.2 Endpoint	Label in UI	Location of Field	Notes	XSD
Expense report	quantity	Quantity	Expenses sublist	These two fields are available only if the category for the expense has the Rate is Required option selected. For details, see the help topic Creating an Expense Category .	transactions.employees.xsd
	rate	Rate			
Expense category	rateRequired	Rate Is Required	Body of record		lists.accounting.xsd
	defaultRate	Default Rate	Body of record	Non-OneWorld accounts only	
	ratesList	Rates sublist	Rates sublist	OneWorld accounts only	

For [Component Yield Preferences](#):

Record	Fields Added to 2013.2 Endpoint	Label in UI	Location of Field	XSD
Work order	componentYield	Component Yield	Items sublist	transactions.inventory.xsd
	bomQuantity	BOM Quantity		
Assembly item (all types)	componentYield	Component Yield	Members sublist	lists.accounting.xsd

Record	Fields Added to 2013.2 Endpoint	Label in UI	Location of Field	XSD
	bomQuantity	BOM Quantity		
Inventory item (all types), assembly item (all types)	roundUpAsComponent	Round Up Quantity as Component	Body of record	
Assembly item (all types)	useComponentYield	Use Component Yield	Body of record	

For Standard Cost Material Overheads:

Record	Fields Added to 2013.2 Endpoint	Label in UI	Location of Field	Notes	XSD
Inventory item (all types)	inventoryCostTemplate	Inventory Cost Template	Locations sublist	Available only when the supply type is buy or inventory item. The cost method must be standard.	lists.accounting.xsd
Inventory item (all types, assembly item (all types)	inventoryCostTemplate	Inventory Cost Template	Locations sublist	Available only when the supply type is buy or inventory item. The cost method must be standard.	
	costingLotSize	Costing Lot Size			
Item (Other Charge for Purchase)	overheadType	Overhead Type	Body of record	Available only when the cost category is Material Overhead or Landed Cost.	

For Landed Cost Allocation Per Line:

Record	Fields Added to 2013.2 Endpoint	Label in UI	Location of Field	XSD	Notes
Check	landedCostPerLine	Landed Cost per Line	Body of record	transactions.bank.xsd (landedCost subrecord is defined in common.xsd)	See also In New Endpoint , LandedCost Complex Type Is Renamed.
	landedCost (this label represents both a field and a new subrecord)	Landed Cost	Items sublist		
Item receipt	landedCostPerLine	Landed Cost per Line	Body of record	transactions.purchases.xsd (landedCost subrecord is defined in common.xsd)	
	landedCost (this label represents both a field and a new subrecord)	Landed Cost	Items sublist		

Record	Fields Added to 2013.2 Endpoint	Label in UI	Location of Field	XSD	Notes
Vendor bill	landedCostPerLine	Landed Cost per Line	Body of record		
	landedCost (this label represents both a field and a new subrecord)	Landed Cost	Items sublist		

For [Exclude from Shipping](#):

Record	Fields Added to 2013.2 Endpoint	Label in UI	Location of Field	XSD
Sales order	excludeFromRateRequest	Exclude Item from Rate Request	Items sublist	transactions.sales.xsd

Projects

The following fields have been added to support the [Expanded Task Scheduling for Advanced Projects](#) feature:

Record	Fields Added to 2013.2 Endpoint	Label in UI	Location of Field	Notes	XSD
Project task	finishbydate	Finish No Later Than	Body of record	See also Changed Behavior when Using SuiteTalk with Project Tasks .	activities.scheduling.xsd
	lagdays	Lag Time	Predecessors sublist		

SuiteBuilder — Customization

The following fields have been added to support SuiteBuilder features.

For [Translation of Custom Record Instance Names](#):

Record	Fields Added to 2013.2 Endpoint	Label in UI	Location of Field/Sublist	Notes	XSD
Custom record type	enableNameTranslation	Enable Name Translation	Body of record		setup.customization.xsd
Custom record	translationsList	Translation	Translation tab	Available only when the Enable Name Translation option has been selected for the custom record type.	

For Custom Fields with Values Derived from Summary Search Results:

Record	Fields Added to 2013.2 Endpoint	Label in UI	Location of Field	Notes	XSD
Custom field (all types except Transaction Column Custom Field)	searchDefault	Search	Body of record (the Validation & Defaulting tab)	When the bodyFieldsOnly search preference is set to true, these fields are not returned. For details on search preferences, see the help topic Setting Search Preferences.	setup.customization.xsd
	searchCompareField	Field			

Additional Values for Enumerations

The 2013.2 endpoint includes the following new enumerations:

New Enumeration Object	XSD
ChargeStage	transactions.customerTypes.xsd
ChargeUse	
ItemCostAccountingStatus	lists.accountingTypes.xsd
ItemOverheadType	
IssueRelationship	lists.supportTypes.xsd
TaxAcctType	lists.accountingTypes.xsd
TransactionChargeType	transactions.saleTypes.xsd

See the 2013.2 Schema Browser for lists of enumeration values.

New and Modified Search Objects

The following are new search objects in the 2013.2 WSDL:

New Search Object	XSD
ChargeSearchBasic, ChargeSearchRowBasic	common.xsd
ChargeSearch, ChargeSearchAdvanced, ChargeSearchRow	transactions.customers.xsd
CustomListSearchAdvanced	setup.customization.xsd
InventoryNumberBinSearchBasic, InventoryNumberBinSearchRowBasic	common.xsd
ItemDemandPlanSearchAdvanced, ItemSupplyPlanSearchAdvanced	transactions.demandPlanning.xsd
ItemBinNumberSearchBasic, ItemBinNumberSearchRowBasic	common.xsd
PayrollItemSearchAdvanced	lists.employees.xsd

The 2013.2 endpoint also includes new values in the following searches:

Modified Search Object	File
ContactSearch, ContactSearchRow	lists.relationships.xsd
CurrencyRateSearch, CurrencyRateSearchRow	lists.accounting.xsd
CustomerSearch, CustomerSearchRow	lists.relationships.xsd
CustomerSearchBasic, CustomerSearchRowBasic	common.xsd
ExpenseCategorySearchBasic, ExpenseCategorySearchRowBasic	
ItemSearch, ItemSearchRow	lists.accounting.xsd
ItemSearchBasic, ItemSearchRowBasic	common.xsd
ManufacturingOperationTaskSearch, ManufacturingOperationTaskSearchRow	lists.supplyChain.xsd
OpportunitySearch, OpportunitySearchRow	transactions.sales.xsd
PhoneCallSearch, PhoneCallSearchRow	activities.scheduling.xsd
ProjectTaskAssignmentSearchBasic, ProjectTaskAssignmentSearchRowBasic	common.xsd
ProjectTaskSearchBasic, ProjectTaskSearchRowBasic	
TransactionSearch, TransactionSearchRow	transactions.sales.xsd
TransactionSearchBasic, TransactionSearchRowBasic	common.xsd

The 2013.2 endpoint also includes new functionality related to searching custom records. For details see, [Search Enhancement for Customized Records](#).

2013.2 WSDL Now Generally Available

A new version of SuiteTalk is available with Version 2013 Release 2. Click [here](#) to view the 2013.2 WSDL.

Araxis Merge Diff File for the 2013.2 Endpoint

The [Araxis Merge Diff file](#) highlights all 2013.1 - 2013.2 schema changes.

Depending on the scope of the release, these changes may include:

- New record types
- Elements (fields) that have been added, removed or changed since 2013.1
- New sublists (lists)
- New joins
- New search filters and search return columns
- New platform operations

Using the Araxis Diff File

Important: The figures in this section are for illustrative purposes **only**. They do NOT reflect schema changes related to any specific endpoint.

The SuiteTalk diff file is generated using Araxis Merge. The following figure shows an Araxis Merge Comparison Report, which is a summary page that highlights all schemas that have been modified for a release. Diff files that are attached to NetSuite Release Notes will always open to the summary page.

Schemas that are highlighted in blue have been modified. Click individual schema links or a Report link to see all changes to a particular schema.

This column represents the previous endpoint.	This column represents the newest endpoint.	Click the Report links to see the differences.
6. Comparison detail		
activities.scheduling.xsd	5 activities.scheduling.xsd	Report 1
activities.schedulingTypes.xsd	0 activities.schedulingTypes.xsd	
documents.fileCabinet.xsd	4 documents.fileCabinet.xsd	Report 2
documents.fileCabinetTypes.xsd	0 documents.fileCabinetTypes.xsd	
general.communication.xsd	2 general.communication.xsd	Report 3
general.communicationTypes.xsd	0 general.communicationTypes.xsd	
lists.accounting.xsd	118 lists.accounting.xsd	Report 4
lists.accountingTypes.xsd	1 lists.accountingTypes.xsd	Report 5
lists.employees.xsd	5 lists.employees.xsd	Report 6
lists.employeeTypes.xsd	0 lists.employeeTypes.xsd	

The following figure shows a snapshot of **documents.fileCabinet.xsd**. The elements highlighted in green are the changed elements. This figure shows:

- The **mediaFile** element has been added to the new endpoint.
- The **isFeatured** element that existed in the previous endpoint been deprecated in the new endpoint.

This column represents the previous endpoint.	This column represents the newest endpoint.
<pre> <element name="url" type="xsd:string" minOccurs="0"/> <element name="description" type="xsd:string" minOccurs="0"/> <element name="encoding" type="docFileCabTyp:FileEncoding" minOccurs="0"/> <element name="altTagCaption" type="xsd:string" minOccurs="0"/> <element name="isOnline" type="xsd:boolean" minOccurs="0"/> <element name="isInactive" type="xsd:boolean" minOccurs="0"/> <element name="class" type="xsd:string" minOccurs="0"/> <element name="department" type="xsd:string" minOccurs="0"/> <element name="hideInBundle" type="xsd:boolean" minOccurs="0"/> <element name="isPrivate" type="xsd:boolean" minOccurs="0"/> <element name="owner" type="platformCore:RecordRef" minOccurs="0"/> <element name="caption" type="xsd:string" minOccurs="0"/> <element name="storeDisplayThumbnail" type="platformCore:RecordRef" minOccurs="0"/> <element name="siteDescription" type="xsd:string" minOccurs="0"/> <element name="isFeatured" type="xsd:boolean" minOccurs="0"/> <element name="featuredDescription" type="xsd:string" minOccurs="0"/> </pre>	<pre> <element name="url" type="xsd:string" minOccurs="0"/> <element name="mediaFile" type="platformCore:RecordRef" minOccurs="0"/> <element name="description" type="xsd:string" minOccurs="0"/> <element name="encoding" type="docFileCabTyp:FileEncoding" minOccurs="0"/> <element name="altTagCaption" type="xsd:string" minOccurs="0"/> <element name="isOnline" type="xsd:boolean" minOccurs="0"/> <element name="isInactive" type="xsd:boolean" minOccurs="0"/> <element name="class" type="xsd:string" minOccurs="0"/> <element name="department" type="xsd:string" minOccurs="0"/> <element name="hideInBundle" type="xsd:boolean" minOccurs="0"/> <element name="isPrivate" type="xsd:boolean" minOccurs="0"/> <element name="owner" type="platformCore:RecordRef" minOccurs="0"/> <element name="caption" type="xsd:string" minOccurs="0"/> <element name="storeDisplayThumbnail" type="platformCore:RecordRef" minOccurs="0"/> <element name="siteDescription" type="xsd:string" minOccurs="0"/> <element name="featuredDescription" type="xsd:string" minOccurs="0"/> </pre>

The diff file also includes data type changes. The following is a snapshot of **activities.scheduling.xsd**. This figure shows:

- The data types for the **estimatedTime**, **estimatedTimeOverride**, **actualTime**, and **timeRemaining** elements have changed from **dateTime** to **Duration**.

121 122 123 124 125 126 127 128	<pre><element name="sendEmail" type="xsd:boolean" minOccurs="0"/> <element name="closedEvent" type="xsd:boolean" minOccurs="0"/> <element name="estimatedTime" type="xsd:dateTime" minOccurs="0"/> <element name="estimatedTimeOverride" type="xsd:dateTime" minOccurs="0"/> <element name="actualTime" type="xsd:dateTime" minOccurs="0"/> <element name="timeRemaining" type="xsd:dateTime" minOccurs="0"/> <element name="percentTimeComplete" type="xsd:double" minOccurs="0"/> <element name="percentComplete" type="xsd:double" minOccurs="0"/></pre>	150 151 152 153 154 155 156 157	<pre><element name="sendEmail" type="xsd:boolean" minOccurs="0"/> <element name="closedEvent" type="xsd:boolean" minOccurs="0"/> <element name="estimatedTime" type="platform:core:Duration" minOccurs="0"/> <element name="estimatedTimeOverride" type="platform:core:Duration" minOccurs="0"/> <element name="actualTime" type="platform:core:Duration" minOccurs="0"/> <element name="timeRemaining" type="platform:core:Duration" minOccurs="0"/> <element name="percentTimeComplete" type="xsd:double" minOccurs="0"/> <element name="percentComplete" type="xsd:double" minOccurs="0"/></pre>
--	---	--	---

2013.2 Schema Browser

The 2013.2 version of the [Schema Browser](#) is now available. You can also access the 2013.2 Schema Browser by:

- In the NetSuite Help Center and in the SuiteTalk (Web Services) Records Guide, clicking the links to the Schema Browser that are provided in the description of each supported record.
- Opening the Help Center and navigating to SuiteCloud (Customization, Scripting, and Web Services) > SuiteTalk (Web Services) > SuiteTalk Schema Browser Version 2013.2. This topic includes a link to the Schema Browser.