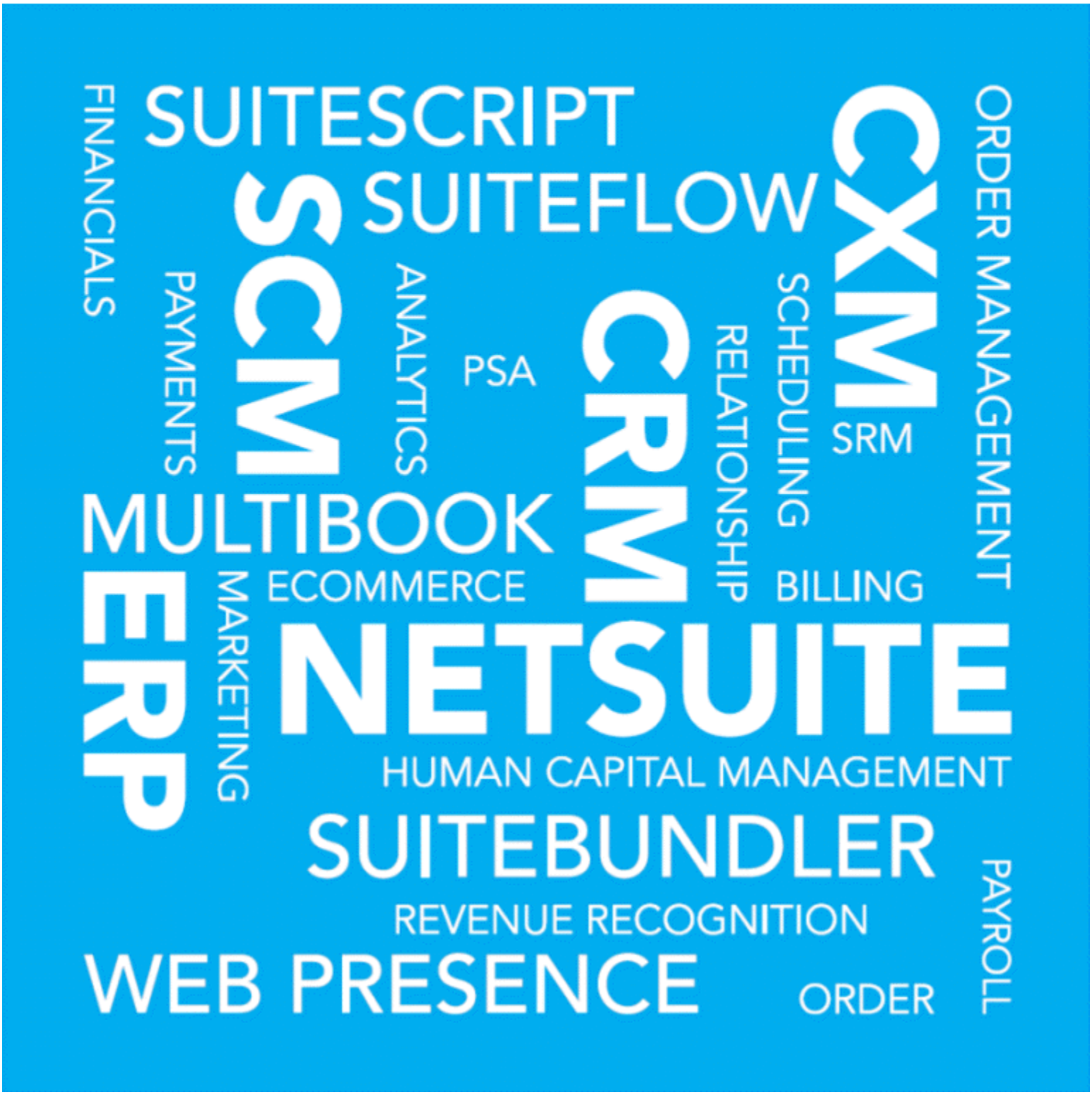
SuiteAnalytics Connect



and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described in this document remains at the sole discretion of Oracle.

This document in any form, software or printed matter, contains proprietary information that is the exclusive property of Oracle. Your access to and use of this confidential material is subject to the terms and conditions of your Oracle Master Agreement, Oracle License and Services Agreement, Oracle PartnerNetwork Agreement, Oracle distribution agreement, or other license agreement which has been executed by you and Oracle and with which you agree to comply. This document and information contained herein may not be disclosed, copied, reproduced, or distributed to anyone outside Oracle without prior written consent of Oracle. This document is not part of your license agreement nor can it be incorporated into any contractual agreement with Oracle or its subsidiaries or affiliates.

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Sample Code

Oracle may provide sample code in SuiteAnswers, the Help Center, User Guides, or elsewhere through help links. All such sample code is provided "as is" and "as available", for use only with an authorized NetSuite Service account, and is made available as a SuiteCloud Technology subject to the SuiteCloud Terms of Service at www.netsuite.com/tos.

Oracle may modify or remove sample code at any time without notice.

No Excessive Use of the Service

As the Service is a multi-tenant service offering on shared databases, Customer may not use the Service in excess of limits or thresholds that Oracle considers commercially reasonable for the Service. If Oracle reasonably concludes that a Customer's use is excessive and/or will cause immediate or ongoing performance issues for one or more of Oracle's other customers, Oracle may slow down or throttle Customer's excess use until such time that Customer's use stays within reasonable limits. If Customer's particular usage pattern requires a higher limit or threshold, then the Customer should procure a subscription to the Service that accommodates a higher limit and/or threshold that more effectively aligns with the Customer's actual usage pattern.

Beta Features

Oracle may make available to Customer certain features that are labeled "beta" that are not yet generally available. To use such features, Customer acknowledges and agrees that such beta features are subject to the terms and conditions accepted by Customer upon activation of the feature, or in the absence of such terms, subject to the limitations for the feature described in the User Guide and as follows: The beta feature is a prototype or beta version only and is not error or bug free and Customer agrees that it will use the beta feature carefully and will not use it in any way which might result in any loss, corruption or unauthorized access of or to its or any third party's property or information. Customer must promptly report to Oracle any defects, errors or other problems in beta features to support@netsuite.com or other designated contact for the specific beta feature. Oracle cannot guarantee the continued availability of such beta features and may substantially modify or cease providing such beta features without entitling Customer to any refund, credit, or other compensation. Oracle makes no representations or warranties regarding functionality or use of beta features and Oracle shall have no liability for any lost data, incomplete data, re-run time, inaccurate input, work delay, lost profits or adverse effect on the performance of the Service resulting from the use of beta features. Oracle's standard service levels, warranties and related commitments regarding the Service shall not apply to beta features and they may not be fully supported by Oracle's customer support. These limitations and exclusions shall apply until the date that Oracle at its sole option makes a beta feature generally available to its customers and partners as part of the Service without a "beta" label.

Table of Contents

SuiteAnalytics Connect

When enabled, the SuiteAnalytics Connect Service, also referred to as the Connect Service, lets you archive, analyze, and report on NetSuite data using a third-party tool or any custom-built application on any type of device using a Windows, Linux, or OS X operating system. If your company elects to enable the Connect Service, NetSuite offers ODBC, JDBC, and ADO.NET drivers that you can download, install, and use to connect to the Connect Service. You can use a variety of compatible applications, including Microsoft® Excel, BIRST, Adaptive, or any other compatible application, to work with the Connect Service.

Important: The SuiteAnalytics Connect Service provides a read-only method for obtaining NetSuite data. You cannot use the Connect Service to update NetSuite data.

This guide is best used in conjunction with the SuiteAnalytics Connect Browser. The browser lists all standard SuiteAnalytics Connect tables and columns, providing information about primary keys, foreign keys, and the related tables. In addition, some tables are joined to business domains that can be explored through domain diagrams. To learn more about the Connect Browser, see Working with the SuiteAnalytics Connect Browser.

Important: As of February 9, 2017, NetSuite no longer supports Windows ODBC drivers prior to version 7.20.54, or JDBC, ADO.NET, and Linux ODBC drivers prior to version 7.20.50. To continue using SuiteAnalytics Connect after this date, you must upgrade to the latest driver version for your operating system. After you have installed the latest driver version, you must also update your NetSuite connections to use certificate-based server authentication and to provide NetSuite Account ID and Role ID on every connection. As of February 9, 2017, any connections based on the new drivers that do not meet these requirements will not work. Additionally, the latest version of the SuiteAnalytics Connect JDBC driver requires at least Java SE 7. For more information about upgrading the SuiteAnalytics Connect driver, see the topic Upgrading an ODBC Driver. For information about setting up certificate-based server authentication and setting up connections to provide NetSuite Account ID and Role ID, see the following topics:

For information about setting up certificate-based server authentication, see the following topics:

- ODBC driver, see Authentication Using Server Certificates for ODBC.
- JDBC driver, see Authentication Using Server Certificates for JDBC.
- ADO.NET driver, see Authentication Using Server Certificates for ADO.NET.

For information about setting up connections to provide NetSuite Account ID and Role ID, see the following topics:

- ODBC driver, see Connection Attributes.
- JDBC driver, see JDBC Connection Properties.
- ADO.NET driver, see ADO.NET Connection Options.

To get started, review Prerequisites for Using the Connect Service, which includes:

- Determining which Type of Connect Driver to Use
- Firewall Restrictions for the Connect Service
- Removing Connect Drivers

After you have reviewed the prerequisites, see SuiteAnalytics Connect Setup, which includes:

- Enabling the Connect Service Feature

Prerequisites for Using the Connect Service

Before you will be able to use the Connect Service, ensure that you have met the prerequisites outlined in the related topics.

Removing Connect Drivers

When you would like to install a newer version of a Connect driver, you must remove any previously installed driver. The uninstall process may require a restart of your system.

For Windows operating systems, you may use the standard uninstall feature located in the Control Panel on your Windows system. You should also remove the existing DSN entries when you uninstall.

For Linux operating systems, delete the installation files.

Firewall Restrictions for the Connect Service

Ensure that port 1708 is not blocked for outgoing connections in the firewall.

SuiteAnalytics Connect Setup

No matter which Connect driver you plan to use for the Connect Service, this section details some key tasks you must complete before you move on to the specific Connect driver you would like to install. Before you start completing the tasks in this section, ensure that you have completed the Prerequisites for Using the Connect Service.

Enabling the Connect Service Feature

Whether you wish to use an ODBC, JDBC, or ADO.NET driver to access the Connect Service, you must first enable the SuiteAnalytics Connect feature.

Important: As of February 9, 2017, NetSuite no longer supports Windows ODBC drivers prior to version 7.20.54, or JDBC, ADO.NET, and Linux ODBC drivers prior to version 7.20.50. To continue using SuiteAnalytics Connect after this date, you must upgrade to the latest driver version for your operating system. After you have installed the latest driver version, you must also update your NetSuite connections to use certificate-based server authentication and to provide NetSuite Account ID and Role ID on every connection. As of February 9, 2017, any connections based on the new drivers that do not meet these requirements will not work. Additionally, the latest version of the SuiteAnalytics Connect JDBC driver requires at least Java SE 7. For more information about upgrading the SuiteAnalytics Connect driver, see the topic Upgrading an ODBC Driver. For information about setting up certificate-based server authentication and setting up connections to provide NetSuite Account ID and Role ID, see the following topics:

For information about setting up certificate-based server authentication, see the following topics:

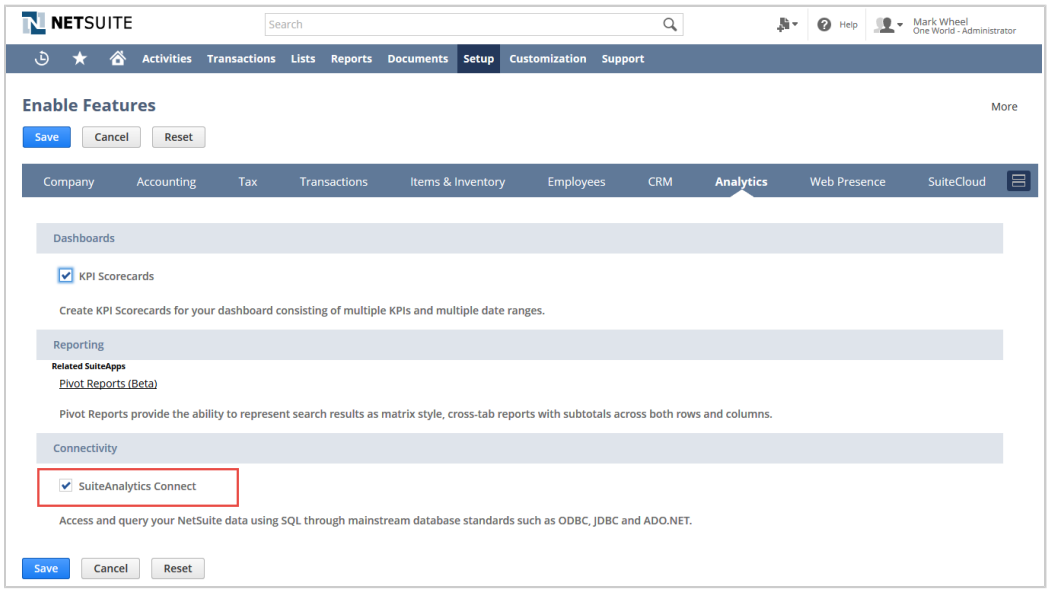
- ODBC driver, see Authentication Using Server Certificates for ODBC.
- JDBC driver, see Authentication Using Server Certificates for JDBC.
- ADO.NET driver, see Authentication Using Server Certificates for ADO.NET.

For information about setting up connections to provide NetSuite Account ID and Role ID, see the following topics:

- ODBC driver, see Connection Attributes.
- JDBC driver, see JDBC Connection Properties.
- ADO.NET driver, see ADO.NET Connection Options.

To enable the Connect Service feature:

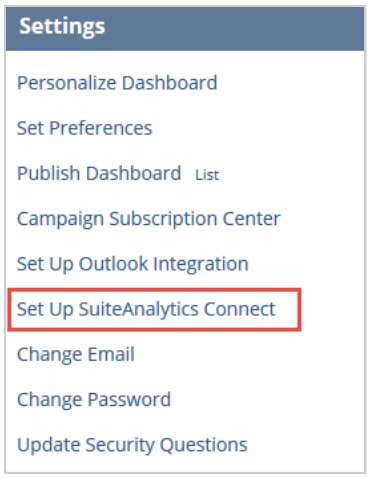
1. Ensure that your Account Administrator has enabled your Account and Role with the Connect Service feature.
2. Navigate to Setup > Company > Enable Features.
3. Click the **Analytics** tab.
4. Check the **SuiteAnalytics Connect** box.



Note: If you do not see this feature, it has not been provisioned for your account. Contact NetSuite Customer Support or your Account Manager for assistance.

Verifying the SuiteAnalytics Connect Permission

You need the SuiteAnalytics Connect permission to download a Connect driver and access the Connect Service. You can verify that you have the SuiteAnalytics Connect permission if the Set Up SuiteAnalytics Connect option appears in the Settings portlet on your home page.



Account administrators can assign the SuiteAnalytics Connect permission as a Setup permission for a role or as a global permission for an employee. The employee permission gives per-user control of Connect access. Account administrators should exercise caution when assigning this permission, however, as some NetSuite permissions and restrictions are not enforced for Connect Service. The same permissions apply for accessing the Connect Service no matter the type of driver used. To learn how to assign the SuiteAnalytics Connect permission to roles and employees, see Providing Users with SuiteAnalytics Connect Permissions.

Important: Certain records in the Connect schema are only obtainable using an Administrator role, despite setting the appropriate permissions in NetSuite. Consequently, users assigned to custom roles with access to the Connect Service may see a discrepancy between the information displayed in NetSuite and the information pulled when running a query using SuiteAnalytics Connect.

Enforced Permissions

The following permissions are enforced for Connect access:

- All Transactions permissions and Lists permissions for employees
- Customers
- Partners
- Vendors
- Accounting registers

Enforcement of these permissions means that users have the same level of access to employee, customer, partner, vendor, types of transaction, and types of register records when they access NetSuite data through the Connect Service as they do in the NetSuite user interface.

Non-enforced Permissions

Other permissions are not enforced for Connect access, including:

- Classes
- Departments
- Locations
- Custom records
- Subsidiary Restrictions (OneWorld only)

This lack of enforcement means that users with the Connect permission enabled can access records of these types through the Connect Service that they cannot access in the NetSuite user interface.

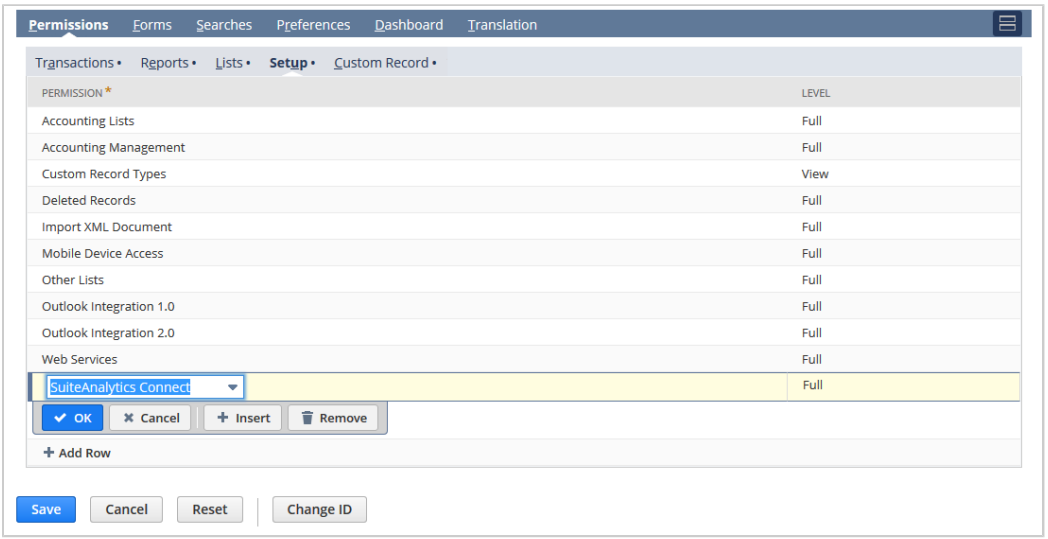
Providing Users with SuiteAnalytics Connect Permissions

As an administrator, you can assign other users with the SuiteAnalytics Connect permission. You can enable permissions for users through the Manage Roles option or through the Employee record.

Important: Certain records in the Connect schema are only obtainable using an Administrator role, despite setting the appropriate permissions in NetSuite. Consequently, users assigned to custom roles with access to the Connect Service may see a discrepancy between the information displayed in NetSuite and the information pulled when running a query using SuiteAnalytics Connect.

To set up SuiteAnalytics Connect permission using Manage Roles:

1. Navigate to Setup > Users/Roles > User Management > Manage Roles.
2. Click **Customize** next to the name of the role for which you would like to add the SuiteAnalytics Connect permission.
3. Click the **Setup** tab under the Permissions tab.
4. Add the SuiteAnalytics Connect permission.



Important: You cannot add the SuiteAnalytics Connect permission to a role that has SAML Single Sign-on permission.

5. Click **Add**.
6. Click **Save**.

To set up SuiteAnalytics Connect permission using the Employee record:

1. Navigate to Lists > Employees > Employees.
2. Click **Edit** next to the name of employee.
3. Click the **Access** tab.
4. Select the role you would like to grant SuiteAnalytics Connect permission, and click the Open icon next to it.

5. Click the **Setup** tab under the Permissions tab.
6. Add the SuiteAnalytics Connect permission.
7. Click **Add**.
8. Click **Save**.

SuiteAnalytics Connect Authentication

SuiteAnalytics Connect has the following authentication considerations for security purposes.

- After six failed attempts, the account is locked. You cannot access your account for thirty minutes, at which point you will have one additional attempt. If another failed attempt occurs, the account is locked again for thirty minutes.
- You cannot access SuiteAnalytics Connect when the Require Password Change on Next Login option is selected on your Employee record's Access tab.
- You cannot access SuiteAnalytics Connect if your Employee record is inactive.
- You cannot access SuiteAnalytics Connect when your role with the SuiteAnalytics Connect permission is inactive.

Supported Windows Versions

The following Windows operating systems are compatible with the SuiteAnalytics Connect service:

- Windows 7
- Windows 8
- Windows 10
- Windows Server 2008 R2
- Windows Server 2012

Note: For more information, see Determining which Type of Connect Driver to Use.

Supported Linux Distributions

The following Linux distributions are compatible with the SuiteAnalytics Connect service:

- CentOS 5.1
- CentOS 6.4
- OpenSUSE 12.3
- OpenSUSE 13.1
- Ubuntu 12.04

Note: For more information, see Determining which Type of Connect Driver to Use.

Determining which Type of Connect Driver to Use

You should select the driver based on your operating system and the application you want to integrate with NetSuite.

The following table lists the supported drivers and platforms.

Driver Type	Operating System (32-bit/64-bit)	Application (32-bit/64-bit)	Driver Type to Use
ODBC	Windows (32-bit)	32-bit	ODBC Driver 32-bit For more information, see Accessing the Connect Service Using an ODBC Driver.
ODBC	Windows (64-bit)	32-bit	ODBC Driver 32-bit For more information, see Accessing the Connect Service Using an ODBC Driver.
ODBC	Windows (64-bit)	64-bit	ODBC Driver 64-bit For more information, see Accessing the Connect Service Using an ODBC Driver.
ODBC	Linux (32-bit)	32-bit	ODBC Driver 32-bit For more information, see Accessing the Connect Service Using an ODBC Driver.
ODBC	Linux (64-bit)	32-bit	ODBC Driver 32-bit For more information, see Accessing the Connect Service Using an ODBC Driver.
ODBC	Linux (64-bit)	64-bit	ODBC Driver 64-bit For more information, see Accessing the Connect Service Using an ODBC Driver.
JDBC	Windows/Linux/OS X (32-bit/64-bit)	any	JDBC Driver For more information, see Accessing the Connect Service Using a JDBC Driver.
ADO.NET	Windows (32-bit/64-bit)	any	ADO.NET Driver For more information, see Accessing the Connect Service Using an ADO.NET Data Provider.

Downloading and Installing Connect Drivers

If you have the SuiteAnalytics Connect permission enabled, you can download and install an ODBC, JDBC, or ADO.NET driver. To ensure that you have this permission, see Verifying the SuiteAnalytics Connect Permission.

For more information on downloading and installing a specific Connect driver, see the following topics.

Reviewing Configuration Information

Click the Your Configuration link on the Drivers download page to view the configuration details you need to install a Connect driver. To connect to the NetSuite account, environment, and role you are currently logged under, your connection configuration must follow the guidelines in the following table.

Parameters	Value
Service Host	Choose the value appropriate for your data center location and account type (production, sandbox, or Release Preview). For more information, see the help topic Understanding NetSuite URLs and Data Centers.
Service Port	1708
Service Data Source	NetSuite.com
Account ID	Your Account ID
Role ID	Your Role ID

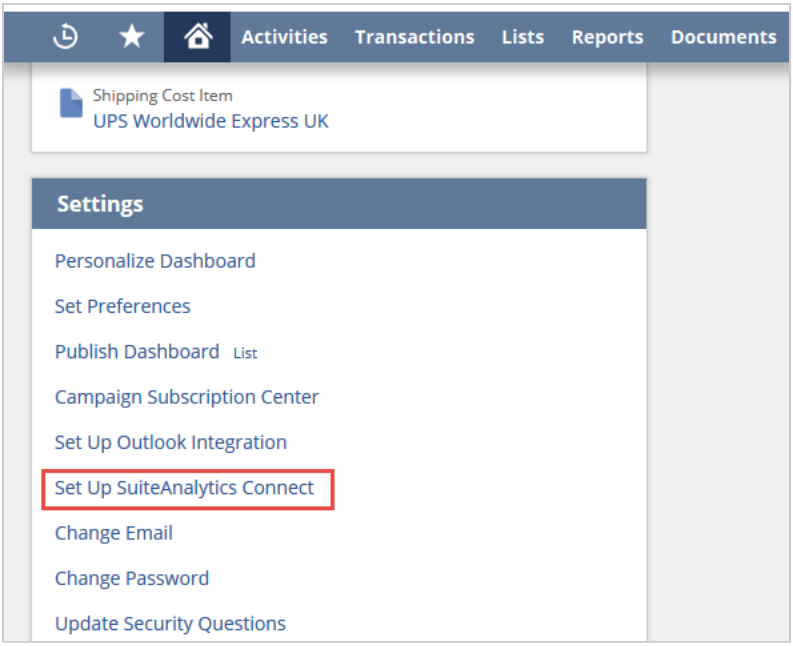
The values of the Service Host configuration parameter determines the environment of your NetSuite account. The environment can be production, sandbox, or Release Preview. You must log in to NetSuite and go to your drivers download page to determine your Service Host. The Service Host you will use depends on the location of the data center where your account resides.

Downloading a Connect Driver

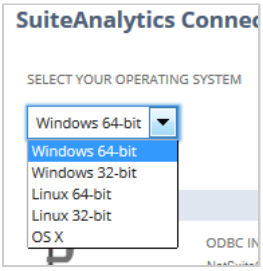
Before installing a SuiteAnalytics Connect driver, make sure that the Connect Service feature is enabled. For more information, see Enabling the Connect Service Feature.

To download a Connect driver:

1. On your NetSuite home page, find the **Settings** portlet and click **Set Up SuiteAnalytics Connect**. If the link is not displayed, you need to enable the Connect Service first. For more information, see Enabling the Connect Service Feature.



2. In the upper left corner, select your platform.

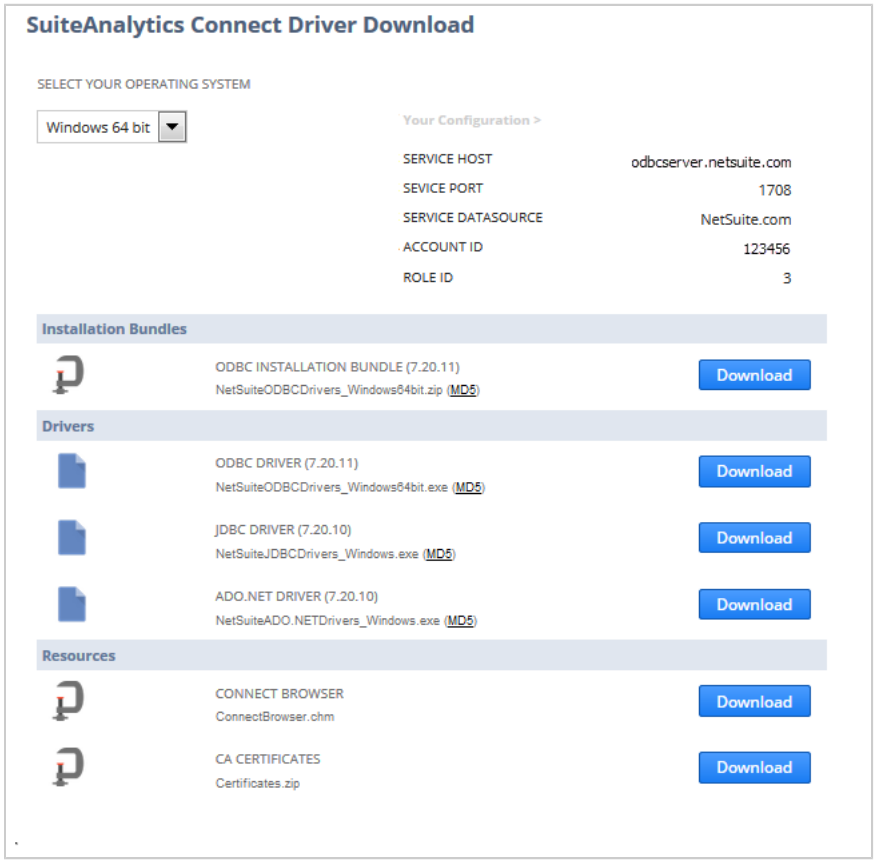


> **Note:** If your platform is 64-bit, but you are planning to use a 32-bit application to get data from NetSuite, choose the 32-bit version of your platform, so you can install a 32-bit SuiteAnalytics Connect driver.

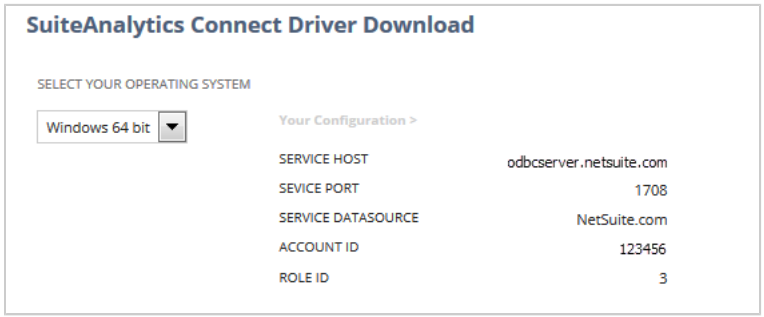
3. Click the **Download** button next to the driver type you want to install. You can choose among the following:

- ODBC driver
- JDBC driver
- ADO.NET driver

If you are not sure which driver to install, see Determining which Type of Connect Driver to Use.



4. The top part of the download page includes an expandable section called **Your Configuration**. This section includes the values of **Service Host**, **Service Port**, **Service Datasource**, **Account ID**, and **Role ID** for your current company and role. Make a note of these values: you will need them when configuring the drivers to connect to the SuiteAnalytics Connect service.



5. In addition to the SuiteAnalytics Connect driver, you can also download the Connect Browser and security certificates (CA Certificates).

For further instructions on how to install and configure the downloaded driver, see:

- Downloading and Installing the ODBC Driver for Windows

Determining Your Connect Driver Version

The way to determine your SuiteAnalytics Connect driver version depends on your platform.

- **On Windows** – Open the Windows Control Panel and browse to Programs > Programs and Features. Find your SuiteAnalytics Connect driver in the list and check its version in the Version column.
- **On Linux** – Currently, there is no simple way to determine your SuiteAnalytics Connect driver version on Linux. As an option, you can check whether your driver installation directory was created after January 7, 2016. If the driver installation directory was created earlier than January 7, 2016, most likely, you are not using the most recent version of the driver.
- **On OS X** – Currently, there is no simple way to determine your SuiteAnalytics Connect driver version on a Mac. As an option, you can check whether your driver installation folder was created after January 7, 2016. If the driver installation folder was created earlier than January 7, 2016, most likely, you are not using the most recent version of the driver.

Accessing the Connect Service Using an ODBC Driver

The following table lists tasks for an ODBC driver to connect to Connect Service; however, some of these tasks are optional to complete.

Note: If your account moves to a different data center, download, and reinstall the appropriate ODBC driver to automatically configure the host to your account's new data center. For more information that ensures your account is data center agnostic and ready for a move if necessary, see the help topic Understanding Multiple Data Centers.

Task	Description
Set up an environment and verify installation prerequisites.	For both Windows and Linux, see Prerequisites to verify the installation prerequisites.
Download and install the driver.	For Windows, see Downloading and Installing the ODBC Driver for Windows. For Linux, see Downloading and Installing the ODBC Driver for Linux.
Set up the ODBC connection.	For Windows, see Configuring the ODBC Data Source on Windows. For Linux, see Configuring the ODBC Data Source on Linux.
Optionally, upgrade from a previous version.	For both Windows and Linux, see Upgrading an ODBC Driver to upgrade a previous installation.
Optionally, enable authentication with server certificates.	For both Windows and Linux, see Authentication Using Server Certificates for ODBC to add increased encryption to secure the data connection.

Prerequisites

Before you begin the download and installation process, complete all prerequisites. For more information, see the Prerequisites for Using the Connect Service.

Downloading and Installing the ODBC Driver for Windows

For Windows, two types of installation methods exist.

- **Driver Installation Bundle.** The installation bundle is a zip file that contains the installer together with a dsn.ini file. The dsn.ini file is automatically generated and contains parameters which are used by the installer to create the DSN. The dsn.ini file contains parameters which are reflecting the user context, for example, the role and the account that you used at the time you downloaded the zip file. The user context data includes the Service Host, Account ID, and Role ID. Use this method if you want to use the driver to connect to your current account. See ODBC Installation on Windows - Bundled Installation.
- **Installer only.** The installer does not include any user context data. You should use this method if you want to install the driver for other users. Installers are available only for the Windows operating

system. The .exe file installs the driver and unpacks other distributable content (for example, the license and certificates) to the folder selected during the installation. For an ODBC driver, the installer also creates the DSN in the operating system unless you choose to skip this step. For more information, see ODBC Installation on Windows - Installer Only.

ODBC Installation on Windows - Bundled Installation

Download an installation bundle to automatically incorporate your current user context data (such as Service Host, Account ID and Role ID) in to the installer. This is the recommended approach if you want to use the driver to connect to your current account. This installer uses the provided `dsn.ini` file to configure the System DSNs using the account information for the user who downloaded the installation bundle. Use this option if you want to install the driver on a single machine and connect to a production account.

When you install the bundle you get DSN content from dsn.ini and thus the DSN is created automatically.

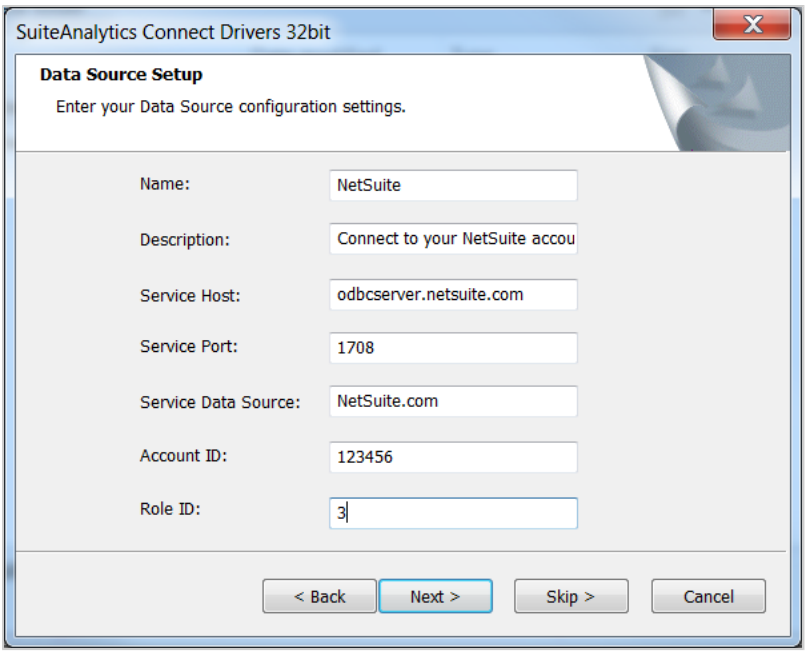
After you complete the installation, you can add additional System DSNs to connect to the Connect Service for a sandbox or Release Preview account. For more information, see Configuring the ODBC Data Source on Windows and Driver Access for a Sandbox or Release Preview Account.

Important: If you have the NetSuite ODBC 6.0 driver installed, please remove it before installing the new driver version. To learn how to determine your driver version, see Determining Your Connect Driver Version.

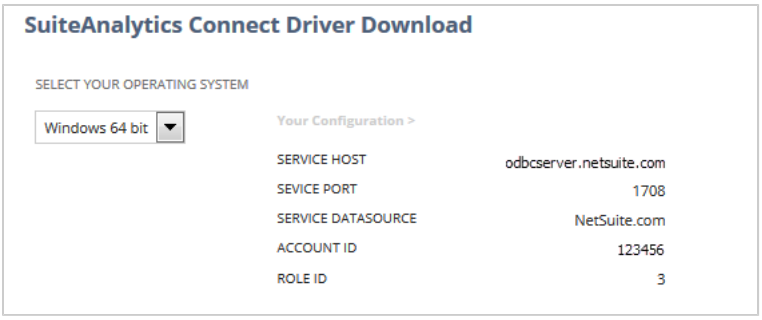
To install or update the ODBC driver using a bundled installation:

1. In the **Settings** portlet on your NetSuite home page, click **Set Up SuiteAnalytics Connect**. If the download link is not displayed, you need to enable the Connect Service. For more information, see Enabling the Connect Service Feature.
2. In the upper left corner, select your platform.

   Note: If your platform is 64-bit, but you are planning to use a 32-bit application to get data from NetSuite, choose the 32-bit version of your platform, so you can install a 32-bit SuiteAnalytics Connect driver.
3. Click the **Download** button next to the ODBC Installation Bundle.
4. When the driver installation bundle is downloaded, extract the .zip file. The dsn.ini file and the driver .exe file are extracted to the same folder. If you decide to move the files elsewhere, make sure you move both the dsn.ini file and the .exe file, as they should always be in the same location.
5. Run **NetSuiteODBCDrivers_Windows32bit.exe** or **NetSuiteODBCDrivers_Windows64bit.exe**, depending on the driver version you downloaded.
6. Follow the wizard's instructions to install or update the driver.
7. If you performed a new installation of the driver, review the provided values in the **Data Source Setup** dialog and modify them, if needed.



You can find the values for this dialog under **Your Configuration** on the drivers download page.



8. When the installation is complete, check that the correct data source is configured for your ODBC driver. For more information, see Verifying the ODBC Driver Installation on Windows.

When you installed the driver and verified the data source settings, you can configure your applications to access the SuiteAnalytics Connect service. One of the options is to connect your application using a connection string. To learn more about connection strings, see Connecting Using a Connection String.

ODBC Installation on Windows - Installer Only

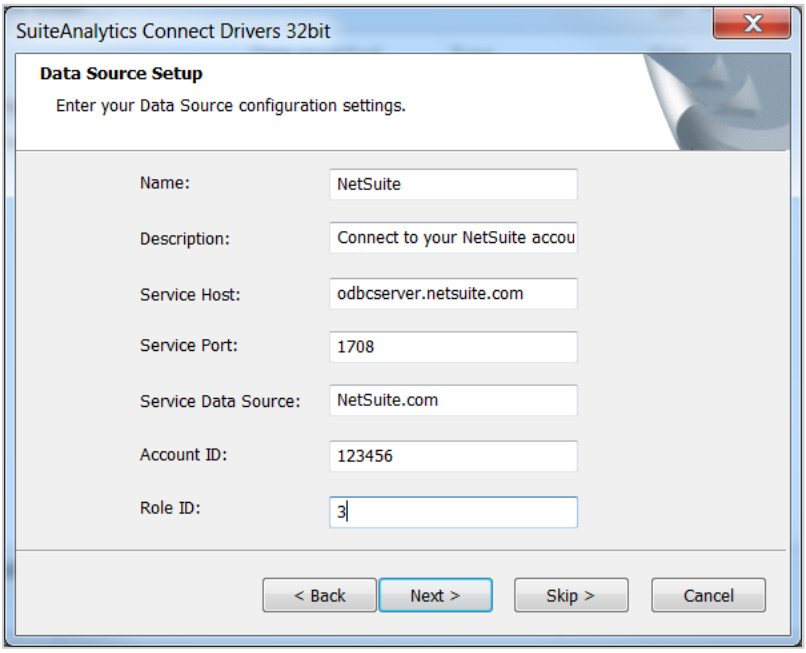
Download an installer if you do not want your current user context data to be used. This is the recommended approach if you want to install the driver or automated services. This installer can create generic System DSNs. If you use only the installer file, and there is no dsn.ini file, then fallback content is provided during installation which configures the default service host without any role/account IDs. You must review the content and add the Account ID and Role ID to get a DSN or you can continue by skipping the step which creates the DSN and complete the installation process. You may need to create additional DSNs manually, if you want to connect to another server or you want to select between different types of environment, for example, both production and sandbox. For more information, see Configuring the ODBC Data Source on Windows.

Important: If you have the NetSuite ODBC 6.0 driver installed, please remove it before installing the new driver version. To learn how to determine your driver version, see Determining Your Connect Driver Version

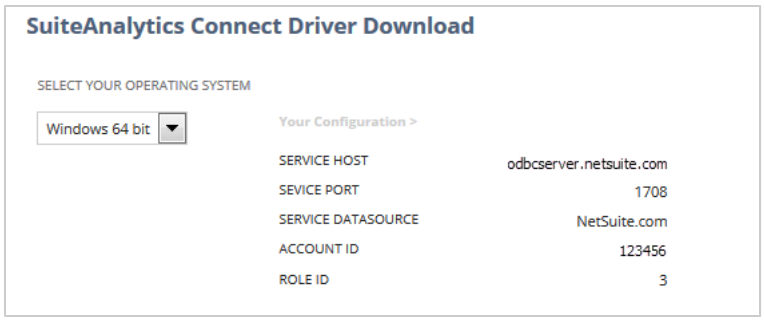
To install or update the ODBC driver using an installer:

1. In the Settings portlet on your NetSuite home page, click **Set Up SuiteAnalytics Connect**. If the download link is not displayed, you need to enable the Connect Service. For more information, see Enabling the Connect Service Feature.
2. In the upper left corner, select your platform.

   Note: If your platform is 64-bit, but you are planning to use a 32-bit application to get data from NetSuite, choose the 32-bit version of your platform, so you can install a 32-bit SuiteAnalytics Connect driver.
3. Click the **Download** button next to the ODBC Driver.
4. When the driver installation file is downloaded to your computer, double-click the file to run the driver installer.
5. Follow the wizard's instructions to install or update the driver.
6. If you performed a new installation of the driver, review the provided values and insert the **Account ID** and **Role ID** for the target user.



You can find the values for this dialog under **Your Configuration** on the drivers download page.



7. When the installation is complete, verify that the data source is configured correctly for your driver. For more information, see Verifying the ODBC Driver Installation on Windows.

When you installed the driver and verified the data source settings, you can configure your applications to access the SuiteAnalytics Connect service. One of the options is to connect your application using a connection string. To learn more about connection strings, see Connecting Using a Connection String.

Verifying the ODBC Driver Installation on Windows

You can verify that the Connect for ODBC driver is installed by running the ODBC Administrator tool. The location of the ODBC Administrator tool depends on your version of Windows.

- 32-bit Windows versions:

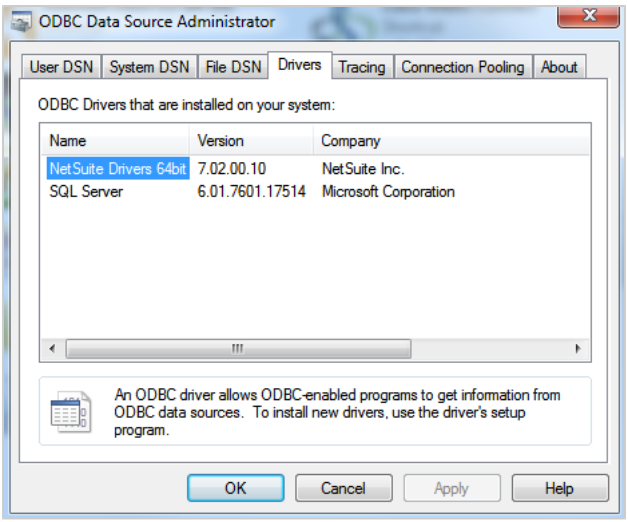
Connect Driver Version	ODBC Administrator Location
32-bit driver	`C:\Windows\System32\odbcad32.exe`

- 64-bit Windows versions:

Connect Driver Version	ODBC Administrator Location
32-bit driver	`C:\Windows\SysWOW64\odbcad32.exe`
64-bit driver	`C:\Windows\System32\odbcad32.exe`

To verify the ODBC driver version on Windows:

1. Open the ODBC Administrator tool.
2. Click the **Drivers** tab.



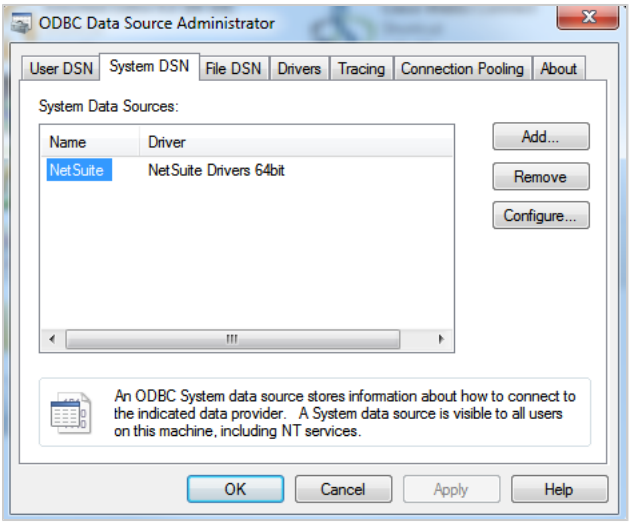
3. In the list of drivers, try to find a driver whose name contains **NetSuite**. This indicates that you have an official NetSuite ODBC driver installed. If no such driver is present, then you either have a non-official ODBC driver or the installation was not successful.
4. After you have located the driver, check the version. For the most up-to-date information, see NetSuite's Connect driver download page.

Configuring the ODBC Data Source on Windows

To set up Connect Service correctly, you must connect to the correct data center. Adding or modifying the data center information can be done in Windows in the ODBC Administrator tool. To learn the tool's location in your system, see Verifying the ODBC Driver Installation on Windows.

To configure the ODBC data source on Windows:

1. Open the ODBC Administrator tool.
2. Click the **System DSN** tab.



3. From the list of **System Data Sources**, select the NetSuite data source name and click **Configure**.

If the **System Data Sources** list doesn't contain the NetSuite data source, click the **Add** button, select the NetSuite driver in the drivers list, and click **Finish**.

4. On the **General** tab, set the configuration fields. You can find these values under **Your Configuration** on the SuiteAnalytics Connect Driver Download page.

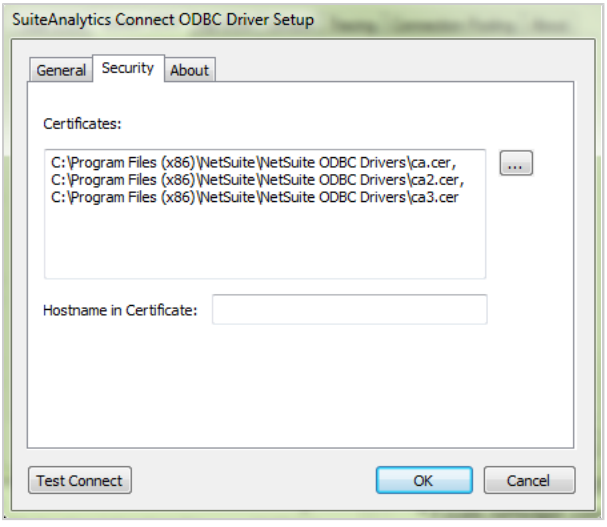
   The following table describes the configuration fields:

Data Source Setup Fields	Description
Name	Enter a data source name (DSN) to help you identify the connection, for example, **NetSuite**.
Description	Enter a description.
Service Host	The Connect Service host name. The host name you should use for your connection is displayed in the **Service Host** field on the SuiteAnalytics Connect Driver Download page, under **Your Configuration**.
Service Port	1708
Service Data Source	NetSuite.com
Account ID	The NetSuite account ID that will access the SuiteAnalytics Connect schema.
Role ID	Role ID corresponding to the Account ID.

5. On the **Security** tab, make sure that the **Certificates** field contains the following values, depending on your driver version. You can use the Browse button ... to locate the certificate files.

   - 32-bit driver: C:\Program Files (x86)\NetSuite\NetSuite ODBC Drivers\ca.cer, C:\Program Files (x86)\NetSuite\NetSuite ODBC Drivers\ca2.cer, C:\Program Files (x86)\NetSuite\NetSuite ODBC Drivers\ca3.cer
   - 64-bit driver: C:\Program Files\NetSuite\NetSuite ODBC Drivers\ca.cer, C:\Program Files \NetSuite\NetSuite ODBC Drivers\ca2.cer, C:\Program Files\NetSuite\NetSuite ODBC Drivers \ca3.cer

   We recommend that you leave the **Hostname in Certificate** field blank.



6. Click **Test Connect** to verify the connection. Use your NetSuite user name and password to connect.

7. Click **OK**.

When you try to connect your external tools to NetSuite using the SuiteAnalytics Connect ODBC driver, in addition to the settings you configured in the ODBC Administrator tool, you may need to provide a user ID (or user name) and a password for your connection. You can use your NetSuite account email address and password as the user ID and password for your connection.

Important: Tracing is an optional feature available in the ODBC Administrator tool that creates a log of all the calls to ODBC drivers. If you enable tracing, you might experience significantly slower performance when accessing the Connect Service using the ODBC driver.

Downloading and Installing the ODBC Driver for Linux

You can install an ODBC driver in a Linux operating system.

- Before you install the driver, make sure your Linux distribution is compatible with the ODBC driver for Linux. For details, see Supported Linux Distributions.
- To learn how to install the ODBC driver for Linux, see Installing the Latest Driver on Linux.
- To learn how to install the driver in a custom location, see Installing a Driver for Linux in to an Alternate Directory.

Installing the Latest Driver on Linux

To install the latest driver on Linux, follow these steps.

To install the latest driver on Linux:

1. In the **Settings** portlet on your NetSuite home page, click **Set Up SuiteAnalytics Connect**. If the download link is not displayed, you need to enable the Connect Service. For more information, see Enabling the Connect Service Feature.
2. In the upper left corner, select your platform.
3. Click the **Download** button next to the ODBC Driver.
4. Agree to the terms of service to be able to start the download. You cannot download the driver archive unless you agree to the terms of service.
5. Save the installation zip file to your computer.
6. In the location where you downloaded the zip file, extract the zip file.
7. Create a new installation directory with the following path:

```
/opt/netsuite/odbcclient
```

Note: If you wish to install the driver to another location, see Installing a Driver for Linux in to an Alternate Directory for more information.

8. Copy the extracted installation files to the installation directory.
9. Set up the DSN entries. For information, see Configuring the ODBC Data Source on Linux.

10. Verify the ODBC driver installation. See Verifying the ODBC Driver Installation on Linux.

Installing a Driver for Linux in to an Alternate Directory

To install a driver for Linux in to an alternate directory, update the oaodbc[64].sh, oaodbc[64].csh, and oaodbc[64].ini files with the installation directory of your choice.

To install a driver for Linux in to an alternate directory:

1. Extract the zip archive content in to a directory of your choice.
2. Update the oaodbc[64].sh, oaodbc[64].csh, and oaodbc[64].ini files with the installation directory of your choice.

   For example if we want to install in to /drivers/netsuite-odbc we replace default path /opt/netsuite/odbcclient in oaodbc64.sh so the result looks like:

   ```
   LD_LIBRARY_PATH=/drivers/netsuite-odbc/lib64${LD_LIBRARY_PATH:+":"}${LD_LIBRARY_PATH:-""}
   export LD_LIBRARY_PATH
   OASDK_ODBC_HOME=/drivers/netsuite-odbc/lib64; export OASDK_ODBC_HOME
   ODBCINI=/drivers/netsuite-odbc/odbc64.ini; export ODBCINI
   ```

   The same has to be done for all oaodbc[64].sh, oaodbc[64].csh, and oaodbc[64].ini files.

Accessing ODBC Data Source on Linux

In Linux, the ODBC Driver Manager accesses defined Data Sources. For the Driver Manager to access the defined drivers, you must set the environment correctly. The following system properties need to be set so the Driver Manager is able to locate and utilize the driver:

```
LD_LIBRARY_PATH - path to ODBC driver libraries

OASDK_ODBC_HOME - path to ODBC driver libraries

ODBCINI - path to ini file holding Data Source definitions
```

You must set correct values for these system variables. You can do so by running:

```
source oaodbc[64].sh
```

on your command line. This command will export the variables with correct values and it will make them available in your current process. Exporting the variables makes them accessible for processes that are started from current shell. Executing this command (exporting the variables) must precede starting an ODBC client that uses the SuiteAnalytics Connect Linux ODBC driver and must be executed within the same environment.

The default values are:

```
LD_LIBRARY_PATH=/opt/netsuite/odbcclient/lib64${LD_LIBRARY_PATH:+":"}${LD_LIBRARY_PATH:-""}

OASDK_ODBC_HOME=/opt/netsuite/odbcclient/lib64;

ODBCINI=/opt/netsuite/odbcclient/odbc64.ini;
```

Note: These values change if you do not install in the default directory. You are prompted to replace these values in oaodbc[64].sh if you are installing in to an alternate directory.

Verifying the ODBC Driver Installation on Linux

You can verify your ODBC driver installation with the `isql` command on Linux.

Note: isql may not be installed on your system by default. You may need to install additional package that contains this application.

To verify the ODBC driver installation on Linux:

1. Go to the installation directory.
2. Execute the `source oaodbc[64].sh` or `source oaodbc[64].csh` shell command.
3. Run the following command from a command prompt:

```
isql NetSuite <user name>@netsuite.com <NetSuite user account password>
```

NetSuite refers to the data source defined in odbc.ini.

Note: You can also use the `–v` switch to run the command in verbose mode to get more information in the event of a connection failure.

Configuring the ODBC Data Source on Linux

To update the ODBC Data Source on Linux, you must edit the `odbc.ini` or `odbc64.ini` file, depending on whether you installed the 32-bit or the 64-bit version of the driver. Use the `odbc64.ini` if you installed a 64-bit ODBC Connect driver.

To edit odbc.ini or odbc64.ini and configure the ODBC data source on Linux:

1. Locate the `odbc.ini` or `odbc64.ini` file. Typically, this file is located in your driver's installation folder.
2. Edit the following values in `odbc.ini` or `odbc64.ini`. You can find many of these values under **Your Configuration** on the SuiteAnalytics Connect Driver Download page.

Variable Name	Description
`Host`	Connect Service host name. The host name you should use for your connection is displayed in the **Service Host** field on the SuiteAnalytics Connect Driver Download page, under **Your Configuration**.
`Port`	Should always be set to `1708`.
`ServerDataSource`	Data source for the Connect Service. This value is `NetSuite.com`.
`Encrypted`	Should always be set to `1`.
`TrustStore`	The security certificates used for the connection.
`AccountID`	The NetSuite account ID that will access the SuiteAnalytics Connect schema. Enter this value under `CustomProperties`.

Variable Name	Description
RoleID	Role ID corresponding to the Account ID. Enter this value under CustomProperties.

To add an ODBC Data Source on Linux

You can have more than one Data Source defined in a Linux ODBC driver configuration.

Note: The below examples are working with a 64-bit version, for a 32–bit version use odbc.ini.

To add a new ODBC Data Source on Linux, you must edit the odbc64.ini file:

1. Locate odbc[64].ini.
2. Modify [ODBC Data Sources] section. Add new line:

```
MyNewDatasource=NetSuite ODBC Drivers 7.2
```

3. Add a new [MyNewDatasource] section:

```
[MyNewDatasource]
Driver=/opt/netsuite/odbcclient/lib64/ivoa25.so
Description=My new Sandbox ODBC datasource
Host=odbcserver.sandbox.netsuite.com
Port=1708
ServerDataSource=NetSuite.com
Encrypted=1
TrustStore=/opt/netsuite/odbcclient/cert/ca.cer,/opt/netsuite/odbcclient/cert/ca2.cer,/opt/nets
uite/odbcclient/cert/ca3.cer
CustomProperties=AccountID=123456;RoleID=3
```

The resulting odbc64.ini file looks like the following:

```
[ODBC Data Sources]
NetSuite_DC001=NetSuite ODBC Drivers 7.2
MyNewDatasource=NetSuite ODBC Drivers 7.2

[NetSuite]
Driver=/opt/netsuite/odbcclient/lib64/ivoa25.so
; enter any description
Description=Connect to your NetSuite account
; enter the <Service Host>
Host=odbcserver.netsuite.com
Port=1708
ServerDataSource=NetSuite.com
Encrypted=1
TrustStore=/opt/netsuite/odbcclient/cert/ca.cer,/opt/netsuite/odbcclient/cert/ca2.cer,/opt/nets
uite/odbcclient/cert/ca3.cer
; enter the <Account ID> and <Role ID> reflecting the format below
CustomProperties=AccountID=123456;RoleID=3

[MyNewDatasource]
Driver=/opt/netsuite/odbcclient/lib64/ivoa25.so
Description=My new Sandbox ODBC datasource
Host=odbcserver.sb.netsuite.com
```

```
Port=1708
ServerDataSource=NetSuite.com
Encrypted=1
TrustStore=/opt/netsuite/odbcclient/cert/ca.cer,/opt/netsuite/odbcclient/cert/ca2.cer,/opt/nets
uite/odbcclient/cert/ca3.cer
CustomProperties=AccountID=123456;RoleID=3

[ODBC]
Trace=0
IANAAppCodePage=4
TraceFile=odbctrace.out
TraceDll=/opt/netsuite/odbcclient/lib64/ddtrc25.so
InstallDir=/opt/netsuite/odbcclient
```

Connecting Using a Connection String

If you want to use a connection string to connect to NetSuite, or if your application requires it, you must specify either a DSN (data source name) or a DSN-less connection in the string.

- For a DSN connection, use the **DSN=** attribute, along with other required attributes.
- For a DSN-less connection, use the **DRIVER=** attribute, along with other required attributes.

A DSN connection string tells the driver where to find the default connection information. Optionally, you may specify **attribute=value** pairs in the connection string to override the default values stored in the data source.

DSN connection

The DSN connection string has the form:

```
DSN=data_source_name[;attribute=value[;attribute=value]...]
```

Note: The data source name (DSN) depends on the driver configuration. When you install the SuiteAnalytics Connect driver, the data source name is set to **NetSuite**.

For example, a connection string may look like the following. Only the data source name (DSN), user name (UID), and password (PWD) are required:

```
DSN=NetSuite;UID=test@netsuite.com;PWD=test123456
```

To learn your data source name, check your SuiteAnalytics Connect driver settings. For details, see Configuring the ODBC Data Source on Windows.

DSN-less connection

The DSN-less connection string specifies a driver instead of a data source name. All connection information must be entered in the connection string because there is no data source storing the information.

The DSN-less connection string has the form:

```
DRIVER=driver_name[;attribute=value[;attribute=value]...]
```

Note: Empty string is the default value for attributes that use a string value unless otherwise noted.

A DSN-less connection string must provide all necessary connection information:

```
DRIVER=NetSuite Drivers 32bit;Host=odbcserver.netsuite.com;Port=1708;Encrypted=1;Truststore=C:\
\Program Files (x86)\\NetSuite\\NetSuite ODBC Drivers\\ca.cer,C:\\Program Files (x86)\\NetSuite
\\NetSuite ODBC Drivers\\ca2.cer,C:\\Program Files (x86)\\NetSuite\\NetSuite ODBC Drivers\\ca3.
cer;SDSN=NetSuite.com;UID=test@netsuite.com;PWD=test123456;CustomProperties=AccountID=123456;Ro
leID=3
```

Note: Driver names may vary, depending on whether you use a 32-bit or the 64-bit version of the driver.

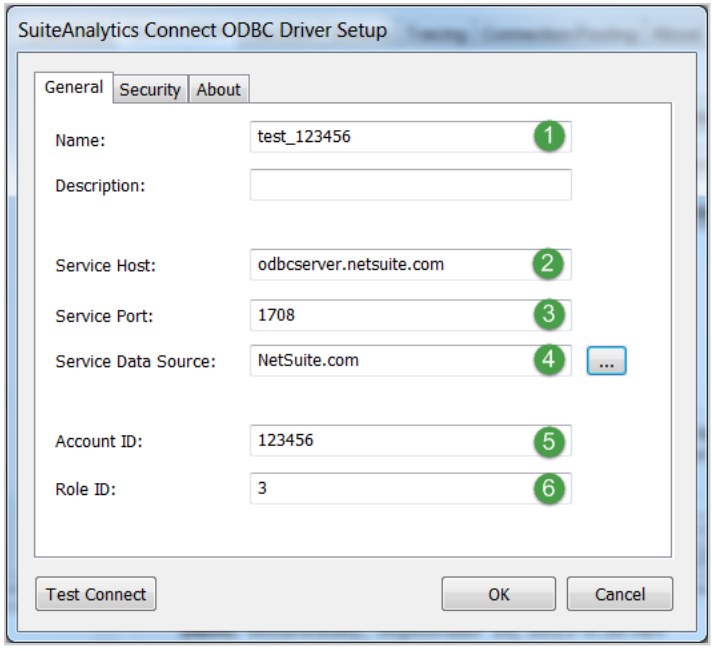
Connection Attributes

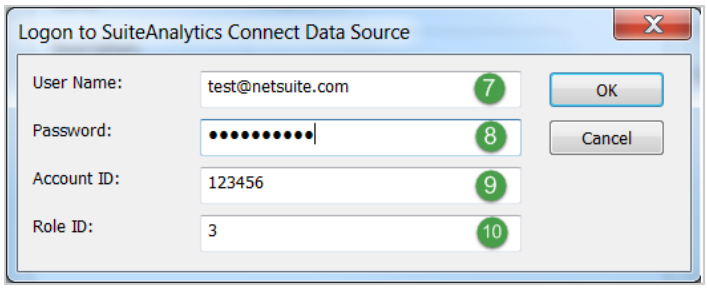
You can modify a connection by specifying connection string options.

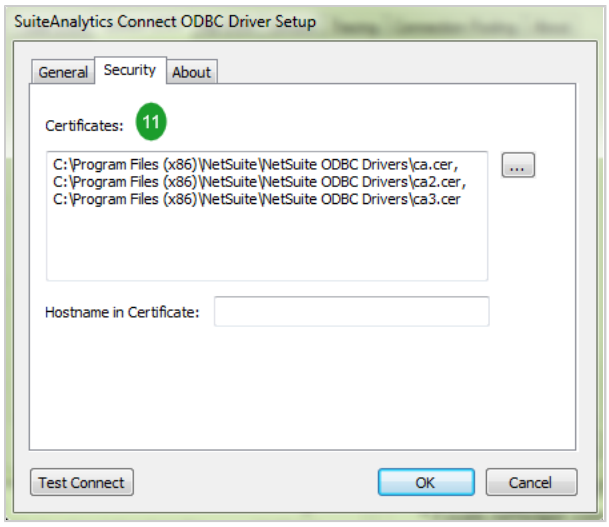
The basic format of a connection string includes a series of keyword/value pairs separated by semicolons. The following example shows the keywords and values for a simple connection string to connect to the SuiteAnalytics Connect server:

```
DRIVER=NetSuite Drivers 32bit;Host=odbcserver.netsuite.com;Port=1708;Encrypted=1;ServerDataSour
ce=NetSuite.com;Truststore= C:\\Program Files (x86)\\NetSuite\\NetSuite ODBC Drivers\\ca.cer, C
:\\Program Files (x86)\\NetSuite\\NetSuite ODBC Drivers\\ca2.cer, C:\\Program Files (x86)\\NetS
uite\\NetSuite ODBC Drivers\\ca3.cer;LogonId=test@netsuite.com;Password=test123456;CustomProper
ties=AccountID=123456;RoleID=3
```

These attributes correspond to the following fields in the driver setup dialog:







	Field	Connection Attribute
1	Name	DSN
2	Service Host	Host
3	Service Port	Port
4	Service Data Source	ServerDataSource
5	Account ID	AccountID
6	Role ID	RoleID
7	User Name	LogonID
8	Password	Password
9	Account ID	AccountID
10	Role ID	RoleID
11	Certificates	C:\Program Files (x86)\NetSuite\NetSuite ODBC Drivers\ca.cer C:\Program Files (x86)\NetSuite\NetSuite ODBC Drivers\ca2.cer C:\Program Files (x86)\NetSuite\NetSuite ODBC Drivers\ca3.cer

This section describes the ODBC connection attributes supported by the ODBC driver on Linux, UNIX, and Windows, on 32-bit and 64-bit platforms.

Connection attributes are listed alphabetically by their names that appear on the driver setup dialog. The connection string attribute name, along with its short name, is listed immediately underneath the user interface name. The list includes long and short names and provides a description of each attribute. Short names are enclosed in parentheses **( )**.

In many cases, the user interface name and the attribute name are the same. However, some connection string attributes may not appear in the user interface at all.

Custom Properties

SuiteAnalytics Connect requires two custom properties attributes: Account ID and Role ID.

AccountID

Attribute	AccountID
Description	Required. The NetSuite account ID.
Default	None

Role ID

Attribute	RoleID
Description	Required. The NetSuite role ID for the specified account
Default	None

These custom properties attributes should be added as the last attribute in the connection string in the following format:

```
CustomProperties=AccountID=123456;RoleID=3
```

Driver

Attribute	Driver
Description	Required when DSN-less connection is used. For more details, see DSN-less connection.
Valid Values	■ on Windows: **NetSuite Drivers 32bit** or **NetSuite Drivers 64bit** ■ on Linux: **NetSuite ODBC Drivers 7.2**
Default	None

DSN

Attribute	DSN (DSN)
Description	Optional. Specifies the SuiteAnalytics Connect data source configuration name for established connections.
Valid Values	A string containing the name of the DSN that is used to connect to NetSuite. For example, **NetSuite**.
Default	Empty string

Encrypted (SSL)

Attribute	Encrypted (ENC)

Description	Required. Enables the use of SSL encryption for data exchanged with the SuiteAnalytics Connect service server. Must be included in the connection string.
Valid Values	**1**
Default	**0**

Password

Attribute	Password (PWD)
Description	Required. The password used to log in to NetSuite.
Valid Values	A string containing a password used to log in to NetSuite.
Default	None

Service Data Source

Attribute	ServerDataSource (SDSN)
Description	Required. The name of the SuiteAnalytics Connect server data source to be used for the connection.
Valid Values	**NetSuite.com**
Default	None

Service Host

Attribute	Host (HST)
Description	Required. The TCP/IP address of the SuiteAnalytics Connect server, specified as a host name.
Valid Values	The Connect Service host name. The host name you should use for your connection is displayed in the **Service Host** field on the SuiteAnalytics Connect Driver Download page, under **Your Configuration**. If the Hostname in Certificate attribute is used, see Authentication Using Server Certificates for ODBC to check which value to use.
Default	None

Service Port

Attribute	Port (PRT)
Description	Required. The TCP/IP port on which the SuiteAnalytics Connect server is listening.
Valid Values	**1708**
Default	None

Truststore

Attribute	Truststore (TS)
Description	Required. A comma-separated list of valid Certificate Authorities (CAs) to be used for server authentication.

	The supported certificate files ca.cer, ca2.cer, and ca3.cer can either be found in the driver installation folder or on the SuiteAnalytics Connect driver download page For more information, see Authentication Using Server Certificates for ODBC.
Valid Values	A string containing a comma-separated list of valid Certificate Authorities (CAs) provided by NetSuite. For example, **Truststore=C:\Program Files (x86)\NetSuite\NetSuite ODBC Drivers\ca.cer, C:\Program Files (x86)\NetSuite\NetSuite ODBC Drivers\ca2.cer**. These certificates are DER-encoded.
Default	None

User Name

Attribute	LogonID (UID)
Description	Required. The email used to log in to NetSuite. This attribute is required.
Valid Values	A string containing an email used to log in to NetSuite.
Default	None

Validate Server Certificate

Attribute	ValidateServerCertificate (VSC)
Description	Optional. If you choose to use this connection attribute, make sure it is set to **1**. Determines whether the driver validates the certificate sent by the SuiteAnalytics Connect server. During SSL server authentication, the SuiteAnalytics Connect server sends a certificate issued by a trusted Certificate Authority (CA). This certificate is validated against the certificate files specified in the **Truststore** attribute.
Valid Values	**0** or **1** If you use this attribute, it must be set to **1**. When set to **1**, the driver validates the certificate sent by the SuiteAnalytics Connect server against the certificate files specified in the **Truststore** attribute. Any certificate from the server must be issued by a trusted CA. When set to **0**, the driver does not validate the certificate sent by the database server. The driver ignores any certificates specified in the **TrustStore** attribute.
Default	**1**

Authentication Using Server Certificates for ODBC

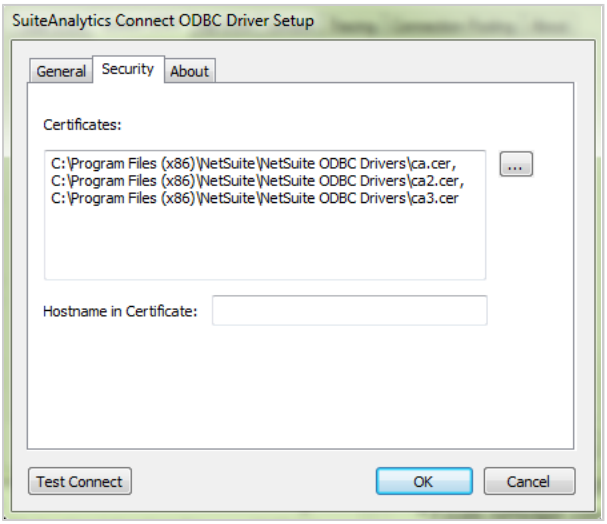
ODBC uses TLS-secured connections. Currently, only TLS 1.2 is supported. Further, we actively look for new vulnerabilities and respond as needed to new threats. We also highly recommend that users leverage the benefits of authentication with server certificates.

To enable authentication using certificates:

1. Open the ODBC Administrator tool. To learn the tool's location in your system, see Verifying the ODBC Driver Installation on Windows.
2. On the **System DSN** tab, configure the DSN you want to secure.
3. Go to the **Security** tab.
4. Fill in the **Certificates** field with the following values, depending on your driver version. You can use the Browse button ... to locate the certificate files.

- 32-bit driver: C:\Program Files (x86)\NetSuite\NetSuite ODBC Drivers\ca.cer, C:\Program Files (x86)\NetSuite\NetSuite ODBC Drivers\ca2.cer, C:\Program Files (x86)\NetSuite\NetSuite ODBC Drivers\ca3.cer
- 64-bit driver: C:\Program Files\NetSuite\NetSuite ODBC Drivers\ca.cer, C:\Program Files \NetSuite\NetSuite ODBC Drivers\ca2.cer, C:\Program Files\NetSuite\NetSuite ODBC Drivers \ca3.cer

We recommend that you leave the **Hostname in Certificate** field blank.



Upgrading an ODBC Driver

To upgrade from a previous version of a Connect for ODBC driver:

- If you use a Windows version of the driver, you can update the driver using the ODBC for Windows driver installer. For more information, see ODBC Installation on Windows - Installer Only.

  Important: If you have the NetSuite ODBC 6.0 driver installed, it cannot be updated using the installer: please remove it before installing the new driver version. To learn how to determine your driver version, see Determining Your Connect Driver Version

- If you use a Linux version of the driver, you need to remove the driver before installing the new version. Back up the existing ODBC driver files by moving them to a different location. To remove the driver, delete the files in its installation folder. The uninstall process may require a restart of your system. When the older version of the driver is removed, install the new version by following the instructions in Installing the Latest Driver on Linux.

Note: As of February 1, 2016, NetSuite no longer supports the legacy ODBC 6.0 driver. To ensure that your SuiteAnalytics Connect integrations continue to work correctly, please upgrade to the latest SuiteAnalytics Connect ODBC driver.

Accessing the Connect Service Using Microsoft Excel

You can use the SuiteAnalytics Connect ODBC driver to load your NetSuite data to Microsoft Excel workbooks.

There are several ways to do this:

- Run a query over a set of tables in the Connect schema and load the results to an Excel worksheet.
- Use Data Connection Wizard to load the table data to an Excel worksheet.
- Use Microsoft Query over a table or a set of tables and load the results to an Excel worksheet.

Before you can configure Microsoft Excel to pull data from the SuiteAnalytics Connect data source, you need to download and install the latest SuiteAnalytics Connect ODBC driver and make sure it is connected to your NetSuite data source. To learn how to download and install the driver, see Downloading and Installing the ODBC Driver for Windows. To learn how to test your connection, see Configuring the ODBC Data Source on Windows.

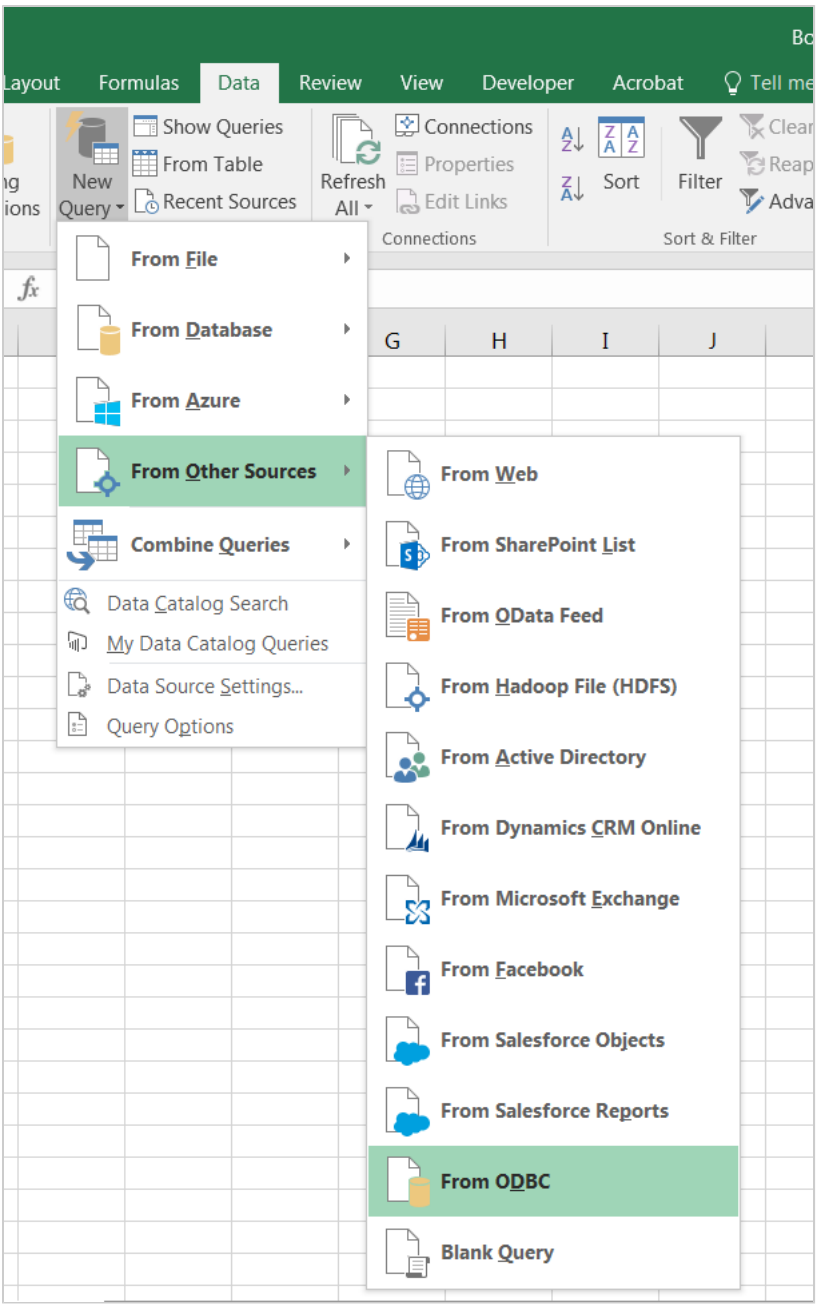
Using a query

One of the most convenient ways to run a database query in Microsoft Excel is to use the New Query option. This option is available starting with Microsoft Excel version 2016. However, if you use Microsoft Excel 2010 or 2013, you can install the Power Query add-in to be able to use this option. For details, see https://www.microsoft.com/en-us/download/details.aspx?id=39379.

Important: The following instruction is based on a 32-bit SuiteAnalytics Connect ODBC driver used in conjunction with 32-bit Microsoft Excel 2016. Please note that the examples in this instruction are provided for illustrative purposes only. The tables available in your NetSuite account may vary.

To load data using a query:

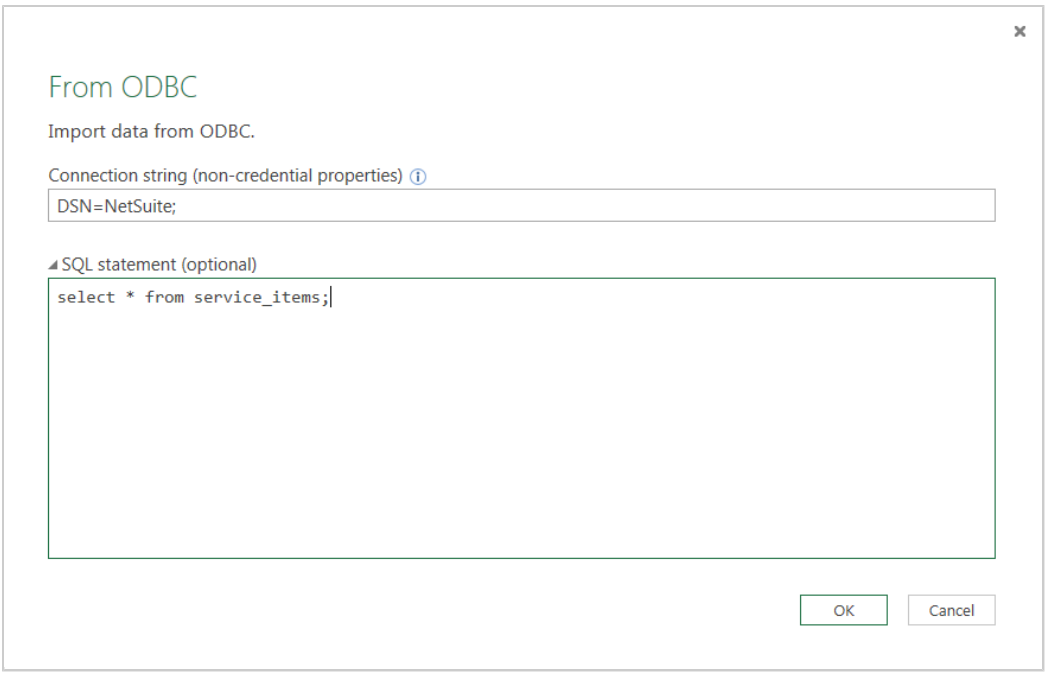
1. In Microsoft Excel, go to the **Data** tab and select **New Query** > **From Other Sources** > **From ODBC**.



2. In the connection string field, type **DSN=NetSuite;** to use the Connect driver DSN for connection. To learn how to check your NetSuite DSN, see Configuring the ODBC Data Source on Windows. For DSN-less connection string options, see Connecting Using a Connection String.
3. Expand the **SQL Statement** area and enter your SQL query. For example, to run a query over all columns in the Service Items table, enter **select * from service_items;**.

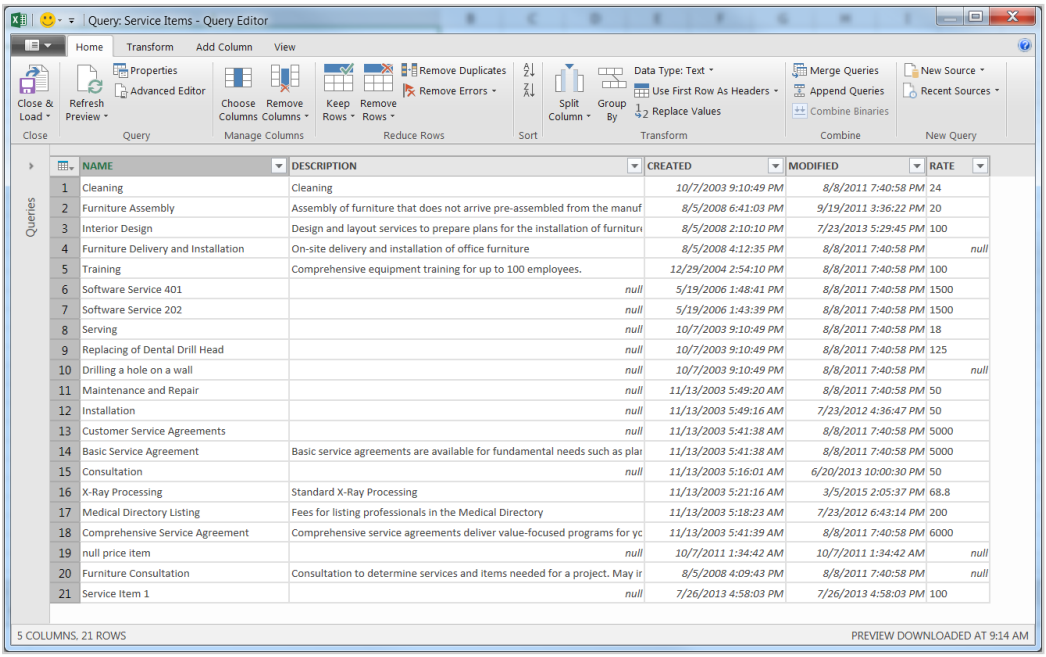
   To explore the tables available in the Connect schema, you can use the Connect Browser. To learn more about the Connect Browser, see Working with the SuiteAnalytics Connect Browser.

   If you don't add any query in the **SQL Statement** field, you can click **OK** and select a table or a set of tables you want to open in the Query Editor. However, there can be Excel limitations to using this option. In this case, try running the query again, adding an explicit query statement in the **SQL Statement** field.

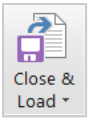


4. Click **OK**.
5. Provide your NetSuite login and password, if prompted, and click **Connect**.
6. The Query Editor opens. You can rearrange the columns in your query, removing those you do not need.

   To learn more about the Query Editor, see https://support.office.com/en-us/article/Introduction-to-the-Query-Editor-Power-Query.



7. When ready, click **Close & Load** to load your query data to your Excel workbook. By default, your data will be loaded to a new worksheet. For more options, expand this menu and click **Close & Load To**.



Note: Due to an Oracle limitation, queries over SuiteAnalytics Connect schema tables including more than 999 fields will not run. This may happen when querying over tables that have many custom fields or when joining multiple tables in a single query and trying to retrieve all their fields. For example, if you add too many custom fields to the Transaction record type, exceeding the 999 fields per table limit, you may get the following error: “Error: Could not find any column information for table:transactions”.

Using Data Connection Wizard or Microsoft Query

You can also connect to the NetSuite data source using the Data Connection Wizard or Microsoft Query and select the tables you would like to import into your Excel workbook. In this case, you can choose whether to display the imported data as a table, a PivotTable report, or a PivotChart. To learn more about Data Connection Wizard and Microsoft Query, please refer to https://support.office.com/en-us/article/Overview-of-connecting-to-importing-data.

Accessing the Connect Service Using a JDBC Driver

Installing a Connect JDBC driver enables you to connect a Java application to the Connect Service. The Connect JDBC driver is a database driver implementation that enables you to use JDBC API to communicate with the SuiteAnalytics Connect service. The advantages of this type of driver include:

- Provides Java platform independence.
- Offers a direct connection from the client application to the Connect Service.

The following table lists the tasks you must complete to set up your environment, download, and install the JDBC driver.

Task	Description
Before you begin.	For more information, see Prerequisites.
Download and install the driver.	For more information, see Installing the JDBC Driver for Windows or Installing the JDBC Driver for Linux.
Register the JDBC driver in your environment by adding Jar File Location to your Class Path.	For more information, see JDBC Code Examples.
Optionally, you can enable authentication with server certificates for the Connect for JDBC driver.	For more information, see Authentication Using Server Certificates for JDBC.

Prerequisites

Before you begin the download and installation process, complete the following tasks:

- Ensure that the Java Virtual Machine (JVM) is installed on your computer. The latest version of the SuiteAnalytics Connect JDBC driver requires at least Java SE 7.
- To connect to the NetSuite account, environment, and role that you used to log in, your JDBC data source configuration must match the values listed in the Reviewing Configuration Information section.
- For more information, see Prerequisites for Using the Connect Service.

Installing the JDBC Driver for Windows

The binary and the download package are the same for both 32–bit and 64–bit JDBC drivers. No installation bundle exists for JDBC drivers, only an installer package. If you are using a Windows operating system, you can use either an .exe install package or a zip file.

To install the JDBC Driver for Windows:

1. In the Settings portlet on your NetSuite home page, click **Set Up SuiteAnalytics Connect**. If the download link is not displayed, you need to enable the Connect Service. For more information, see Enabling the Connect Service Feature.
2. In the upper left corner, select your platform.
3. Click the **Download** button next to the JDBC Driver.

4. When the download is complete, run the installer and follow the wizard's instructions.
5. When the installation is complete, locate the NQjc.jar file. This file contains the driver that you can register in your Java environment by adding the NQjc.jar file to your class path. For more information, see JDBC Code Examples.

Installing the JDBC Driver for Linux

The binary and the download package are the same for both 32–bit and 64–bit JDBC drivers. No installation bundle exists for JDBC drivers, only an installer package. For a Linux operating system, you must use a .zip file to install the JDBC driver. In the install procedure, you use the install package to extract the jar file from the package.

To install the JDBC driver for Linux:

1. In the **Settings** portlet on your NetSuite home page, click **Set Up SuiteAnalytics Connect**. If the download link is not displayed, you need to enable the Connect Service. For more information, see Enabling the Connect Service Feature.
2. In the upper left corner, select your platform.
3. Click the **Download** button next to the JDBC Driver.
4. Agree to the terms of service to be able to start the download. You cannot download the driver archive unless you agree to the terms of service.
5. Save the installation .zip file to your computer.
6. When the download is complete, extract the content of the .zip file to your desired location.
7. In the location where you extracted the content of the .zip file, locate the NQjc.jar file. This file contains the driver that you can register in your Java environment by adding the NQjc.jar file to your class path. For more information, see JDBC Code Examples.

Installing the JDBC Driver for OS X

For a OS X operating system, you must use a .zip file to install the JDBC driver.

To install the JDBC driver for OS X:

1. In the **Settings** portlet on your NetSuite home page, click **Set Up SuiteAnalytics Connect**. If the download link is not displayed, you need to enable the Connect Service. For more information, see Enabling the Connect Service Feature.
2. In the upper left corner, select **OS X**.
3. Click the **Download** button next to the JDBC Driver.
4. Agree to the terms of service to be able to start the download. You cannot download the driver archive unless you agree to the terms of service.
5. Save the installation .zip file to your computer.
6. When the download is complete, extract the content of the .zip file to your desired location.

Specifying Connection Properties

You can specify connection properties using a connection URL, the JDBC Driver Manager, or JDBC data sources. For a list of the connection properties, see JDBC Connection Properties.

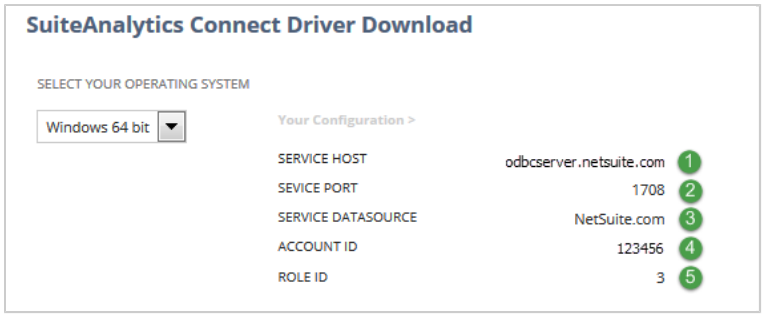
Using the Connection URL

The following example shows a typical SuiteAnalytics Connect JDBC driver connection URL:

```
jdbc:ns://odbcserver.netsuite.com:1708;ServerDataSource=NetSuite.com;Encrypted=1;CipherSuites=T
LS_RSA_WITH_AES_128_CBC_SHA;CustomProperties=(AccountID=123456;RoleID=3);
```

The user and password connection properties are not shown. These properties are usually specified in the connection properties stored in the java.util.Properties object. Alternatively, they can be supplied as parameters to the getConnection() method.

The connection properties used in this connection URL correspond to the following fields on the drivers download page:



	Field	Connection Property
1	Service Host	ServerName
2	Service Port	PortNumber
3	Service Data Source	ServerDataSource
4	Account ID	CustomProperties (AccountID)
5	Role ID	CustomProperties (RoleID)

For examples of connection URLs, see Connection URL Used with JDBC Driver Manager Example.

Using the JDBC Driver Manager

In order of precedence, you can specify connection properties using:

- **getConnection(url, user, password)**, where **user** and **password** are specified using the **getConnection** method defined in **java.sql.DriverManager**.

```
DriverManager.getConnection(
connectionURL,
user,
password);
```

- **java.util.properties** object.

```
DriverManager.getConnection(
connectionURL, (java.util.Properties)
properties);
```

- Connection URL specified using the URL parameter of the **getConnection** method defined in **java.sql.DriverManager**.

```
DriverManager.getConnection(
connectionURL)
```

For details, see JDBC Code Examples.

Using JDBC Data Sources

In order of precedence, you can specify connection properties using:

- **getConnection(user, password)**, where **user** and **password** are specified using the **getConnection** method defined in **javax.sql.DataSource**.

```
getConnection("test@netsuite.com", "test123456")
```

- JDBC DataSource object.

For details, see JDBC Data Source Example.

JDBC Connection Properties

This section lists the JDBC connection properties supported by the SuiteAnalytics Connect service and describes each property. The properties have the form:

```
property = value
```

Connection property names are not case sensitive: they may contain both lowercase and uppercase letters.

CipherSuites

Description	Required. Enables authentication with server certificates. For more information, see Authentication Using Server Certificates for JDBC.
Valid Values	For supported cipher suites, see Supported Cipher Suites for JDBC.
Example	`jdbc:ns://odbcserver.netsuite.com:1708;CipherSuites=TLS_RSA_WITH_AES_128_CBC_SHA`
Default	None
Data type	string

CustomProperties

SuiteAnalytics Connect requires two custom properties: AccountID and RoleID.

AccountID

Description	Required. The NetSuite account ID.

Default	None

RoleID

Description	Required. The NetSuite role ID for the specified account.
Default	None

These custom properties should be added as the last property in the connection URL in the following format:

```
CustomProperties=(AccountID=123456;RoleID=3)
```

Encrypted

Description	Required. Enables the use of SSL encryption for the data exchanged with the SuiteAnalytics Connect service server. Must be included in the connection URL.
Valid Values	**1**
Example	`jdbc:ns://odbcserver.netsuite.com:1708;encrypted=1`
Default	**0**
Data type	boolean

Password

Description	Required. The password used to log in to NetSuite.
Example	For examples of using this connection property with the getConnection attribute, see JDBC Code Examples.
Default	None
Data type	String. We recommend that the getConnection() method be used instead of typing the password directly into the connection string.

PortNumber

Description	Required. The TCP/IP port on which the SuiteAnalytics Connect server is listening.
Valid Values	**1708**
Default	None
Data type	String

ServerDataSource

Description	Required. The name of the SuiteAnalytics Connect server data source to be used for the connection.
Valid Values	**NetSuite.com**

Default	None
Data type	String

ServerName

Description	Required. The TCP/IP address of the SuiteAnalytics Connect server, specified as a host name.
Valid Values	Depends on the data center used. For example, accepted values include: ■ **odbcserver.netsuite.com** ■ **odbcserver.na1.netsuite.com** ■ **odbcserver.na2.netsuite.com** ■ **odbcserver.sandbox.netsuite.com** For a complete list of NetSuite data centers, see the help topic Understanding NetSuite URLs and Data Centers.
Default	None
Data type	String

TrustStore

Description	Optional. A path to a valid truststore containing the security certificates to be used for server authentication. **Note:** The certificates provided on the SuiteAnalytics Connect driver download page are usually already included in the internal Java truststore. In this case, it is not required to add the TrustStore attribute to the connection string. Specifies the directory of the truststore file to be used when SSL is enabled using the **Encrypted** property and when service authentication is used. The truststore file contains a list of the Certificate Authorities (CAs) that the client trusts. This value overrides the directory of the Java truststore file specified by the javax.net.ssl.trustStore Java system property. If this property is not specified, the truststore directory is specified by the javax.net.ssl.trustStore Java system property. The truststore file is a Java keystore file, generated by the keytool utility, which is a part of the JDK. For more information on configuring a truststore using the keytool utility, see Setting Up a Truststore and http://docs.oracle.com/cd/E19509-01/820-3503/6nf1il6er/index.html. The TrustStore property is ignored if the **ValidateServerCertificate** property is set to **false**.
Valid Values	A string containing the path to a truststore file.
Default	None
Data type	String

User

Description	Required. The email used to log in to NetSuite.
Valid Values	A string containing an email used to log in to NetSuite.
Default	None
Data type	String

ValidateServerCertificate

Description	Optional. If you choose to use this connection property, make sure it is set to **true**. Determines whether the driver validates the certificate sent by the SuiteAnalytics Connect server. During SSL server authentication, the SuiteAnalytics Connect server sends a certificate issued by a trusted Certificate Authority (CA). The required CAs are usually included in the Java truststore, but you can also specify them using the **TrustStore** property.
Valid Values	**true** or **false** If you use this property, it must be set to **true**. When set to **true**, the driver validates the certificate sent by the SuiteAnalytics Connect server. Any certificate from the server must be issued by a trusted CA in the truststore file. When set to **false**, the driver does not validate the certificate sent by the SuiteAnalytics Connect server. The driver ignores any truststore information specified by the **TrustStore** property.
Default	**true**
Data type	Boolean

JDBC Code Examples

To use the Connect for JDBC driver when writing your own application, the NQjc.jar file needs to be on the Classpath. The Connect for JDBC driver provides two primary options for connecting to the Connect Service: using a connection URL in conjunction with the JDBC Driver Manager and using the JDBC DataSource class. The two approaches produce the same results.

Note: The installation package contains additional code examples.

Connection URL Used with JDBC Driver Manager Example

You can use code like the following example to connect through a connection URL used with JDBC Driver Manager. Replace sample values with the values listed in the Reviewing Configuration Information section.

```
import java.sql.Connection;
import java.sql.DriverManager;

public class ConnectionTest
{
   public static void main(String[] args) throws Exception
   {
      Connection connection = null;
      try
      {
         Class.forName("com.netsuite.jdbc.openaccess.OpenAccessDriver");
         String connectionURL =
                  "jdbc:ns://
odbcserver.netsuite.com:
1708;" +
                  "ServerDataSource=
NetSuite.com;" +
                  "Encrypted=
1;" +
                  "CipherSuites=
```

```
TLS_RSA_WITH_AES_128_CBC_SHA;" +
                  "CustomProperties=(AccountID=
123456;RoleID=
3)";
         connection = DriverManager.getConnection(connectionURL, "
User", "
Password");
         System.out.println("Connection success");
      }
      finally
      {
         if (connection != null)
               connection.close();
      }
   }
}
```

JDBC Data Source Example

You can use code like the following example to connect through JDBC Data Source. Replace bold text with the values listed in the Reviewing Configuration Information section.

```
import com.netsuite.jdbcx.openaccess.OpenAccessDataSource;
import java.sql.Connection;

public class ConnectionTest
{
   public static void main(String[] args) throws Exception
   {
         Connection connection = null;
         try
         {
            OpenAccessDataSource sds = new OpenAccessDataSource();
            sds.setServerDataSource("
NetSuite.com");
            sds.setServerName("
odbcserver.netsuite.com");
            sds.setPortNumber(
1708);
            sds.setCipherSuites("TLS_RSA_WITH_AES_128_CBC_SHA");
            sds.setCustomProperties("(AccountID=
123456;RoleID=
3)");
            sds.setEncrypted(
1);
            connection = sds.getConnection("
User", "
Password");
         }
         finally
         {
            if (connection != null)
                  connection.close();
         }
```

```
    }
}
```

Authentication Using Server Certificates for JDBC

JDBC uses TLS-secured connections. Currently, only TLS 1.2 is supported. Further, we actively look for new vulnerabilities and respond as needed to new threats. We also highly recommend that users leverage the benefits of authentication with server certificates. To enable authentication with server certificates, you must change the Cipher Suite in the JDBC driver from anonymous to non-anonymous.

To use server certificates authentication, your connection URL must include the following connection properties:

- **CipherSuites** – should contain one of the supported cipher suites. For the list of the supported cipher suites, see Supported Cipher Suites for JDBC.

In addition, the connection URL may include the following properties that are also used for certificate authentication:

- **TrustStore** – Optional. If specified, should contain the path to a valid truststore that includes the security certificates to be used for server authentication.

  The required certificates are usually already included in the internal Java truststore, so it is not required to add the TrustStore property to the connection URL. If your Java truststore doesn't include the required certificates, you can download them from the SuiteAnalytics Connect drivers download page and add them to the Java truststore or create a new truststore for them using the keytool utility. For more information, see Setting Up a Truststore.
- **ValidateServerCertificate** – Optional. If specified, should be set to **true**.

Supported Cipher Suites for JDBC

To enable authentication with server certificates, you must change the default Cipher Suite to one of the following:

- TLS_RSA_WITH_AES_128_CBC_SHA
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA

Setting Up a Truststore

All the required security certificates are usually already included in the Java truststore, so it is not required to set up a new truststore for them.

However, if your Java truststore does not include the required certificates, you can download them from the SuiteAnalytics Connect drivers download page. Then you can use the Java keytool utility to create a new truststore for the certificates or import them into your Java truststore. To learn more about the keytool utility and setting up truststores, see docs.oracle.com/cd/E19509-01/820-3503/6nf1il6er/index.html. You can also use the following instructions for reference.

To create a new truststore:

1. In the **Settings** portlet on your NetSuite home page, click **Set Up SuiteAnalytics Connect**. If the download link is not displayed, you need to enable the Connect Service. For more information, see Enabling the Connect Service Feature.
2. Click the **Download** button next to CA CERTIFICATES.

3. When the certificates .zip file is downloaded, extract its contents, the ca.cer and ca2.cer files, to any location on your computer.
4. Find the keytool utility. It is usually located in the `\bin` folder inside your Java installation folder, for example, `C:\Program Files\Java\jdk1.7.0_71\bin\`.

   If you have multiple Java versions installed, choose the keytool utility for the Java version you use with the SuiteAnalytics Connect JDBC driver.
5. Run the keytool utility, using the following command to create a new truststore. Replace `CertificatesFolder` with the path to the folder where you extracted the downloaded certificates and `TrustStoreLocation` with the path where you want to create the new truststore.

   ```
   keytool -import -file "
   CertificatesFolder\ca.cer" -alias nsca -keystore "
   TrustStoreLocation\NSCA"
   ```

   This will create an NSCA truststore in the location you specified.
6. To import the second certificate into the new truststore, run the following command, replacing `CertificatesFolder` with the path to the folder where you extracted the downloaded certificates and `TrustStoreLocation` with the path to the newly created truststore.

   ```
   keytool -import -file "
   CertificatesFolder\ca2.cer" -alias nsca2 -keystore "
   TrustStoreLocation\NSCA"
   ```
7. To import the third certificate into the new truststore, run the following command, replacing `CertificatesFolder` with the path to the folder where you extracted the downloaded certificates and `TrustStoreLocation` with the path to the newly created truststore.

   ```
   keytool -import -file "
   CertificatesFolder\ca2.cer" -alias nsca3 -keystore "
   TrustStoreLocation\NSCA"
   ```
8. When you have created the new truststore and imported all the certificates, modify your JDBC driver connection strings, adding the TrustStore property. The TrustStore property should contain the path to the truststore that you have just created.

   For example, if you created the NSCA truststore in `C:\Program Files\NetSuite\`, your connection URL should look like the following:

   ```
   jdbc:ns://odbcserver.netsuite.com:1708;ServerDataSource=NetSuite.com;Encrypted=1;TrustStore=
   C:\\Program Files\\NetSuite\\NSCA;CipherSuites=TLS_RSA_WITH_AES_128_CBC_SHA;CustomProperties=(AccountID=123456;RoleID=3);
   ```

Alternatively, you can add the certificates to your Java truststore. In this case, you do not have to add the TrustStore property to your connection URL.

To add the certificates to your Java truststore:

1. In the **Settings** portlet on your NetSuite home page, click **Set Up SuiteAnalytics Connect**. If the download link is not displayed, you need to enable the Connect Service. For more information, see Enabling the Connect Service Feature.
2. Click the **Download** button next to CA CERTIFICATES.
3. When the certificates .zip file is downloaded, extract its contents, the ca.cer and ca2.cer files, to any location on your computer.

4. Find the keytool utility. It is usually located in the `\bin` folder inside your Java installation folder, for example, `C:\Program Files\Java\jdk1.7.0_71\bin\`.

   If you have multiple Java versions installed, choose the keytool utility for the Java version you use with the SuiteAnalytics Connect JDBC driver.

5. Run the keytool utility, using the following command to add the first certificate, ca.cer, to your Java truststore. Replace `CertificatesFolder` with the path to the folder where you extracted the downloaded certificates and `JavaTrustStore` with the path to the `\lib\security\cacerts` file in your Java installation folder, for example, `C:\Program Files\Java\jdk1.7.0_71\lib\security\cacerts`. You may also need to specify the truststore password to access your Java truststore.

   ```
   keytool -import -file "
   CertificatesFolder\ca.cer" -alias nsca -keystore "
   JavaTrustStore"
   ```

6. To add your second certificate to your Java truststore, run the following command, replacing `CertificatesFolder` with the path to the folder where you extracted the downloaded certificates and `JavaTrustStore` with the path to the `\lib\security\cacerts` folder in your Java installation folder, for example, `C:\Program Files\Java\jdk1.7.0_71\lib\security\cacerts`. You may also need to specify the store password to access your Java truststore.

   ```
   keytool -import -file "
   CertificatesFolder\ca2.cer" -alias nsca2 -keystore "
   JavaTrustStore"
   ```

7. To add your third certificate to your Java truststore, run the following command, replacing `CertificatesFolder` with the path to the folder where you extracted the downloaded certificates and `JavaTrustStore` with the path to the `\lib\security\cacerts` folder in your Java installation folder, for example, `C:\Program Files\Java\jdk1.7.0_71\lib\security\cacerts`. You may also need to specify the store password to access your Java truststore.

   ```
   keytool -import -file "
   CertificatesFolder\ca2.cer" -alias nsca3 -keystore "
   JavaTrustStore"
   ```

If you added all three certificates to your Java truststore, you do not have to add the TrustStore property to your connection URLs.

Using a Non-Anonymous Cipher Suite to Establish a Connection

Familiarize yourself with JDBC examples. For more information, see JDBC Code Examples.

To establish a connection using a non-anonymous Cipher Suite:

1. Modify connection properties by specifying the supported Cipher Suites (only names of supported Cipher Suites can be used). For more information, see Supported Cipher Suites for JDBC.

   When a connection string is used, for example, fill in the bold Cipher Suite and valid certificate as shown in the following code block in bold.

   ```
   jdbc:ns://odbcserver.netsuite.com:1708;ServerDataSource=NetSuite.com;Encrypted=1;CustomProperti
   ```

```
es=(AccountID=123456;RoleID=3);
CipherSuites=TLS_RSA_WITH_AES_128_CBC_SHA;
```

When the OpenAccessDataSource class is used, for example, fill in the bold Cipher Suite and valid certificate for the server as shown in bold in the following code block.

```
OpenAccessDataSource sds = new OpenAccessDataSource();
sds.setServerDataSource("NetSuite.com");
sds.setServerName("odbcserver.netsuite.com);
sds.setPortNumber(1708);
sds.setCustomProperties(String.format("(AccountID=123456;RoleID=3)",
accountId, roleId));
sds.setEncrypted(1);
sds.setCipherSuites("TLS_RSA_WITH_AES_128_CBC_SHA");
```

2. Optionally, you can specify a host name for certificate validation. The driver compares the specified host name to the value included in the certificate. When a connection string is used, fill in the Host Name in Certificate, as shown in the following code block in bold.

```
jdbc:ns://odbcserver.netsuite.com:1708;ServerDataSource=NetSuite.com;Encrypted=1;CustomProperti
es=(AccountID=123456;RoleID=3);
CipherSuites=TLS_RSA_WITH_AES_128_CBC_SHA;HostNameInCertificate=odbcserver.netsuite.com
```

When a OpenAccessDataSource class is used, fill in the HostNameInCertificate method in bold, as shown in the following code block.

```
OpenAccessDataSource sds = new OpenAccessDataSource();
sds.setServerDataSource("NetSuite.com");
sds.setServerName("odbcserver.netsuite.com);
sds.setPortNumber(1708);
sds.setCustomProperties(String.format("(AccountID=123456;RoleID=3)", accountId, roleId));
sds.setEncrypted(1);
sds.setCipherSuites("TLS_RSA_WITH_AES_128_CBC_SHA");
sds.setHostNameInCertificate("odbcserver.netsuite.com");
```

Accessing the Connect Service Using an ADO.NET Data Provider

The SuiteAnalytics Connect for ADO.NET data provider enables you to access the Connect Service from the Microsoft .NET platform. This type of data provider is appropriate for multi-tier architectures where you want to perform more complex data processing and analysis. Using a Connect ADO.NET data provider provides increased scalability for applications. The two main components of ADO.NET for accessing and manipulating data are the .NET Framework data providers and the DataSet. You can incorporate the Connect Service as one data source (of possibly multiple data sources) in to your ADO.NET schema.

The following table lists the tasks you must complete to set up your environment, download, and install the ADO.NET driver.

Task	Description
Prepare for installation	For more information, see Prerequisites.
Download and install the driver	For more information, see Downloading and Installing the ADO.NET Driver.
Access the Connect Service using ADO.NET data provider	For more information, see Accessing the Connect Service Using an ADO.NET Data Provider.

Note: The ADO.NET data provider automatically uses server certificates authentication.

Prerequisites

Before you download and install the driver, make sure that:

- You have both the .NET Framework 2.0 and .NET Framework 4.5 or newer.
- You use either of the following Visual Studio versions: Visual Studio 2012, 2013, or 2015.
- Port 1708 is not blocked by firewall.

Note: The installation does not add the data provider to the Global Assembly Cache.

Review the ADO.NET Data Server Configuration

To connect to the NetSuite account, environment, and role you used to log in, your connection parameters must match the values listed under Your Configuration on the driver download page in NetSuite.

Parameters	Value
Service Host	See your download page. A sample value is “odbcserver.netsuite.com”
Service Port	1708
Service Data Source	NetSuite.com
Account ID	See your download page. A sample value is “123456”
Role ID	See your download page. A sample value is “3”

The value of the Service Host configuration parameter determines the environment of your NetSuite account. The environment can be Production, Release Preview, or Sandbox.

Environment	Service Host
Production	odbcserver.netsuite.com, odbcserver.na1.netsuite.com, or odbcserver.na2.netsuite.com
Release Preview	odbcserver.netsuite.com or odbcserver.eu1.netsuite.com
Sandbox	odbcserver.sandbox.netsuite.com

Downloading and Installing the ADO.NET Driver

The following steps provide instructions to install a SuiteAnalytics Connect ADO.NET driver. For more information on downloading the ADO.NET driver, see SuiteAnalytics Connect Setup.

To install the ADO.NET driver:

1. In the **Settings** portlet on your NetSuite home page, click **Set Up SuiteAnalytics Connect**. If the download link is not displayed, you need to enable the Connect Service. For more information, see Enabling the Connect Service Feature.
2. In the upper left corner, select your Windows platform.
3. Click the **Download** button next to the ADO.NET Driver.
4. When the download is complete, run the **NetSuiteADO.NETDrivers_Windows.exe** installer.
5. Follow the wizard's instructions to complete the installation.

When the installation is complete, you should register the `NetSuite.SuiteAnalyticsConnect.dll` data provider library in the Global Assembly Cache. You can register the library using Windows command prompt or the VsUtil tool found in the driver installation folder. Alternatively, you can include the `NetSuite.SuiteAnalyticsConnect.dll` library directly in your project, by using it in your .NET environment or embedding it in your custom application.

To register the data provider using the VsUtil tool:

1. Locate the `VsUtil.exe` file. It is usually installed in the ADO.NET driver installation folder. The default location on a 64-bit Windows is C:\Program Files (x86)\NetSuite\NetSuite ADO.NET Drivers.
2. At the command prompt, run the following as administrator, in the folder where VsUtil.exe is installed:

   ```
   vsutil.exe -i
   ```

   To learn about other installation options, run `VsUtil.exe` without any parameters.
3. The VsUtil tool does the following:
   - Registers the `NetSuite.SuiteAnalyticsConnect.dll` data provider library in the Global Assembly Cache.
   - Installs the Visual Studio extension that enables you to connect your Visual Studio environment to the SuiteAnalytics Connect service.
   - Adds C# and Visual Basic templates to your Visual Studio library.

When the VsUtil tool finishes running, you can connect your Visual Studio environment to the SuiteAnalytics Connect service. To learn more, see Connecting with the ADO.NET Data Provider.

Connecting with the ADO.NET Data Provider

You can use the ADO.NET data provider to access the Connect Service from your Visual Studio environment or your ADO.NET application.

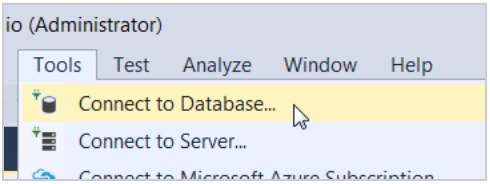
Connecting from Visual Studio

Before you can start using the ADO.NET data provider in your Visual Studio projects, you should set up a connection from your Visual Studio environment to the SuiteAnalytics Connect service.

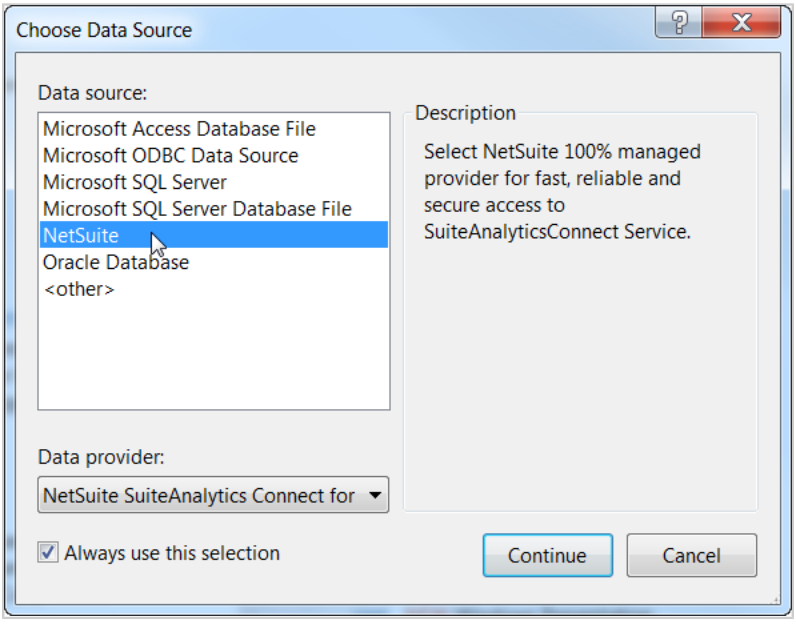
Please note that the following steps are not supported in the Express editions of Visual Studio.

To set up a connection:

1. Open your Visual Studio environment.
2. On the **Tools** menu, click **Connect to Database**.



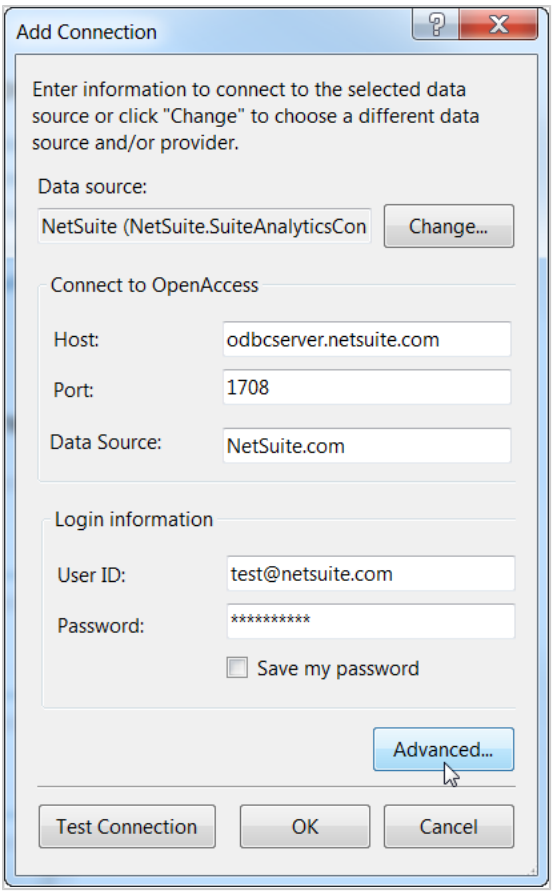
3. In the **Choose Data Source** window, select **NetSuite** and click **Continue**.



4. Fill in the **Host**, **Port**, and **Data Source** fields, using the values found in **Your Configuration** area on the drivers download page.

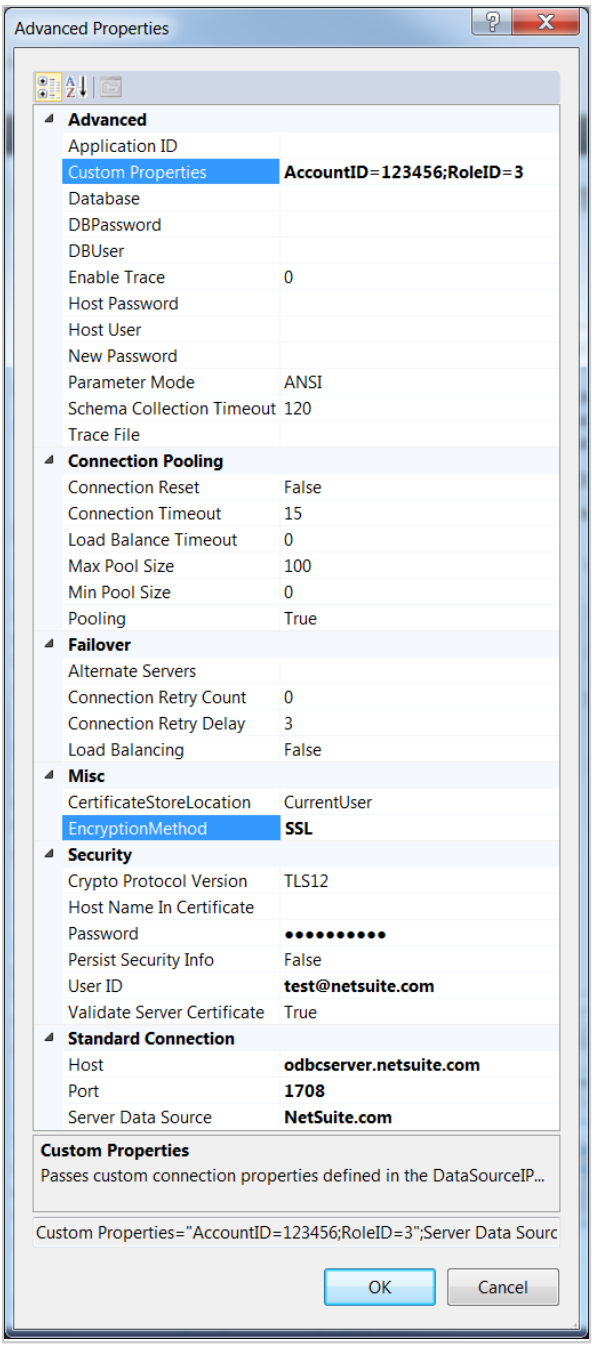
   In the **User ID** and **Password** fields, type the email and password you use to log in to NetSuite.

   When ready, click **Advanced**.

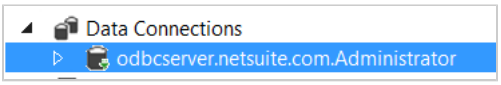


5. In the **Advanced Properties** window, edit the following fields:
   - **Custom Properties** – Type **AccountID**=*your Account ID*;**RoleID**=*your Role ID*, using your NetSuite Account ID and Role ID numbers found on the drivers download page.
   - **EncryptionMethod** – Type **SSL**.

   When ready, click **OK**.



6. Test your connection settings by clicking **Test Connection**.
7. When you have connected successfully, click **OK** to save this connection in your environment.
8. The connection appears in your Visual Studio Server Explorer as an active connection.



You may need to refresh this connection after you close and reopen your Visual Studio environment.

Connecting from an ADO.NET application

As a starting point, you can use the following example to connect to the Connect Service.

To connect using the Common Programming Model:

1. Register the driver `NetSuite.SuiteAnalyticsConnect.dll` file in the Global Assembly Cache or, alternatively, include this file directly in your project. To learn how to register your driver file using the VsUtil.exe tool, see Downloading and Installing the ADO.NET Driver.
2. You can use code like the following example to utilize ADO.NET connectivity in your application. Replace the bold values with the values in Review the ADO.NET Data Server Configuration.

```
using System;
using System.Data;
using NetSuite.SuiteAnalyticsConnect;

namespace AdoExample
{
      class AdoExample
      {
            public static void Main(string[] args)
            {
                  string connectionString =
                        "Host=
odbcserver.netsuite.com;"+
                        "Port=
1708;"+
                        "ServerDataSource=
NetSuite.com;"+
                        "User Id=
test@netsuite.com;"+
                        "Password=
test123456;"+
                        "CustomProperties='AccountID=
123456;RoleID=
3';EncryptionMethod=
SSL;";

                  using (OpenAccessConnection connection = new OpenAccessConnection(connectionS
tring))
                  {
                        Connection.Open();
                        Console.WriteLine("Connection successful");
                  }
            }
      }
}
```

In addition, you can check out sample code examples in the SuiteAnalytics Connect ADO.NET data provider installation folder. The default location on a 64-bit Windows is C:\Program Files (x86)\NetSuite \NetSuite ADO.NET Drivers. If you used the VsUtil tool to register the data provider, the C# and Visual Basic templates should be available in your Visual Studio templates library.

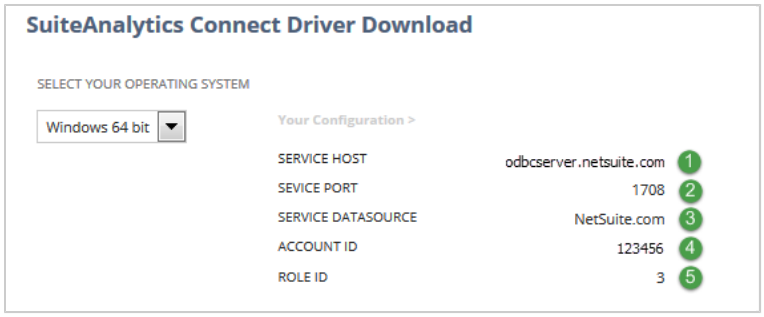
ADO.NET Connection Options

You can modify a connection by specifying connection string options.

The basic format of a connection string includes a series of option/value pairs separated by semicolons. The following example shows the connection options and values for a simple connection string to connect to the SuiteAnalytics Connect service:

```
Host=odbcserver.netsuite.com;Port=1708;EncryptionMethod=SSL;ServerDataSource=NetSuite.com;UserI
D=test@netsuite.com;Password=test123456;CustomProperties='AccountID=123456;RoleID=3';
```

The connection options used in this connection string correspond to the following fields on the driver download page:



	Field	Connection Option
1	Service Host	Host
2	Service Port	Port
3	Service Data Source	ServerDataSource
4	Account ID	CustomProperties (AccountID)
5	Role ID	CustomProperties (RoleID)

Use the following guidelines when specifying a connection string:

- Connection string option names are not case-sensitive. For example, you can use **Password** or **password** as the connection option name. However, the values for the **User ID** and **Password** connection options may be case-sensitive.
- Special characters can be used in the value of the connection string option. If a value contains special characters, enclose it in double quotes.

CustomProperties

SuiteAnalytics Connect requires two custom properties connection options:

- **Account ID** – your NetSuite account ID.
- **Role ID** – the NetSuite role ID for the specified account.

EncryptionMethod

Property	EncryptionMethod

Description	Required. Enables the use of SSL encryption for data exchanged with the SuiteAnalytics Connect service server. Must be included in the connection string.
Valid Values	**SSL**
Default	None

Host

Property	Host
Description	Required. The TCP/IP address of the SuiteAnalytics Connect server, specified as a host name.
Valid Values	Depends on the data center used. For example, accepted values include: ▪ **odbcserver.netsuite.com** ▪ **odbcserver.na1.netsuite.com** ▪ **odbcserver.na2.netsuite.com** ▪ **odbcserver.sandbox.netsuite.com** For a complete list of NetSuite data centers, see the help topic Understanding NetSuite URLs and Data Centers.
Default	None

Password

Property	Password
Description	Required. The password used to log in to NetSuite.
Valid Values	A string containing a password used to log in to NetSuite.
Default	None

Port

Property	Port
Description	Required. The TCP/IP port on which the SuiteAnalytics Connect server is listening.
Valid Values	**1708**
Default	None

Server Data Source

Property	ServerDataSource
Description	Required. The name of the SuiteAnalytics Connect server data source to be used for the connection.
Valid Values	**NetSuite.com**
Default	None

User ID

Property	UserID
Description	Required. The email used to log in to NetSuite.
Valid Values	A string containing an email used to log in to NetSuite.
Default	None

Validate Server Certificate

Property	ValidateServerCertificate
Description	Optional. If you choose to use this connection option, make sure it is set to **1**. Determines whether the driver validates the certificate sent by the SuiteAnalytics Connect server. During SSL server authentication, the SuiteAnalytics Connect server sends a certificate issued by a trusted Certificate Authority (CA).
Valid Values	**0** or **1** If you use this option, it must be set to **1**. When set to **1**, the driver validates the certificate sent by the SuiteAnalytics Connect server. Any certificate from the server must be issued by a trusted CA. When set to **0**, the driver does not validate the certificate sent by the database server.
Default	**1**

Authentication Using Server Certificates for ADO.NET

ADO.NET uses TLS-secured connections. Currently, only TLS 1.2 is supported. Further, we actively look for new vulnerabilities and respond as needed to new threats. We also highly recommend that users leverage the benefits of authentication with server certificates.

The required server certificates are usually already included in your Windows Trusted Root Certification Authorities store. In case they are not, you can download the security certificates from the SuiteAnalytics Connect driver download page and add them manually.

To add the required security certificates to the certificates store:

1. In the **Settings** portlet on your NetSuite home page, click **Set Up SuiteAnalytics Connect**. If the download link is not displayed, you need to enable the Connect Service. For more information, see Enabling the Connect Service Feature.
2. Click the **Download** button next to CA CERTIFICATES.
3. When the certificates .zip file is downloaded, extract it to any location on your computer.
4. Next, go to your Windows **Start** menu, type **mmc** in the search field and press Enter. This opens the Microsoft Management Console.
5. On the **File** menu, select **Add/Remove Snap-in**.
6. Select the **Certificates** snap-in and click **Add**.

   You can set the snap-in to manage security certificates for your user account, service account, or computer account.

7. When you have finished setting up the snap-in, click **OK**.
8. In the console tree, double-click **Certificates**.
9. Right-click the **Trusted Root Certification Authorities** store and select **All Tasks** > **Import**.
10. Follow the wizard's instructions to specify the downloaded security certificate files and import them to your Trusted Root Certification Authorities store.

When the security certificates are imported, the SuiteAnalytics Connect ADO.NET integrations will automatically start using them when connecting to NetSuite.

Removing the ADO.NET Driver

To remove the SuiteAnalytics Connect ADO.NET driver, you first need to use the VsUtil tool that will remove the data provider library from your system and then uninstall the driver like any other Windows program.

To remove the driver:

1. Locate the `VsUtil.exe` file. It is usually installed in the ADO.NET driver installation folder. The default location on a 64-bit Windows is C:\Program Files (x86)\NetSuite\NetSuite ADO.NET Drivers.
2. At the command prompt, run the following as administrator, in the folder where VsUtil.exe is installed:

   ```
   vsutil.exe -u
   ```

   To learn about other options, run `VsUtil.exe` without any parameters.
3. When the VsUtil tool has finished running, in Windows Control Panel, click **Uninstall a program**, select **SuiteAnalytics Connect ADO.NET Drivers**, and click **Uninstall**.

Connect Service Considerations

This section provides best practices for querying the Connect Service using SQL and examples.

- SQL Compliance reviews some nuances regarding SQL queries.
- Exceptions
- Replication of Tables
- Incremental Backups
- Column Joins in the Connect Service uses the Transactions table to show how a table is linked to the schema and how to use the schema to identify the same links.
- Custom Columns, Lists, and Records in the Connect Service shows how NetSuite represents custom columns, lists, and records in a table.
- Driver Access for a Sandbox or Release Preview Account.
- Troubleshooting SuiteAnalytics Connect Connection.

SQL Compliance

You should use generic SQL-92 syntax for the Connect Service. Single-line comments are not supported; use C style comments instead. Take note of the following cases that may affect your queries.

We do not guarantee full SQL-92 compliance and are limited both by OA SDK and Oracle.

- Non-deterministic Column Order Using the Column Selector
- UNION Processing
- UNION ALL
- Nested SELECT Statements
- Interval Types Not Supported
- CAST and CONVERT Function
- Reserved Words

Non-deterministic Column Order Using the Column Selector

Using the * column selector returns a non-deterministic column order.

UNION Processing

UNION query processing is LTR (Left-to-Right Subquery), as per the SQL standard. Consequently, queries that contain inner queries combined with the UNION operator may fail if the names of the selected inner queries' columns do not match. For example, the following query will fail because the company_id and customer_id columns do not match:

```
select *
from
   (
   select COMPANY_ID from NOTES_SYSTEM
   union
   select CUSTOMER_ID from CUSTOMERS
)
```

To prevent this failure, you can rewrite this query to use aliased column names:

```
select *
from
   (select COMPANY_ID as 'ALIASED_COLUMN_1' from NOTES_SYSTEM
   union
   select CUSTOMER_ID as 'ALIASED_COLUMN_1' from CUSTOMERS
   )
```

UNION ALL

The following examples of UNION ALL are supported:

```
select N'' as COL1
from accounts
where account_id < 10
union all
select is_balancesheet as COL1
from accounts
where
account_id < 10

select to_number(sum(openbalance))
from accounts
where account_id < 100
union all
select to_number('')
from accounts
where account_id < 100

select sum(openbalance)
from accounts
where account_id < 100
union all
select to_number('')
from accounts
where account_id < 100
```

Select All Rows Statements

Using 'select * from certain tables' may lead to a timeout on large data volumes. For more information, see Custom Field Limitations.

Smaller Queries

For improved performance, try using multiple smaller queries instead one long query. Also, if too many concurrent queries are run, the concurrent queries may be killed.

Nested SELECT Statements

The following Nested SELECT statements examples are supported:

```
select d1.department_id, (select full_name from departments d2 where d2.department_id = d1.depa
rtment_id) id from departments d1

select * from (select * from departments) d1

select * from (select department_id from departments) d1

select * from departments d1 where department_id in ( select department_id from departments d2
where d1.department_id = d2.department_id)

select d1.department_id, (select full_name from departments d2 where d2.department_id = d1.depa
rtment_id) id  from departments d1 where department_id  in ( select department_id from departme
nts d2 where d1.department_id = d2.department_id)
```

The following nested SELECT statements examples are not supported:

```
select d1.department_id, (select full_name from departments d2 where d2.department_id = d1.depa
rtment_id) id from (select department_id from departments) d1

select (select full_name from departments  d1 where d1.department_id = d2.department_id) d from
 departments d2

select d1.department_id, (select full_name from departments d2 where d2.department_id = d1.depa
rtment_id) id  from (select department_id from departments) d1     where department_id  in ( se
lect department_id from departments d2 where d1.department_id = d2.department_id)
```

Interval Types Not Supported

Connect drivers do not support interval types. You can use the following alternatives for interval types.

- **Interval Day** that is applied directly to TIMESTAMP or DATE types can be replaced with a positive or negative integer.

  For example, replace **[current_date - interval '7' Day]** with **[current_date - 7]**.
- Intervals for values smaller than one day can be replaced in a similar manner by converting them to fractions of a day. However, current_date does not support HH:MM:SS, so SYSDATE with the TIMESTAMP type is preferable.

  For example, replace **[current_date - interval '13' Hour]** with **[sysdate - 13/24]** or **[sysdate - 1/1.84615]**.
- For values larger than one day, the function Add_Months( ) can be used. Also, the numeric parameter can be multiplied by 12 for years.

For example, **[current_date + interval '2' Year]** can be replaced with **[Add_Months(current_date, 12*2)]**.

CAST and CONVERT Function

The CAST and CONVERT functions convert a value from one data type to another. The Connect driver supports the CAST and CONVERT functionality by providing alternatives to employing character, number, and date conversions using the following methods.

Character Conversion

For example, you to convert a VARCHAR type, you could use this as an alternative to the CAST and CONVERT function.

```
select to_char(account_id) from REVRECSCHEDULELINES
```

Numeric Conversion

For a NUMBER type, you could use to_number for conversion in to a number example as an alternative to the CAST and CONVERT function.

```
select to_number('10'+amount) from REVRECSCHEDULELINES
```

Date Conversion

For a DATE conversion, you could use to_date for conversion in to date.

```
select to_date('01-01-2013', 'dd-mm-yyyy') from REVRECSCHEDULELINES
```

Reserved Words

Reserved words include the following:

- RANK
- ROW_NUMBER
- STDDEV
- STDDEV_POP
- STDDEV_SAMP
- VAR_POP
- VAR_SAMP

These reserved words must be quoted in queries, to avoid parsing errors. For example:

```
select "RANK" from TABLE
```

Note: There are no standard NetSuite records or columns that conflict with the new reserved words. You will encounter this issue only if you have a custom record or custom column exposed through the Connect Service with a name that matches a new reserved word. Case does not matter and the conflict also occurs when the underscore is replaced with a space. For example, custom columns named both var samp (with a space) and var_samp (with and underscore) would both conflict with a new reserved word and would need to be quoted in queries.

The following list of reserve words, if used for a SuiteAnalytics custom table or field, will append suffixes to appear. If you attempt to name a SuiteAnalytics custom field a reserve word, the field will appear as "reserveword_0". If you attempt to name a custom table a reserve word, the table will appear as "reserveword_0". This is a list of reserve words where these rules apply.

- ABORT
- ACCESS
- ACCESSED
- ACCOUNT
- ACTIVATE
- ADD
- ADMIN
- ADMINISTER
- ADMINISTRATOR
- ADVISE
- ADVISOR
- AFTER
- ALIAS
- ALL
- ALL_ROWS
- ALLOCATE
- ALLOW
- ALTER
- ALWAYS
- ANALYZE
- ANCILLARY
- AND
- AND_EQUAL
- ANTIJOIN
- ANY
- APPEND
- APPLY
- ARCHIVE
- ARCHIVELOG
- ARRAY
- AS

- ASC
- ASSOCIATE
- AT
- ATTRIBUTE
- ATTRIBUTES
- AUDIT
- AUTHENTICATED
- AUTHENTICATION
- AUTHENTICATION
- AUTHID
- AUTHORIZATION
- AUTO
- AUTOALLOCATE
- AUTOEXTEND
- AUTOMATIC
- AVAILABILITY
- AVG
- BACKUP
- BATCH
- BECOME
- BEFORE
- BEGIN
- BEGIN_OUTLINE_DATA
- BEHALF
- BETWEEN
- BFILE
- BIGFILE
- BINARY_DOUBLE
- BINARY_DOUBLE_INFINITY
- BINARY_DOUBLE_NAN
- BINARY_FLOAT
- BINARY_FLOAT_INFINITY
- BINARY_FLOAT_NAN
- BINDING
- BITMAP
- BITMAP_TREE
- BITMAPS
- BITS
- BLOB
- BLOCK

- BLOCK_RANGE
- BLOCKS
- BLOCKSIZE
- BODY
- BOTH
- BOUND
- BROADCAST
- BUFFER
- BUFFER_CACHE
- BUFFER_POOL
- BUILD
- BULK
- BY
- BYPASS_RECURSIVE_CHECK
- BYPASS_UJVC
- BYTE
- CACHE
- CACHE_CB
- CACHE_INSTANCES
- CACHE_TEMP_TABLE
- CALL
- CANCEL
- CARDINALITY
- CASCADE
- CASE
- CATEGORY
- CAST
- CERTIFICATE
- CFILE
- CHAINED
- CHANGE
- CHAR
- CHAR_CS
- CHARACTER
- CHECK
- CHECKPOINT
- CHILD
- CHOOSE
- CHUNK
- CIV_GB

- CLASS
- CLEAR
- CLOB
- CLONE
- CLOSE
- CLOSE_CACHED_OPEN_CURSORS
- CLUSTER
- CLUSTERING_FACTOR
- COALESCE
- COARSE
- COLLECT
- COLUMN
- COLUMN_STATS
- COLUMN_VALUE
- COLUMNS
- COMMENT
- COMMIT
- COMMITTED
- COMPACT
- COMPATIBILITY
- COMPILE
- COMPLETE
- COMPOSITE_LIMIT
- COMPRESS
- COMPUTE
- CONFORMING
- CONNECT
- CONNECT_BY_COMBINE_SW
- CONNECT_BY_COST_BASED
- CONNECT_BY_FILTERING
- CONNECT_BY_ISCYCLE
- CONNECT_BY_ISLEAF
- CONNECT_BY_ROOT
- CONNECT_TIME
- CONSIDER
- CONTAINER
- CONTINUE
- CONSISTENT
- CONSTRAINT
- CONSTRAINTS

- CONTENT
- CONTENTS
- CONTEXT
- CONTROLFILE
- CONVERT
- CORRUPTION
- COST
- COUNT
- CPU_COSTING
- CPU_PER_CALL
- CPU_PER_SESSION
- CREATE
- CREATE_STORED_OUTLINES
- CROSS
- CUBE
- CUBE_GB OWN
- CURRENT_DATE
- CURRENT_SCHEMA
- CURRENT_TIME
- CURRENT_TIMESTAMP
- CURRENT_USER
- CURSOR
- CURSOR_SHARING_EXACT
- CURSOR_SPECIFIC_SEGMENT
- CYCLE
- DANGLING
- DATA
- DATABASE
- DATAFILE
- DATAFILES
- DATAOBJNO
- DATE
- DATE_MODE
- DAY
- DB_ROLE_CHANGE
- DBA
- DBA_RECYCLEBIN
- DBMS_STATS
- DBTIMEZONE
- DDL

- DEALLOCATE
- DEBUG
- DEC
- DECIMAL
- DECLARE
- DECREMENT
- DECRYPT
- DEFAULT
- DEFERRABLE
- DEFERRED
- DEFINED
- DEFINER
- DEGREE
- DELAY
- DELETE
- DEMAND
- DENSE_RANK
- DEQUEUE
- DEREF
- DEREF_NO_REWRITE
- DESC
- DETACHED
- DETERMINES
- DICTIONARY
- DIMENSION
- DIRECTORY
- DISABLE
- DISABLE_RPKE EXTRACT
- DISASSOCIATE
- DISCONNECT
- DISK
- DISKGROUP
- DISKS
- DISMOUNT
- DISTINCT
- DISTINGUISHED
- DISTRIBUTED
- DML
- DML_UPDATE
- DOCUMENT

- DOMAIN_INDEX_NO_SORT
- DOMAIN_INDEX_SORT
- DOUBLE
- DOWNGRADE
- DRIVING_SITE
- DROP
- DUMP
- DYNAMIC
- DYNAMIC_SAMPLING
- DYNAMIC_SAMPLING_EST_CDN
- EACH
- ELEMENT
- ELIMINATE_JOIN
- ELIMINATE_OBY
- ELIMINATE_OUTER_JOIN
- ELSE
- EMPTY
- ENABLE
- ENCRYPT
- ENCRYPTION
- END
- END_OUTLINE_DATA
- ENFORCE
- ENFORCED
- ENQUEUE
- ENTERPRISE
- ENTRY
- ERROR
- ERROR_ON_OVERLAP_TIME
- ERRORS
- ESCAPE
- ESTIMATE
- EVALNAME
- EVALUATION
- EVENTS
- EXCEPT
- EXCEPTIONS
- EXCHANGE
- EXCLUDING
- EXCLUSIVE

- EXECUTE
- EXEMPT
- EXISTS
- EXPAND_GSET_TO_UNION
- EXPIRE
- EXPLAIN
- EXPLOSION
- EXPORT
- EXPR_CORR_CHECK
- EXTENDS
- EXTENT
- EXTENTS
- EXTERNAL
- EXTERNALLY
- FACT
- FAILED
- FAILED_LOGIN_ATTEMPTS
- FAILGROUP
- FALSE
- FAST
- FBTSCAN
- FFS
- FIC_CIV
- FIC_PIV
- FILE
- FILTER
- FINAL
- FINE
- FINISH
- FIRST
- FIRST_ROWS
- FLAGGER
- FLASHBACK
- FLOAT
- FLOB
- FLUSH
- FOLLOWING
- FOR
- FORCE
- FORCE_XML_QUERY_REWRITE

- FOREIGN
- FREELIST
- FREELISTS
- FREEPOOLS
- FRESH
- FROM
- FULL
- FUNCTION
- FUNCTIONS
- G
- GATHER_PLAN_STATISTICS
- GBY_CONC_ROLLUP
- GENERATED
- GLOBAL
- GLOBAL_NAME
- GLOBAL_TOPIC_ENABLED
- GLOBALLY
- GRANT
- GROUP_BY
- GROUPING
- GROUP
- GROUPS
- GUARANTEE
- GUARANTEED
- GUARD
- HASH
- HASH_AJ
- HASH_SJ
- HASHKEYS
- HAVING
- HEADER
- HEAP
- HIERARCHY
- HIGH
- HINTSET_BEGIN
- HINTSET_END
- HOUR CURRENT
- HWM_BROKERED
- ID
- IDENTIFIED

- IDENTIFIER
- IDENTITY
- IDGENERATORS
- IDLE_TIME
- IF
- IGNORE
- IGNORE_OPTIM_EMBEDDED_HINTS
- IGNORE_WHERE_CLAUSE
- IMMEDIATE
- IMPORT
- IN
- IN_MEMORY_METADATA
- INCLUDE_VERSION
- INCLUDING
- INCREMENT
- INCREMENTAL
- INDEX
- INDEX_ASC
- INDEX_COMBINE
- INDEX_DESC
- INDEX_FFS
- INDEX_FILTER
- INDEX_JOIN
- INDEX_ROWS
- INDEX_RRS
- INDEX_RS
- INDEX_RS_ASC
- INDEX_RS_DESC
- INDEX_SCAN
- INDEX_SKIP_SCAN
- INDEX_SS
- INDEX_SS_ASC
- INDEX_SS_DESC
- INDEX_STATS
- INDEXED
- INDEXES
- INDEXTYPE
- INDEXTYPES
- INDICATOR
- INFINITE

- INFORMATIONAL
- INITIAL
- INITRANS
- INLINE
- INNER
- INSTANCE
- INSTANTLY
- INTERMEDIATE
- INTERNAL_CONVERT
- INTERNAL_USE
- INITIALIZED
- INITIALLY
- INLINE_XMLTYPE_NT
- INSERT
- INSTANCES
- INSTANTIABLE
- INSTEAD
- INT
- INTEGER
- INTERPRETED
- INTERSECT
- INTERVAL
- INTO
- INVALIDATE
- IS
- ISOLATION
- ISOLATION_LEVEL
- ITERATE
- ITERATION_NUMBER
- JAVA
- JOB
- JOIN
- K
- KEEP
- KERBEROS
- KEY
- KEY_LENGTH
- KEYS
- KEYSIZE
- KILL

- LAST
- LATERAL
- LAYER
- LDAP_REG_SYNC_INTERVAL
- LDAP_REGISTRATION
- LDAP_REGISTRATION_ENABLED
- LEADING
- LEFT
- LENGTH
- LESS
- LEVEL
- LEVELS
- LIBRARY
- LIKE
- LIKE2
- LIKE4
- LIKEC
- LIKE_EXPAND
- LIMIT
- LINK
- LIST
- LOB
- LOCAL
- LOCAL_INDEXES
- LOCALTIME
- LOCALTIMESTAMP
- LOCATION
- LOCATOR
- LOCK
- LOCKED
- LOG
- LOGFILE
- LOGGING
- LOGICAL
- LOGICAL_READS_PER_CALL
- LOGICAL_READS_PER_SESSION
- LOGOFF
- LOGON
- LONG
- MAIN

- MANAGE
- MANAGED
- MANAGEMENT
- MANUAL
- MAPPING
- MASTER
- MATCHED
- MATERIALIZE
- MATERIALIZED
- MAX
- MAXARCHLOGS
- MAXDATAFILES
- MAXEXTENTS
- MAXINSTANCES
- MAXIMIZE
- MAXLOGFILES
- MAXLOGHISTORY
- MAXLOGMEMBERS
- MAXSIZE
- MAXTRANS
- MAXVALUE
- MEASURES
- MEMBER
- MEMORY
- MERGE
- MERGE_AJ
- MERGE_CONST_ON
- MERGE_SJ
- METHOD
- MIGRATE
- MIN
- MINEXTENTS
- MINIMIZE
- MINIMUM
- MINUS
- MINUS_NULL
- MINUTE
- MINVALUE
- MIRROR
- MLSLABEL

- MODE
- MODEL
- MODEL_COMPILE_SUBQUERY
- MODEL_DONTVERIFY_UNIQUENESS
- MODEL_DYNAMIC_SUBQUERY
- MODEL_MIN_ANALYSIS
- MODEL_NO_ANALYSIS
- MODEL_PBY
- MODEL_PUSH_REF
- MODIFY
- MONITORING
- MONTH
- MOUNT
- MOVE
- MOVEMENT
- MULTISET
- MV_MERGE
- NAMED
- NAN
- NATIONAL
- NATIVE
- NATIVE_FULL_OUTER_JOIN
- NATURAL
- NAV
- NESTED_TABLE_FAST_INSERT
- NESTED_TABLE_SET_REFS
- NCHAR
- NCHAR_CS
- NCLOB
- NEEDED
- NESTED
- NESTED_TABLE_GET_REFS
- NESTED_TABLE_ID
- NESTED_TABLE_SET_SETID
- NETWORK
- NEVER
- NEW
- NEXT
- NL_AJ
- NL_SJ

- NLS_CALENDAR
- NLS_CHARACTERSET
- NLS_COMP
- NO_CONNECT_BY_FILTERING
- NLS_CURRENCY
- NLS_DATE_FORMAT
- NLS_DATE_LANGUAGE
- NLS_ISO_CURRENCY
- NLS_LANG
- NLS_LANGUAGE
- NLS_LENGTH_SEMANTICS
- NLS_NCHAR_CONV_EXCP
- NLS_NUMERIC_CHARACTERS
- NLS_SORT
- NLS_SPECIAL_CHARS
- NLS_TERRITORY
- NO
- NO_BASETABLE_MULTIMV_REWRITE
- NO_BUFFER
- NO_CARTESIAN
- NO_CONNECT_BY_COMBINE_SW
- NO_CONNECT_BY_COST_BASED
- NO_CPU_COSTING
- NO_ELIMINATE_JOIN
- NO_ELIMINATE_OBY
- NO_ELIMINATE_OUTER_JOIN
- NO_EXPAND
- NO_EXPAND_GSET_TO_UNION
- NO_FACT
- NO_FILTERING
- NO_INDEX
- NO_INDEX_
- NO_INDEX_RS
- NO_INDEX_SS
- NO_MERGE
- NO_MODEL_PUSH_REF
- NO_MONITORING
- NO_MULTIMV_REWRITE
- NO_NATIVE_FULL_OUTER_JOIN
- NO_ORDER_ROLLUPS

- NO_PARALLEL
- NO_PARALLEL_INDEX
- NO_PARTIAL_COMMIT
- NO_PULL_PRED
- NO_PUSH_PRED
- NO_PUSH_SUBQ
- NO_PRUNE_GSETS
- NO_PX_JOIN_FILTER
- NO_QKN_BUFF
- NO_QUERY_TRANSFORMATION
- NO_REF_CASCADE
- NO_REWRITE
- NO_SEMIJOIN
- NO_SET_TO_JOIN
- NO_SQL_TUNE
- NO_STAR_TRANSFORMATION
- NO_STATS_GSETS
- NO_SWAP_JOIN_INPUTS
- NO_TEMP_TABLE
- NO_UNNEST
- NO_USE_HASH
- NO_USE_HASH_AGGREGATION
- NO_USE_MERGE
- NO_USE_NL
- NO_XML_DML_REWRITE
- NO_XML_QUERY_REWRITE
- NO_ACCESS
- NOAPPEND
- NOARCHIVELOG
- NOAUDIT
- NOCACHE
- NOCOMPRESS
- NOCPU_COSTING
- NOCYCLE
- NODELAY
- NOFORCE
- NOGUARANTEE
- NOMAPPING
- NOMAXVALUE
- NOMINVALUE

- NONE
- NOPARALLEL_INDEX
- NORELY
- NOREPAIR
- NORESETLOGS
- NOREVERSE
- NOREWRITE
- NOROWDEPENDENCIES
- NOSEGMENT
- NOSWITCH
- NOT
- NOTIFICATION
- NOVALIDATE
- NOLOGGING
- NOMINIMIZE
- NOMONITORING
- NOORDER
- NOOVERRIDE
- NOPARALLEL
- NORMAL
- NOSORT
- NOSTRICT
- NOTHING
- NOWAIT
- NULL
- NULLS
- NUM_INDEX_KEYS
- NUMBER
- NUMERIC
- NVARCHAR2
- OBJECT
- OBJNO
- OBJNO_REUSE
- OF
- OFF
- OFFLINE
- OID
- OIDINDEX
- OLD
- OLD_PUSH_PRED

- ON
- ONLINE
- ONLY
- OPAQUE
- OPAQUE_
- OPAQUE_TRANSFORM
- OPCODE
- OPEN
- OPERATOR
- OPT_ESTIMATE
- OPT_PARAM
- OPTIMAL
- OPTIMIZER_FEATURES_ENABLE
- OPTIMIZER_GOAL
- OPTION
- OR
- OR_EXPAND
- ORA_ROWSCN
- ORDER
- ORDERED
- ORDERED_PREDICATES
- ORDINALITY
- ORGANIZATION
- OUT_OF_LINE
- OUTER
- OUTLINE
- OUTLINE_LEAF
- OVER
- OVERFLOW
- OVERFLOW_NOMOVE
- OVERLAPS
- P
- PACKAGE
- PACKAGES
- PARALLEL
- PARALLEL_INDEX
- PARAM
- PARAMETERS
- PARENT
- PARITY

- PARTIALLY
- PARTITION
- PARTITIONS
- PARTITION_HASH
- PARTITION_LIST
- PARTITION_RANGE
- PASSING
- PASSWORD
- PASSWORD_GRACE_TIME
- PASSWORD_LIFE_TIME
- PASSWORD_LOCK_TIME
- PASSWORD_VERIFY_FUNCTION
- PASSWORD_REUSE_MAX
- PASSWORD_REUSE_TIME
- PATH
- PATHS
- PCTINCREASE
- PCTFREE
- PCTTHRESHOLD
- PCTUSED
- PCTVERSION
- PERCENT
- PERFORMANCE
- PERMANENT
- PFILE
- PHYSICAL
- PIV_GB
- PIV_SSF
- PLAN
- PLSQL_CCFLAGS
- PLSQL_CODE_TYPE
- PLSQL_DEBUG
- PLSQL_OPTIMIZE_LEVEL
- PLSQL_WARNINGS
- POINT
- POLICY
- POST_TRANSACTION
- POWER
- PQ_DISTRIBUTE
- PQ_MAP

- PQ_NOMAP
- PREBUILT
- PRECEDING
- PRECISION
- PRECOMPUTE_SUBQUERY
- PREPARE
- PRESENT
- PRESERVE
- PRESERVE_OID
- PRIMARY
- PRIOR
- PRIVATE
- PRIVATE_SGA
- PRIVILEGE
- PRIVILEGES
- PROCEDURE
- PROFILE
- PROGRAM
- PROJECT
- PROTECTED
- PROTECTION
- PUBLIC
- PULL_PRED
- PURGE
- PUSH_PRED
- PUSH_SUBQ
- PX_GRANULE
- PX_JOIN_FILTER
- QB_NAME
- QUERY
- QUERY_BLOCK
- QUEUE
- QUEUE_CURR
- QUEUE_ROWP
- QUIESCE
- QUOTA
- RANDOM
- RANGE
- RAPIDLY
- RAW

- RBA
- RBO_OUTLINE
- READ
- READS
- REAL
- REBALANCE
- REBUILD
- RECORDS_PER_BLOCK
- RECOVER
- RECOVERABLE
- RECOVERY
- RECYCLE
- RECYCLEBIN
- REDUCED
- REDUNDANCY
- REF
- REF_CASCADE_CURSOR
- REFERENCE
- REFERENCED
- REFERENCES
- REFERENCING
- REFRESH
- REGEXP_LIKE
- REGISTER
- REJECT
- REKEY
- RELATIONAL
- RELY
- REMOTE_MAPPED
- RENAME
- REPAIR
- REPLACE
- REQUIRED
- RESET
- RESETLOGS
- RESIZE
- RESOLVE
- RESOLVER
- RESOURCE
- RESTORE

- RESTORE_AS_INTERVALS
- RESTRICT
- RESTRICT_ALL_REF_CONS
- RESTRICTED
- RESUMABLE
- RESUME
- RETENTION
- RETURN
- RETURNING
- REUSE
- REVERSE
- REVOKE
- REWRITE
- REWRITE_OR_ERROR
- RIGHT
- ROLE
- ROLES
- ROLLBACK
- ROLLING
- ROLLUP
- ROW
- ROW_LENGTH
- ROWDEPENDENCIES
- ROWID
- ROWNUM
- ROWS
- RULE
- RULES
- SALT
- SAMPLE
- SAVE_AS_INTERVALS
- SAVEPOINT
- SB4
- SCALE
- SCALE_ROWS
- SCAN
- SCAN_INSTANCES
- SCHEDULER
- SCHEMA
- SCN

- SCN_ASCENDING
- SCOPE
- SD_ALL
- SD_INHIBIT
- SD_SHOW
- SECOND
- SECURITY
- SEED
- SEG_BLOCK
- SEG_FILE
- SEGMENT
- SELECT
- SELECTIVITY
- SEMIJOIN
- SEMIJOIN_DRIVER
- SET
- SEQUENCE
- SEQUENCED
- SEQUENTIAL
- SERIALIZABLE
- SERVERERROR
- SESSION
- SESSIONTZNAME
- SESSION_CACHED_CURSORS
- SESSIONS_PER_USER
- SESSIONTIMEZONE
- SET_TO_JOIN
- SETS
- SETTINGS
- SEVERE
- SIBLINGS
- SIZE
- SHARE
- SHARED
- SHARED_POOL
- SHRINK
- SHUTDOWN
- SID
- SIMPLE
- SINGLE
- SINGLETASK

- SKIP
- SKIP_EXT_OPTIMIZER
- SKIP_UNQ_UNUSABLE_IDX
- SKIP_UNUSABLE_INDEXES
- SMALLFILE
- SMALLINT
- SNAPSHOT
- SOME
- SORT
- SOURCE
- SPACE
- SPECIFICATION
- SPFILE
- SPLIT
- SPREADSHEET
- SQL
- SQL_TRACE
- STANDALONE
- STANDBY
- STAR
- STAR_TRANSFORMATION
- STARTUP
- STATEMENT_ID
- STATIC
- STATISTICS
- STORAGE
- STORE
- STREAMS
- STRICT
- STRING
- STRIP
- SUBMULTISET
- SUBPARTITIONS
- SUM
- START
- STOP
- STRUCTURE
- SUBPARTITION
- SUBPARTITION_REL
- SUBQUERIES
- SUBSTITUTABLE

- SUCCESSFUL
- SUMMARY
- SUPPLEMENTAL
- SUSPEND
- SWAP_JOIN_INPUTS
- SWITCH
- SWITCHOVER
- SYNONYM
- SYSAUX
- SYSDATE
- SYSDBA
- SYS_DL_CURSOR
- SYS_FBT_INSDEL
- SYS_OP_BITVEC
- SYS_OP_CAST
- SYS_OP_EXTRACT
- SYS_OP_ENFORCE_NOT_NULL$
- SYS_OP_NOEXPAND
- SYS_OP_NTCIMG$
- SYS_PARALLEL_TXN
- SYS_RID_ORDER
- SYSOPER
- SYSTEM
- SYSTIMESTAMP
- T
- TABLE
- TABLE_STATS
- TABLES
- TABLESPACE
- TABLESPACE_NO
- TABNO
- TEMP_TABLE
- TEMPFILE
- TEMPLATE
- TEMPORARY
- TEST
- THAN
- THE
- THEN
- THREAD
- THROUGH

- TIME
- TIME_ZONE
- TIMEOUT
- TIMESTAMP
- TIMEZONE_ABBR
- TIMEZONE_HOUR
- TIMEZONE_MINUTE
- TIMEZONE_OFFSET
- TIMEZONE_REGION
- TIV_GB
- TIV_SSF
- TO
- TO_CHAR
- TOPLEVEL
- TRACE
- TRACING
- TRACKING
- TRAILING
- TRANSACTION
- TRANSITIONAL
- TREAT
- TRIGGER
- TRIGGERS
- TRUE
- TRUNCATE
- TRUSTED
- TX
- TUNING
- TYPE
- TYPES
- TZ_OFFSET
- U
- UB2
- UBA
- UID
- UNARCHIVED
- UNBOUND
- UNBOUNDED
- UNDER
- UNDO
- UNDROP

- UNIFORM
- UNION
- UNIQUE
- UNLIMITED
- UNLOCK
- UNNEST
- UNQUIESCE
- UNPACKED
- UNPROTECTED
- UNRECOVERABLE
- UNTIL
- UNUSABLE
- UNUSED
- UPD_INDEXES
- UPD_JOININDEX
- UPDATABLE
- UPDATE
- UPDATED
- UPGRADE
- UPSERT
- UROWID
- USAGE
- USE
- USE_ANTI
- USE_CONCAT
- USE_HASH
- USE_HASH_AGGREGATION
- USE_MERGE
- USE_MERGE_CARTESIAN
- USE_NL
- USE_NL_WITH_INDEX
- USE_PRIVATE_OUTLINES
- USE_SEMI
- USE_STORED_OUTLINES
- USE_TTT_FOR_GSETS
- USE_WEAK_NAME_RESL
- USER
- USER_DEFINED
- USER_RECYCLEBIN
- USERS
- USING

- VALIDATE
- VALIDATION
- VALUE
- VALUES
- VARCHAR
- VARCHAR2
- VARRAY
- VARYING
- VECTOR_READ
- VECTOR_READ_TRACE
- VERSION
- VERSIONS
- VIEW
- WAIT
- WALLET
- WELLFORMED
- WHEN
- WHENEVER
- WHERE
- WHITESPACE
- WITH
- WITHIN
- WITHOUT
- WORK
- WRAPPED
- WRITE
- X_DYN_PRUNE
- XCANONICAL
- XID
- XML_DML_RWT_STMT
- XMLATTRIBUTES
- XMLCOLATTVAL
- XMLELEMENT
- XMLFOREST
- XMLNAMESPACES
- XMLPARSE
- XMLPI
- XMLQUERY
- XMLROOT
- XMLSCHEMA
- XMLSERIALIZE

- XMLTABLE
- XMLTYPE
- YEAR
- YES
- ZONE

New Connections

To conserve system resources, you should close unused connections and maintain the number of open connections to a minimum. If a connection fails with a Connect Timeout Error or the Connection Reset By Peer Error, you should retry to obtain a new connection.

Important: SuiteAnalytics Connect allows for multiple concurrent sessions, however if an account makes hundreds or thousands of connections within minutes, it is flagged. Access for these accounts can also be temporarily suspended to prevent overloading hardware resources, which can affect other Oracle | NetSuite customers.

Exceptions

New connection and query execution requests may occasionally fail during moments of peak usage. Use an exception handling mechanism to automatically re-run such operations.

Replication of Tables

The following are some techniques for replicating of large data for when using SELECT * FROM TABLE statement leads to an error.

- Return columns subset. Use only the necessary subset of return columns. For example, instead of:

  ```
  SELECT
  * FROM TRANSACTIONS
  ```

  use:

  ```
  SELECT
  trandate, trandoc, total_amount FROM TRANSACTIONS
  ```

  Using * column selector is very demanding on resources.
- Slicing. Slice one page result set by a condition in where clause that helps to divide the replicated data into blocks with predicted size. This requires a cycle. For example, instead of:

  ```
  SELECT * FROM TRANSACTIONS
  ```

  use:

  ```
  select * from TRANSACTIONS where transaction_id >
  0 and transaction_id <=
  ```

```
1000000

select * from TRANSACTIONS where transaction_id >
1000000 and transaction_id <=
2000000

select * from TRANSACTIONS where transaction_id
2000000 and transaction_id <=
3000000

...
```

- List of IDs and replicate. Fetch a list of IDs and then replicate by batches then iterate over the returned list and retrieve one transaction at a time. For example, use

```
SELECT TRANSACTION_ID FROM TRANSACTIONS
```

then iterate over all TRANSACTION_ID list

```
SELECT trandate, trandoc, total_amount FROM TRANSACTIONS where TRANSACTION_ID =
<xxx>
```

- Combination of Return column subset and Slicing. For example,

```
select trandate, trandoc, total_amount from TRANSACTIONS where transaction_id >0 and transactio
n_id <= 1000000

select trandate, trandoc, total_amount from TRANSACTIONS where transaction_id >1000000 and tran
saction_id <= 2000000

select trandate, trandoc, total_amount from TRANSACTIONS where transaction_id >2000000 and tran
saction_id <= 3000000
```

- Combination of Slicing and List of IDs and replicate.

```
SELECT TRANSACTION_ID FROM TRANSACTIONS

SELECT trandate, trandoc, total_amount FROM TRANSACTIONS where TRANSACTION_ID =
<xxx>
```

- Combination of all techniques.

```
select TRANSACTION_ID from TRANSACTIONS where transaction_id >0 and transaction_id <= 1000000

select TRANSACTION_ID from TRANSACTIONS where transaction_id >1000000 and transaction_id <= 200
0000

select TRANSACTION_ID from TRANSACTIONS where transaction_id >2000000 and transaction_id <= 300
0000
```

then iterate over all TRANSACTION_ID list

```
SELECT trandate, trandoc, total_amount FROM TRANSACTIONS where TRANSACTION_ID =
<xxx>
```

Incremental Backups

Use the following points as best practices for incremental backups.

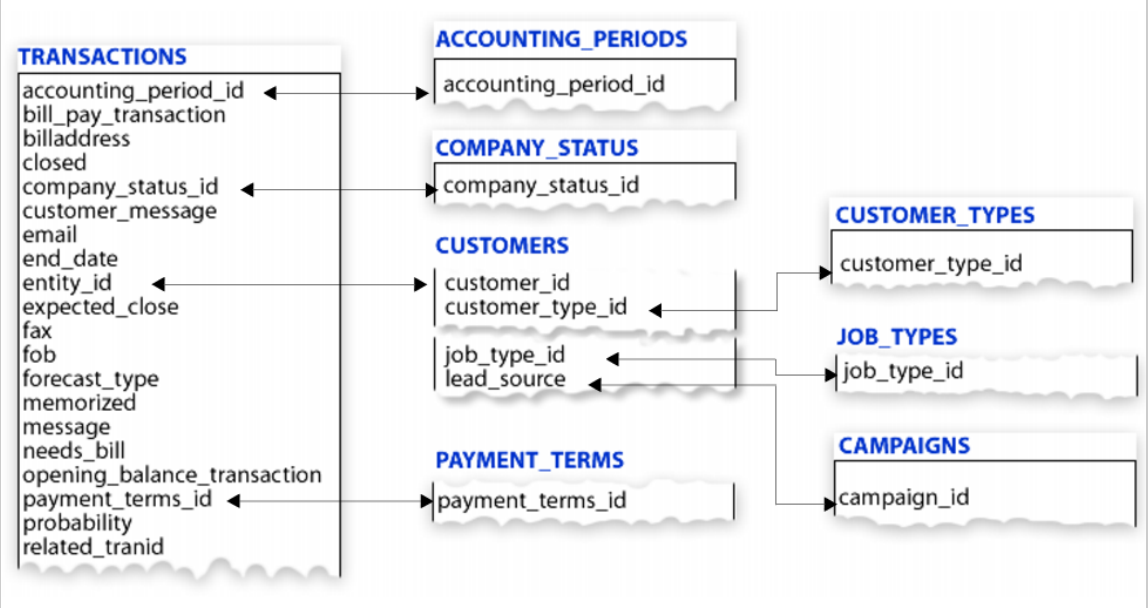
- Use **last_modified_date** or **date_last_modified** columns where possible.
- Use **deleted_records** table where possible.
- When you cannot identify the **last_modified_date**, **date_last_modified**, or **deleted_records**, you should file an issue.
- Some tables do not support **last_modified_date**, **date_last_modified**, and **deleted_records**, for example, mapping tables. You must download all data from these tables every time, it is not possible to perform incremental downloads at this time.

Column Joins in the Connect Service

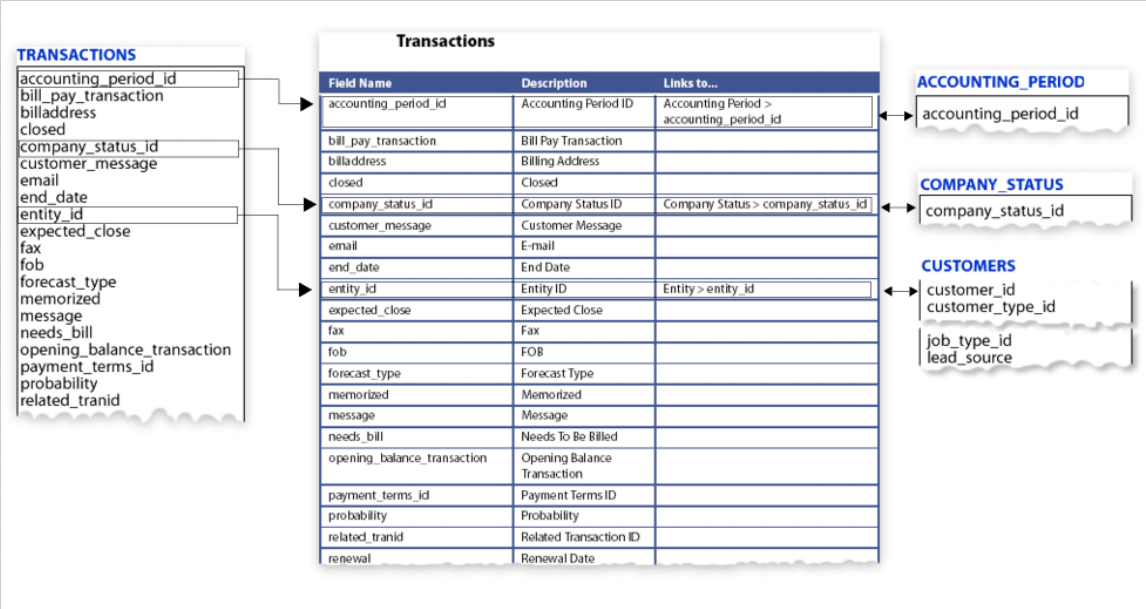
In the following example, the Transactions table contains the **accounting_period_id** column. The arrows show the foreign key between the **accounting_period_id** column in the Transactions table and the corresponding column in the Accounting Periods table.

The Transactions table also contains an **entity_id** column. In NetSuite, entities are defined as companies, contacts, customers, employees, leads, partners, prospects, and vendors, and have records stored in NetSuite. When browsing the schema, the **entity_id** column may be listed for multiple tables. In the case of the Transactions table, this column is linked to the **customer_id** column in the Customers table, but this column can also be linked to other tables, depending on the table in which the column is listed.

The following figure contains portions of eight different tables and columns within the table. The schema can be listed similarly in a reporting tool.



The following graphic shows the Transactions table with a table of its corresponding columns. The table contains three columns: **Column Name, Descriptions,** and **Links to...**. In the Links to... column, if there is a link to a column in another table, the name of the other table and the column linked to within that table appears.



Custom Columns, Lists, and Records in the Connect Service

The following examples show how NetSuite represents customizations in the Connect Service.

- Custom List, Record, or Column Name Conflicts
- Custom Transaction Body Column
- Custom List and Free Form Custom Column
- Custom List and Multiple Select Custom Column
- Custom Record and Custom Free Form Custom Column
- Custom Record and Multiple Select Custom Column
- Custom Field Limitations

For more information on custom record types and custom lists in the SuiteAnalytics Connect schema, see Custom Lists and Custom Record Types.

Custom List, Record, or Column Name Conflicts

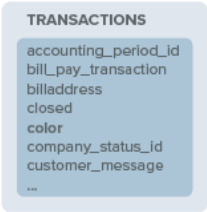
You should not define names for custom lists, records, and columns that are the same as the names of the Connect Service and columns as this will negatively affect querying data via Connect drivers. For example, “events” is an SQL reserved word, so when a table or column is named “event” in the Connect Service schema, it could have a negative result.

It is recommended that you do not define names for custom lists or custom records that duplicate the name of a standard table in the SuiteAnalytics Connect schema. If you do create a custom list or record that has the same name as an existing standard table, the list or record will be exposed to Connect with a suffix of _1, or with a higher number in the case case of multiple duplicates. For a list of standard tables, see Connect Schema.

Custom Transaction Body Column

In this example, an administrator has created a transaction body column called **Color**. The custom column is a free-form text column. This column is applied to Purchase and Sales transaction forms.

In the Connect Service, the custom column is added to the Transactions table.



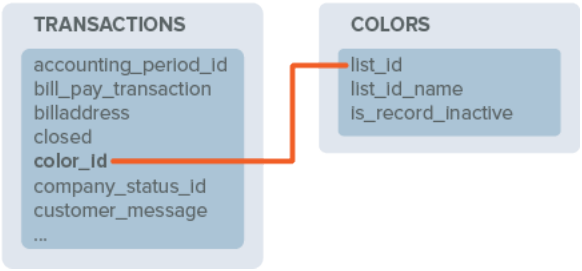
Custom List and Free Form Custom Column

In this example, the administrator has created a custom list called **Colors**. The values included in this list are Blue, Red, Yellow, and Green.

Let's assume when you created the custom **Color** column in example 1, you selected the **Colors** list as the **List/Record** for that column.

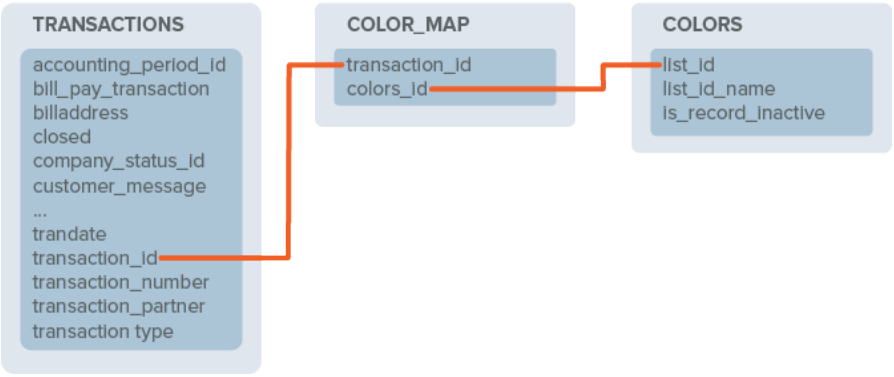
The **color** column shown in the Transactions table in example 1 has been replaced by a **color_id** column and a new table has been created for the **Colors** list.

The **color_id** column in the Transactions table links to the **list_id** column on the Colors table.



Custom List and Multiple Select Custom Column

This example illustrates the changes made to the Connect Service if the custom column, **Color**, is changed from a free-form text column to a multiple select column.



A new table, Color Map, is created and is linked to from the Transactions and Colors tables.

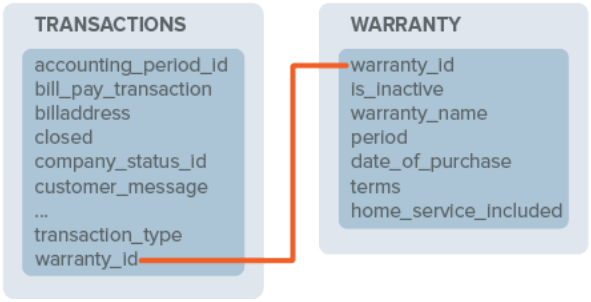
Custom Record and Custom Free Form Custom Column

In this example, an administrator creates a custom record called **Warranty** and adds the following four columns to the custom record:

Column Name	Column Type
Period	Numeric
Date of Purchase	Date
Terms	Free-form Text
Home Service Included	Check box

The administrator also creates a free-form transaction body column called **Warranty** and selects the new custom **Warranty** record as the **List/Record** for this column. This column is applied to Purchase and Sales transaction forms.

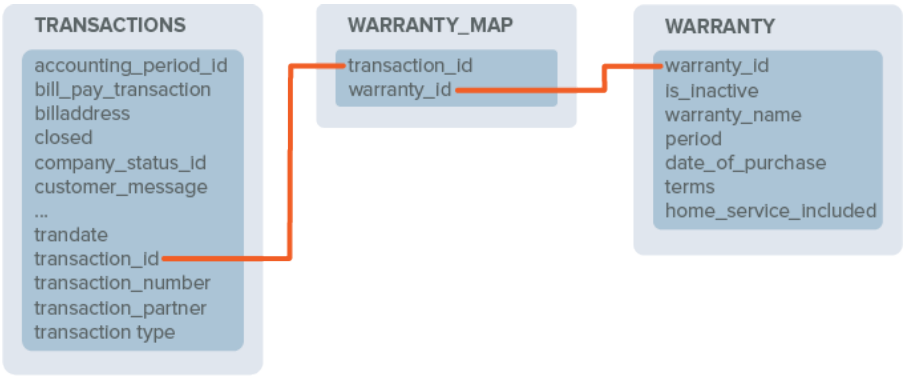
In the following figure, a **warranty_id** column is included in the Transactions table. A new table, named Warranty, represents the custom record. The Warranty table contains a column for each of the custom record columns. It also includes a **warranty_name** column. This column indicates that the **Include Name Column** preference was checked during the initial setup of the custom record.



Custom Record and Multiple Select Custom Column

This example is similar to Custom Record and Custom Free Form Custom Column except the custom **Warranty** column changes from a free-form text column to a multiple select column. The settings for the custom record are not changed.

A new table, Warranty Map, is created and is linked to from the Transactions table and the Warranty table.



Note: Custom column names that contain non-ASCII characters appear without those characters in the SuiteAnalytics Connect schema. If the custom column name contains only non-ASCII characters, the column's field ID is used instead.

Custom Field Limitations

Due to an internal limitation, queries over SuiteAnalytics Connect schema tables that have more than 1000 columns only work if you retrieve 1000 columns or less in the query. For example, if the Transactions table has more than 1000 columns due to the number of custom fields that have been added to the Transaction record type, attempting to query the table using the “Select * From” construct results in the following error: “Error: Could not find any column information for table:transactions”.

This run failure may also occur when joining multiple tables in a single query and trying to retrieve all of their fields.

To query over a table with more than 1000 columns, you must enumerate the specific columns that you want to retrieve or, if you have to use the “Select * From” construct, you must deactivate some of the custom fields that have been added to the table so that there are 1000 columns or less.

Driver Access for a Sandbox or Release Preview Account

After you have installed the NetSuite driver, you can use Windows administrative tools to set up NetSuite driver access to work with your NetSuite Sandbox or Release Preview account instead of your production account.

To configure the NetSuite driver for a Sandbox or Release Preview account, follow the steps outlined in Configuring the ODBC Data Source on Windows to change your Service Host entry from the **Production** account setting to either the **Sandbox** or **Release Preview** account setting.

Troubleshooting SuiteAnalytics Connect Connection

See the following topics for tips on troubleshooting Connect connection issues: Rules for Accessing the Connect Service Using Third-Party Applications

Rules for Accessing the Connect Service Using Third-Party Applications

Use the following rules and guidelines when accessing the Connect Service using third-party applications with ODBC:

- During the development stages of working with the SuiteAnalytics Connect for ODBC driver, you can use Windows administrative tools to set up ODBC access to work with your NetSuite Sandbox or Release Preview account instead of your production account. For more information, see Driver Access for a Sandbox or Release Preview Account.
- Accessing the Connect Service does not respect IP address restriction rules for any of the SuiteAnalytics Connect drivers. Users may be able to access the Connect Service through any of the SuiteAnalytics drivers from IP addresses where they cannot access the NetSuite application directly.

- Logins to SuiteAnalytics Connect are tracked in the Login Audit Trail, available at Setup > Users/Roles > User Management > View Login Audit Trail.
- The order in which column values are returned is arbitrary. NetSuite does not guarantee the return order. In addition, Microsoft® Access only supports 255 columns per result set. If you use Access to get results from the Connect Service that includes more than 255 columns, it is not possible to see all of the table's column values in a single result set.
- Due to an internal limitation, queries over SuiteAnalytics Connect schema tables that have more than 1000 columns only work if you retrieve 1000 columns or less in the query. This run failure may occur when using the "Select * From" construct to query tables that have many custom fields or when joining multiple tables in a single query and trying to retrieve all their fields. For more information, see Custom Field Limitations.

Connect Schema

The following tables describe the Connect Schema available to external reporting tools when you enable the Connect Service. For each Connect table, exists a list of the names of columns included in the table, column descriptions, and a list of keys, if applicable.

The best way to explore the Connect Schema is to view the standard tables and columns in the Connect Browser. To learn more about the Connect Browser, see Working with the SuiteAnalytics Connect Browser.

The SuiteAnalytics Connect feature introduces new terms that differ from the terminology used for the ODBC Connections for the Advanced Reporting feature.

- Views are now referred to as tables.
- Fields are now referred to as columns.
- The entire group of the Connect Service tables are known as the Connect Schema.

The following is a list of tables that make up the Connect Schema.

- Campaign Event
- Campaign Families
- Campaign Item
- Campaign Offer
- Campaign Response
- Campaign Response History
- Campaign Search Engine
- Campaign Subscription Statuses
- Campaign Subscriptions
- Campaign Verticals
- Campaigns
- Case Escalation History
- Case Issue
- Case Origin
- Case Origins
- Case Stage Changes
- Case Type
- Case Types
- Charges
- Classes
- Commission Authorization Link
- Commission Plan
- Commission Rate
- Commission Schedule
- Companies
- Company Contact Map
- Company Status
- Competitor
- Competitor Opportunity Map
- Components Per Routing Steps
- Consolidated Exchange Rates
- Contact Role
- Contact Types
- Contacts
- Countries
- Coupon Codes
- CRM Group
- CRM Group Map

- CRM Template
- Currencies
- Currency Rates
- Custom Lists
- Custom Record Types
- Customer Currencies
- Customer Group Pricing
- Customer Item Pricing
- Customer Partner Sales Teams
- Customer Sales Teams
- Customer Types
- Customers
- Deleted Records
- Departments
- Distribution Categories
- Distribution Networks
- Employee Time
- Employee Types
- Employees
- Entity
- Entity Category
- Entity Event Map
- Entity Role Map
- Entity Status
- Entity Status History
- Entity Territory Map
- Event Attendees
- Events
- Expense Accounts
- Expense Based Charge Rules
- Expense Categories
- Expense Categories Rates
- Expense Detail
- Fair Value Prices
- Fixed Fee Charge Rules
- Generic Resources
- Gift Certificates
- Global Account Map

- Group Test Cell
- Income Accounts
- Inventory Cost Template
- Inventory Cost Template Items
- Inventory Items
- Inventory Number
- tem Account Map
- Item Billing Rates
- Item Demand Plan Lines
- Item Demand Plans
- Item Fulfillments
- Item Group
- Item Location Map
- Item Prices
- Item Quantity
- Item Revisions
- Item Ship Methods
- Item Site Categories
- Item Subsidiary Map
- Item Supply Plan Attributes
- Item Supply Plan Lines
- Item Supply Plan Source
- Item Supply Plan Source Types
- Item Supply Plans
- Item Vendor Map
- Item Vendor Pricing
- Items
- Job Resource Role
- Job Resources
- Job Types
- Locations
- Memorized Transactions
- Message
- Message Recipient
- MFG Cost Template
- MFG Cost Template Items
- MFG Routing
- MFG Routing Steps

- Nexus
- NLCompany
- Note Type
- Notes System
- Notes System Custom
- Notes User
- Opportunities
- Opportunity Contact Map
- Opportunity Lines
- Originating Leads
- Other Names
- Partner Sales Roles
- Partner Types
- Partners
- Payment Methods
- Payment Terms
- Payroll Item Types
- Payroll Items
- Percent Complete Overrides
- Plan Assignment Map
- Plan Schedule Map
- Posting Account Activity
- Price Types
- Pricing Groups
- Project Billing Budgets
- Project Cost Budgets
- Project Cost Categories
- Project Expense Types
- Project Task Assignments
- Project Task Billing Budgets
- Project Task Cost Budgets
- Project Task Dependencies
- Project Tasks
- Project Templates
- Promotion Codes
- Quota
- Resource Allocations
- Revaluation

- Revenue Elements
- Revenue Plan Lines
- Revenue Plans
- Revenue Recognition Rules
- Revenue Recognition Schedule Lines
- Revenue Recognition Schedules
- Role Subsidiary Map
- Roles
- Sales Forecast
- Sales Reps
- Sales Roles
- Sales Territories
- Service Items
- Shipment Packages
- Shipping Items
- Solution
- Solution Case Map
- Solution Topic Map
- States
- Subsidiaries
- Subsidiary Book Map
- Subsidiary Class Map
- Subsidiary Department Map
- Subsidiary Location Map
- Subsidiary Nexus Map
- Supplier Categories
- Support Case History
- Support Incidents (Cases)
- Support Reps
- Support Territories
- Task Contacts
- Tasks
- Tax Items
- Territory
- Time Based Charge Rules
- Topic
- Transaction Address
- Transaction Bin Numbers

- Transaction Book Map
- Transaction Cost Components
- Transaction Inventory Numbers
- Transaction Line Book Map
- Transaction Lines
- Transaction Links
- Transaction Partner Sales Teams
- Transaction Sales Teams
- Transaction Tax Detail
- Transaction Tracking Numbers
- Transactions
- Units Type
- UOM
- Usages
- Vendor Currencies
- Vendor Types
- Vendors
- Win Loss Reason
- Work Calendars

In addition to standard tables, SuiteAnalytics Connect includes system tables. For more information, see SuiteAnalytics Connect System Tables.

Note: Additional schema tables are available only if you have the Advanced Revenue Management feature enabled in your account. For more information on those tables, please refer to the Advanced Revenue Management documentation.

Working with the SuiteAnalytics Connect Browser

The SuiteAnalytics Connect Browser includes a summary of NetSuite data available through SuiteAnalytics Connect. The Browser provides a page for each of the standard tables in the Connect schema. This page lists the table's columns, its primary and foreign keys, and the related tables.

Note: The Connect Browser includes only a subset of the primary and foreign key data in SuiteAnalytics Connect tables.

Important: The data types listed in the Connect Browser follow Oracle naming conventions. Consequently, depending on the SuiteAnalytics Connect driver you use, the data types of certain columns may vary.

For some tables, the Connect Browser also provides domain diagrams that illustrate the relationships among tables in a specific business domain. Currently, only the most widely used domains are included.

The Connect Browser is integrated with the SuiteScript Records Browser and the SuiteTalk Schema Browser, which enables you to compare records type support across SuiteScript, SuiteTalk, and SuiteAnalytics Connect.

To learn how to use the Connect Browser, see the following:

- Finding a Table
- Table Summary
- Domains and Domain Diagrams

Working Offline

If you want to use the SuiteAnalytics Connect Browser when you are working offline, you can download the .zip file that contains the SuiteTalk Schema Browser, SuiteScript Records Browser, and SuiteAnalytics Connect Browser.

After downloading the .zip file, extract it and navigate to the \odbc directory. To view the content, open the index.htm file in the browser of your choice.

Alternatively, you can download the SuiteAnalytics Connect Browser .chm file that contains only the SuiteAnalytics Connect Browser. To download the file, on your NetSuite home page, find the **Settings** portlet and click **Set Up SuiteAnalytics Connect**, then click the **Connect Browser** link.

After you have downloaded the Connect Browser .chm file, you may need to unblock the file to be able to use it.

To unblock the file:

1. Right-click the .chm file and choose **Properties**.
2. In the file properties window, click the **Unblock** button, and then click **Apply**.

When the file is unblocked, you can open it to work with the Connect Browser.

Finding a Table

To find a table in the Connect Browser, use the A-Z index at the top of the browser window.

To find a table:

1. Click the appropriate letter at the top of the browser window.

   The pane at the left updates to include a list of all tables with names that begin with the selected letter. The center pane updates to show details of the first table in the list.

2. Click the name of the table you are interested in.

   The center pane updates to show details of the table.

Table Summary

For each table, the browser displays a series of tables summarizing the following:

- **Columns** – the table's columns.
- **Primary key** – the table's primary key.
- **Foreign keys in this table** – foreign keys in this table that reference columns in other tables.
- **Foreign keys referencing this table** – foreign keys in other tables that reference columns in this table.
- **This table is included in the following domains** – lists the business domains this table is a part of. Domains are currently available only for some tables.
- **Domain diagrams** – if the table is included in a domain, you can use the domain diagram to explore the relationships between all tables included in that domain.

Primary keys use yellow highlighting, whereas foreign keys are highlighted in green. Pink highlighting is applied only to the **date_last_modified** columns, which are used for incremental backups.

Some of the labels used in the Connect Browser are described below.

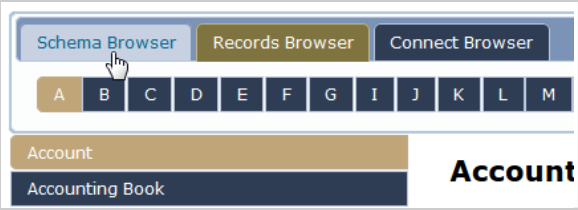
Important: The data types listed in the Connect Browser follow Oracle naming conventions. Consequently, depending on the SuiteAnalytics Connect driver you use, the data types of certain columns may vary.

Label	Description
Name	Column name
Type	Column data type
Length	Maximum field length, in bytes
Precision	Maximum digits in a number or maximum characters in a string
Scale	Maximum decimal places in a number
References	A primary key column in a different table that is referenced by this column
In	The primary key table that is referenced by this column
Description	Column description
PK Column Name	Primary key column name
PK Table Name	Primary key table name
FK Name	Foreign key name
FK Column Name	Foreign key column name
FK Table Name	Foreign key table name
Key Seq	For composite keys, the order of columns in a key
Domains	The domains this table is included in

Comparing Connect, SuiteScript, and SuiteTalk Exposure

You can check whether the table you are viewing in the Connect Browser is supported as a record in SuiteScript or SuiteTalk.

- To check the SuiteScript support, click the Records Browser tab at the top of the page.
- To check the SuiteTalk support, click the Schema Browser tab at the top of the page.



If the record is supported in SuiteTalk or SuiteScript, you are directed to the corresponding page in the SuiteTalk Schema Browser or the SuiteScript Records Browser. Otherwise, you are directed to the first page of the browser.

To compare record type support in SuiteScript, SuiteTalk, and SuiteAnalytics Connect, see .

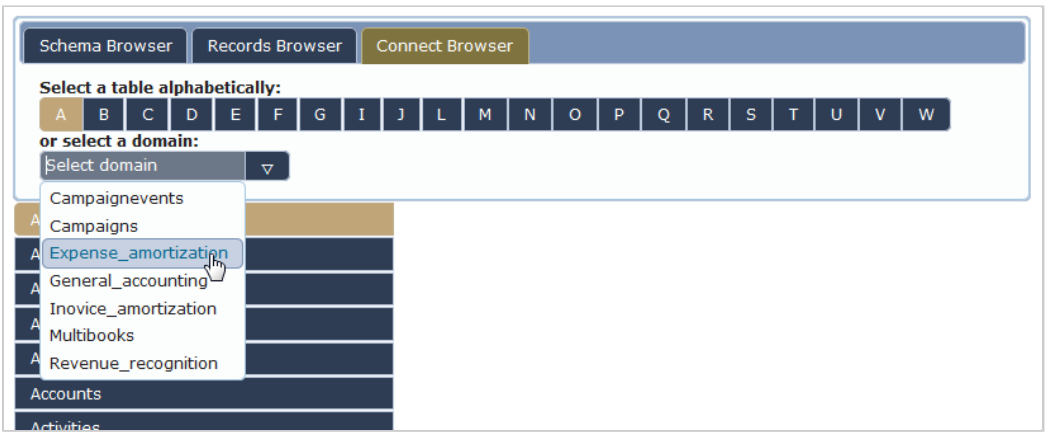
Domains and Domain Diagrams

Some tables in the Connect Browser are combined to form domains. Domains are groups of tables that are related to each other. The relationships between these tables are illustrated by domain diagrams provided for each of the domains.

Currently, the Connect Browser includes only the most widely used domains and their diagrams.

To find a domain:

1. Select the domain from the list at the top of the browser window.



2. A list of all tables included in this domain appears. Click a table to view its summary.

   The diagrams for the domains in which this table is included are displayed at the bottom of the page.

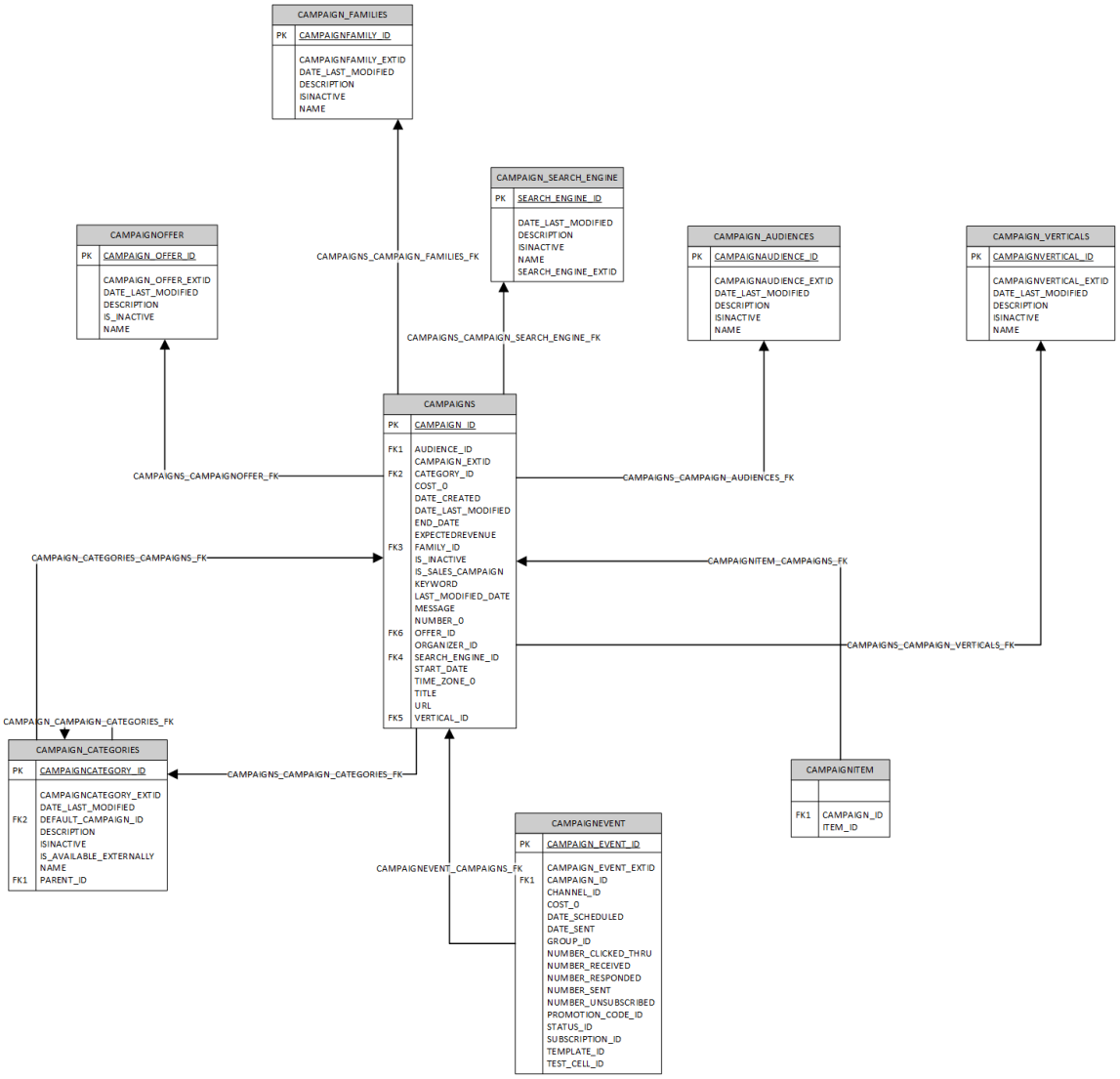
Understanding Domain Diagrams

Domain diagrams outline the relationships between tables in the same domain.

- Primary keys in each table appear at the top of the table and are labeled **PK**.

- Foreign keys are labeled **FK**. If a table contains multiple foreign keys, a number is added to the **FK** label.
- An arrow from one table to another indicates that this table contains a foreign key that is a primary key in the other table.

In the following diagram, the **campaigns** table has one primary key, **campaign_id**, and multiple foreign keys. Through its foreign keys, the **campaigns** table is related to the **campaignoffer**, **campaign_families**, **campaign_search_engine**, **campaign_audiences**, **campaign_verticals**, and **campaign_categories** tables. The tables that reference the **campaigns** table through its primary key are **campaignitem**, **campaign_categories**, and **campaignevent**.



SuiteAnalytics Connect System Tables

In addition to standard and custom NetSuite tables, SuiteAnalytics Connect includes the following system tables:

- oa_tables – lists all available tables, including custom lists and custom records.

- oa_columns – lists all available columns in all available tables, including custom columns and tables.
- oa_fkeys – shows the relations between columns in all available tables.
- **oa_proc** – this table is present in the schema but currently not supported.
- **oa_info** – this table is present in the schema but currently not supported.
- **oa_proccolumns** – this table is present in the schema but currently not supported.
- **oa_statistics** – this table is present in the schema but currently not supported.
- **oa_types** – this table is present in the schema but currently not supported.

These system tables can be used to show all columns and tables available in your account and the relations between them.

Viewing the Tables and Columns Available in Your Account

To find all tables or columns that are available in your account, run either of the following queries in the database client of your choice:

- To see all available tables, run

```
select * from oa_tables;
```

- To see all available columns, run

```
select * from oa_columns;
```

Note: Due to an internal limitation, queries over SuiteAnalytics Connect schema tables that have more than 1000 columns only work if you retrieve 1000 columns or less in the query. This run failure can occur when using the "Select * From" construct to query tables that have many custom fields or when joining multiple tables in a single query and trying to retrieve all their fields. For more information, see Custom Field Limitations.

oa_tables

This is a system table that contains the table name, table owner, table type and descriptions of all tables available in the SuiteAnalytics Connect schema.

Table name: oa_tables	
Column Name	**Description**
table_qualifier	Name of the table qualifier
table_owner	Name of the table owner
table_name	Name of the table
table_type	Table type (NetSuite table, system table)
table_path	not supported
oa_userdata	not supported
oa_support	not supported
remarks	Table description

oa_columns

This is a system table that defines the column name, type, length, and description for all columns in all tables available in the SuiteAnalytics Connect schema.

Table name: oa_columns	
Column Name	**Description**
table_qualifier	Name of the table qualifier
table_owner	Name of the table owner
table_name	Name of the table
column_name	Name of the column
data_type	The column's SQL data type
type_name	Data source dependent data type
oa_length	The length in bytes of data transferred for Fetch, GetData, etc.
oa_precision	The precision of the column on the data source
oa_radix	Radix of data type, NULL for data types where radix does not apply
oa_scale	Scale of the column on the data source
oa_nullable	not supported
oa_scope	not supported
oa_userdata	not supported
oa_support	not supported
pseudo_column	not supported
oa_columntype	not supported
remarks	Column description

Example Queries

- To find all columns in a specific table, use the following query:

```
select * from oa_columns where table_name = 'TABLE_NAME';
```

For example, if you search for all columns in the **departments** table, the output may include the following rows:

table_name	column_name	type_name	oa_length	oa_precision	oa_scale
DEPARTMENTS	DATE_LAST_MODIFIED	TIMESTAMP	0	0	0
DEPARTMENTS	DEPARTMENT_EXTID	VARCHAR2	255	255	
DEPARTMENTS	DEPARTMENT_ID	NUMBER	8	22	0
DEPARTMENTS	FULL_NAME	VARCHAR2	1791	1791	
DEPARTMENTS	ISINACTIVE	VARCHAR2	3	3	
DEPARTMENTS	NAME	VARCHAR2	31	31	

table_name	column_name	type_name	oa_length	oa_precision	oa_scale
DEPARTMENTS	PARENT_ID	NUMBER	8	22	0

- To find all tables that include a specific column, use the following query:

```
select * from oa_columns where column_name = 'COLUMN_NAME';
```

For example, if you search for all tables that include the **subsidiary_id** column, the output result may include the following rows:

table_name	column_name	type_name	oa_length	oa_precision	oa_scale
CUSTOMERS	SUBSIDIARY_ID	NUMBER	8	22	0
SUBSIDIARIES	SUBSIDIARY_ID	NUMBER	8	22	0
EMPLOYEES	SUBSIDIARY_ID	NUMBER	8	22	0
SUBSIDIARY_CLASS_MAP	SUBSIDIARY_ID	NUMBER	8	22	0
SUBSIDIARY_LOCATION_MAP	SUBSIDIARY_ID	NUMBER	8	22	0

- To see the available column descriptions for columns in a specific table, use the following query:

```
select table_name, column_name, remarks from oa_columns where table_name = 'TABLE_NAME' AND rem
arks !='';
```

For example, if you try to find the descriptions for columns in the **vendors** table, the output result may include the following rows:

table_name	column_name	remarks
VENDORS	INDUSTRY_2_ID	Select the industry that best describes the business of the individual or company. You can enter new industry options by selecting new on the list.
VENDORS	NO__OF_EMPLOYEES	The number of employees working for the company.

oa_fkeys

This is a system table that contains information about foreign keys referenced by each table and the primary keys in each table.

A row in the **oa_fkeys** table is considered a primary key if its **pktable_name** and **pkcolumn_name** values are not **NULL**.

Table name: oa_fkeys	
Column Name	**Description**
pktable_qualifier	Primary key table qualifier
pktable_owner	Primary key table owner
pktable_name	Primary key table name
pkcolumn_name	Primary key column name
fktable_qualifier	Foreign key table qualifier
fktable_owner	Foreign key table owner

Table name: oa_fkeys	
Column Name	**Description**
fktable_name	Foreign table name
fkcolumn_name	Foreign key column name
key_seq	The column sequence number in the key, starting with 1
update_rule	not supported
delete_rule	not supported
fk_name	Name of the foreign key
pk_name	Name of the primary key

Example Queries

- To find all tables that reference a specific table, use the following query:

```
select pktable_name, pkcolumn_name, fktable_name, fkcolumn_name, fk_name from oa_fkeys where pk
table_name = 'TABLE_NAME';
```

For example, if you try to find all tables that reference the **accounts** table, the output result may include the following rows:

pktable_name	pkcolumn_name	fktable_name	fkcolumn_name	fk_name
ACCOUNTS	ACCOUNT_ID	EXPENSE_ACCOUNTS	EXPENSE_ACCOUNT_ID	EXPENSE_ACCOUNTS_ACCOUNTS_FK
ACCOUNTS	ACCOUNT_ID	TRANSACTION_LINES	ACCOUNT_ID	TRANSACTION_LINES_ACCOUNTS_FK

- To find all tables that are referenced by a specific table, use the following query:

```
select pktable_name, pkcolumn_name, fktable_name, fkcolumn_name, fk_name from oa_fkeys where fk
table_name = 'TABLE_NAME';
```

For example, if you try to find all tables that are referenced by the **accounts** table, the output result may include the following rows:

pktable_name	pkcolumn_name	fktable_name	fkcolumn_name	fk_name
ACCOUNTS	ACCOUNT_ID	ACCOUNTS	DEFERRAL_ACCOUNT_ID	ACCOUNTS_ACCOUNTS_FK
ACCOUNTS	ACCOUNT_ID	ACCOUNTS	PARENT_ID	ACCOUNTS_ACCOUNTS_FK_2
ACCOUNTS	ACCOUNT_ID	EXPENSE_ACCOUNTS	EXPENSE_ACCOUNT_ID	EXPENSE_ACCOUNTS_ACCOUNTS_FK
ACCOUNTS	ACCOUNT_ID	TRANSACTION_LINES	ACCOUNT_ID	TRANSACTION_LINES_ACCOUNTS_FK

Please note that the table may reference itself.

- To find all tables that contain a specific column as the primary key, use the following query:

```
select pktable_name, pkcolumn_name, key_seq from oa_fkeys where pkcolumn_name = 'COLUMN_NAME';
```

For example, if you try to search for the tables that contain the **location_id** column as the primary key, the output may include the following rows:

pktable_name	pkcolumn_name	key_seq
LOCATIONS	LOCATION_ID	1
SUBSIDIARY_LOCATION_MAP	LOCATION_ID	2

In this example, the **location_id** column is the primary key in both the **locations** and the **subsidiary_location_map** tables. However, the **subsidiary_location_map** table has a composite primary key, consisting of two primary key columns: **subsidiary_id** and **location_id**. The **subsidiary_id** column is the first in the primary key sequence, and the **location_id** column is the second.

- To find all tables that contain a specific column as a foreign key and see which tables include that column as the primary key, use the following query:

```
select fktable_name, fkcolumn_name, pktable_name, pkcolumn_name, key_seq from oa_fkeys where fk
column_name = 'COLUMN_NAME';
```

For example, if you try to find all tables that contain the **location_id** column as a foreign key, the output may include the following rows:

fktable_name	fkcolumn_name	pktable_name	pkcolumn_name	key_seq
ACCOUNTS	LOCATION_ID	LOCATIONS	LOCATION_ID	1
SUBSIDIARY_LOCATION_MAP	LOCATION_ID	LOCATIONS	LOCATION_ID	1

In this example, **location_id** is a foreign key column in the **accounts** and **subsidiary_location_map** tables, whereas in the **locations** table it is the primary key. This means that both the **accounts** and the **subsidiary_location_map** tables are related to the **locations** table through the **location_id** column.

Custom Lists

Data for each custom list in your account is exposed as a table. The name of each table corresponds to the name of the custom list. Each table includes all custom columns and the following additional columns.

Table name: any		
Column Name	**Description**	**Relates to**
date_created	Date Created (GMT)	
last_modified_date	Last Modified Date (GMT)	
is_record_inactive	Value of the Inactive check box. F = the list is active. T = the list is inactive.	
list_id	Unique ID (auto-incremented number)	

Table name: any		
Column Name	**Description**	**Relates to**
list_item_name	This column contains the Name of the list. If translations have been provided for this value, then the column will contain the value translated in to the user's selected locale.	
<ListName>_extid	External ID If a custom list's name starts with a number or an underscore (_), the column name is N_<ListName>_extid.	

Note: Column names are case insensitive and do not contain spaces. Any spaces in your custom column names will be replaced by underscores in the underlying column names.

Custom column names that contain non-ASCII characters appear without those characters in the SuiteAnalytics Connect schema. If the custom column name contains only non-ASCII characters, the column's field ID is used instead.

Identifying primary and foreign keys in custom lists

To find primary and foreign keys in a custom list, you can use the following query, replacing `CUSTOM_LIST_NAME` with your custom list name:

```
select pktable_name, pkcolumn_name, pk_name, fktable_name, fkcolumn_name, fk_name from oa_fkeys
 where pktable_name = 'CUSTOM_LIST_NAME';
```

In the query output, the **pk_name** column contains the primary key name, and the **fk_name** column shows the foreign key name. For more examples of queries over primary and foreign keys, see oa_fkeys.

Custom Record Types

Data for each custom record type in your account is exposed as a table. The name of each table corresponds to the name of the custom record type. Each table includes all custom columns and the following additional columns.

Table name:		
Column Name	**Description**	**Relates to**
date_created	Date Created (GMT)	
last_modified_date	Last Modified Date (GMT)	
is_inactive	Value of the Inactive check box. F = the record type is active. T = the record type is inactive.	
<RecordTypeName>_id	Unique ID (auto-incremented number)	
<RecordTypeName>_extid	External ID If a custom record type's name starts with a number or an underscore (_), the column name is N_<RecordTypeName>_extid.	

Table name:		
Column Name	**Description**	**Relates to**
<RecordTypeName>_name	This column will only be available if you marked the Include Name Column check box. It contains the Name Column value. Note that this column, if selected, is required for all records (see the help topic Creating Custom Record Types)	
<RecordTypeName>_number	This column will only be available if you have specified a numbering format for your custom record type. It contains an automatically generated number, formatted to your specification. (See the help topic Numbering Custom Record Types)	

Note: Column names are case insensitive and do not contain spaces. Any spaces in your custom column names will be replaced by underscores in the underlying column names. <RecordTypeName> is the name of the custom record type.

Custom column names that contain non-ASCII characters appear without those characters in the SuiteAnalytics Connect schema. If the custom column name contains only non-ASCII characters, the column's field ID is used instead.

Identifying primary and foreign keys in custom record types

To find primary and foreign keys in a custom record type, you can use the following query, replacing `CUSTOM_RECORD_NAME` with your custom record type name:

```
select pktable_name, pkcolumn_name, pk_name, fktable_name, fkcolumn_name, fk_name from oa_fkeys
 where pktable_name = 'CUSTOM_RECORD_NAME';
```

In the query output, the **pk_name** column contains the primary key name, and the **fk_name** column shows the foreign key name. For more examples of queries over primary and foreign keys, see oa_fkeys.

Linking Gift Certificates to Transaction Line Items

You can return Transaction Lines table data together with Gift Certificates table data by querying for records where the Transaction Lines `memo` column value matches the Gift Certificates `gift_certificate_id` value. In addition, you can return Items table data by querying for records where the Transaction Lines `item_id` column value matches the Items `item_id` column value, and the value for the Items `type_name` column is '`Gift Certificate`'.

Use queries like the following for these data joins:

```
select * from TRANSACTION_LINES tl, GIFT_CERTIFICATES gc
where tl.MEMO = gc.GIFT_CERTIFICATE_ID

select *from TRANSACTION_LINES tl, GIFT_CERTIFICATES gc, ITEMS i
where tl.MEMO = gc.GIFT_CERTIFICATE_ID
  and tl.ITEM_ID = i.ITEM_ID
  and i.TYPE_NAME = 'Gift Certificate'
```

Connect Access to Transaction Credit and Debit Amounts

Credit and debit amounts are not exposed as columns in the Transactions or Transaction Lines tables. However, you can obtain transaction credit and debit amounts from the Transaction Lines table with queries like the following:

To obtain the credit amount for a transaction:

```
SELECT TRANSACTION_ID, NULLIF(GREATEST(-1*TRANSACTION_LINES.AMOUNT,0),0) "CREDITAMOUNT" FROM TR
ANSACTION_LINES WHERE COMPANY_ID = YOUR_ID
```

To obtain the debit amount for a transaction:

```
SELECT TRANSACTION_ID, NULLIF(GREATEST(TRANSACTION_LINES.AMOUNT,0),0) "DEBITAMOUNT" FROM TRANSA
CTION_LINES WHERE COMPANY_ID = YOUR_ID
```

These results can be useful for financial reporting purposes.

Connect Access to Transaction Quantities

Item count and quantity values are not exposed as columns in the Transactions table. However, you can obtain these values from the Transaction Lines table with a query like the following:

```
SELECT TRANSACTIONS.TRANSACTION_ID, TRANSACTION_LINES.ITEM_COUNT FROM TRANSACTIONS, TRANSACTION
_LINES WHERE TRANSACTION_LINES.TRANSACTION_ID = TRANSACTIONS.TRANSACTION_ID AND TRANSACTION_LIN
ES.TRANSACTION_LINE_ID = 0;
```

These results can be useful for inventory reporting purposes.

Using Qualified Queries

If you prefer to use fully qualified table names in your SuiteAnalytics Connect queries, make sure that you use exact qualifier values. Inexact qualifier values are those that contain additional spaces or use lowercase characters instead of uppercase characters, or vice versa. Queries that use inexact qualifier values will fail.

To learn exact qualifier values for your account:

1. Run the following query:

   ```
   select distinct table_qualifier, table_owner from oa_tables where table_owner !='SYSTEM';
   ```

2. Make a note of the **table_qualifier** and **table_owner** values that are returned. These are the values that should be used in your qualified queries.

   The **table_qualifier** and **table_owner** values should correspond to the company name and role name for the Account ID and Role ID you use to connect to the SuiteAnalytics Connect service.

   For example, if the returned value of **table_qualifier** is **Wolfe Company** and the value of **table_owner** is **Administrator**, your qualified queries should reference SuiteAnalyitcs Connect tables in the following way:

```
select * from "Wolfe Company"."Administrator".<table_name>;
```

If this query used an inexact **table_qualifier** value, such as **WolfeCompany** or **wolfecompany**, it would fail to run.