



# **Tibco Data Virtualization<sup>®</sup>**

## **Oracle Eloqua Adapter Guide**

*Version 8.7.0*  
*April 2023*



# Contents

---

<b>Contents</b>	<b>2</b>
<b>Oracle Eloqua Adapter</b>	<b>4</b>
Getting Started	4
Basic Tab	5
Logging	5
Using OAuth Authentication	7
Fine-Tuning Data Access	11
Changelog	12
Advanced Features	14
User Defined Views	15
SSL Configuration	18
Firewall and Proxy	18
Query Processing	19
Logging	20
SQL Compliance	23
SELECT Statements	24
SELECT INTO Statements	26
INSERT Statements	26
UPDATE Statements	27
DELETE Statements	28
EXECUTE Statements	29
PIVOT and UNPIVOT	29
Data Model	31
Tables	33
Views	76
Stored Procedures	187
Connection String Options	196

Authentication .....	201
OAuth .....	203
SSL .....	210
Firewall .....	212
Proxy .....	215
Logging .....	222
Schema .....	222
Miscellaneous .....	224
<b>TIBCO Product Documentation and Support Services .....</b>	<b>236</b>
How to Access TIBCO Documentation .....	236
How to Contact TIBCO Support .....	237
Release Version Support .....	237
How to Join TIBCO Community .....	238
<b>Legal and Third-Party Notices .....</b>	<b>239</b>

# Oracle Eloqua Adapter

---

## Oracle Eloqua Version Support

The adapter models the Oracle Eloqua REST API as bidirectional tables. Bulk operations are supported through version 2.0 of the Bulk API.

## SQL Compliance

The [SQL Compliance](#) section shows the SQL syntax supported by the adapter and points out any limitations.

# Getting Started

## Connecting to Oracle Eloqua

[Basic Tab](#) shows how to authenticate to Oracle Eloqua and configure any necessary connection properties. Additional adapter capabilities can be configured using the available [Connection](#) properties on the Advanced tab. The Advanced Settings section shows how to set up more advanced configurations and troubleshoot connection errors.

## Deploying the Oracle Eloqua Adapter

To deploy the adapter, you can execute the `server_util` utility via the command line by

1. Unzip the `tdv.oracleeloqua.zip` file to the location of your choice.
2. Open a command prompt window.
3. Navigate to the `<TDV_install_dir>/bin`
4. Enter the `server_util` command with the `-deploy` option:

```
server_util -server <hostname> [-port <port>] -user <user> -  
password <password> -deploy -package <TDV_install_  
dir>/adapters/tdv.oracleeloqua/tdv.oracleeloqua.jar
```

Note: When deploying a build of an existing adapter, you will need to undeploy the existing adapter using the `server_util` command with the `-undeploy` option.

```
server_util -server <hostname> [-port <port>] -user <user> -password  
<password> -undeploy -version 1 -name Eloqua
```

## Basic Tab

### Authenticating to Oracle Eloqua

#### Connect via Login

To connect via Login, set AuthScheme to Basic. The standard method of authentication to Oracle Eloqua is via the login method. The Login method requires you to set Company and to set User, and Password to the credentials you use to log in. This method uses HTTP Basic authentication over SSL.

#### Connect via OAuth

To connect via OAuth, set AuthScheme to OAuth. See [Using OAuth Authentication](#) for an authentication guide.

## Logging

The adapter uses TDV Server's logging (log4j) to generate log files. The settings within the TDV Server's logging (log4j) configuration file are used by the adapter to determine the type of messages to log. The following categories can be specified:

- Error: Only error messages are logged.
- Info: Both Error and Info messages are logged.
- Debug: Error, Info, and Debug messages are logged.

The Other property of the adapter can be used to set Verbosity to specify the amount of detail to be included in the log file, that is:

```
Verbosity=4;
```

You can use Verbosity to specify the amount of detail to include in the log within a category. The following verbosity levels are mapped to the log4j categories:

- 0 = Error
- 1-2 = Info
- 3-5 = Debug

For example, if the log4j category is set to DEBUG, the Verbosity option can be set to 3 for the minimum amount of debug information or 5 for the maximum amount of debug information.

Note that the log4j settings override the Verbosity level specified. The adapter never logs at a Verbosity level greater than what is configured in the log4j properties. In addition, if Verbosity is set to a level less than the log4j category configured, Verbosity defaults to the minimum value for that particular category. For example, if Verbosity is set to a value less than 3 and the Debug category is specified, the Verbosity defaults to 3.

The following list is an explanation of the Verbosity levels and the information that they log.

- 1 - Will log the query, the number of rows returned by it, the start of execution and the time taken, and any errors.
- 2 - Will log everything included in Verbosity 1 and HTTP headers.
- 3 - Will additionally log the body of the HTTP requests.
- 4 - Will additionally log transport-level communication with the data source. This includes SSL negotiation.
- 5 - Will additionally log communication with the data source and additional details that may be helpful in troubleshooting problems. This includes interface commands.

## Configure Logging for the Oracle Eloqua Adapter

By default, logging is turned on without debugging. If debugging information is desired, uncomment the following line in the TDV Server's log4j.properties file (default location of this file is: C:\Program Files\TIBCO\TDV Server <version>\conf\server):

```
log4j.logger.com.cdata=DEBUG
```

The TDV Server must be restarted after changing the log4j.properties file, which can be accomplished by running the composite.bat script located at: C:\Program Files\TIBCO\TDV

Server <version>\bin. Note that reauthenticating to the TDV Studio is required after restarting the server.

Here is an example of the calls:

```
.\composite.bat monitor restart
```

All logs for the adapter are written to the "cs\_server\_dsrc.log" file as specified in the log4j properties.

**Note:** The "log4j.logger.com.cdata=DEBUG" option is not required if the **Debug Output Enabled** option is set to true within the TDV Studio. To set this option, navigate to **Administrator > Configuration**. Select **Server > Configuration > Debugging** and set the Debug Output Enabled option to **True**.

## Using OAuth Authentication

Use the OAuth authentication standard to connect to Oracle Eloqua. OAuth requires the authenticating user to interact with Oracle Eloqua using the browser.

### Custom Credentials

You will need to register an app to obtain the OAuthClientId and OAuthClientSecret. See [Custom Credentials](#) for a procedure.

### Headless Machines

In the headless OAuth flow, users need to authenticate via a browser on another machine. You need to create a custom OAuth app. See [Creating a Custom OAuth App](#) for the procedure.

### Creating a Custom OAuth App

You may want to create a custom OAuth app to change the information displayed when users log into the Oracle Eloqua. You can register an app to obtain the OAuthClientId and OAuthClientSecret.

See [Creating a Custom OAuth App](#) for a procedure.

## Creating a Custom OAuth App

To obtain an OAuthClientId, OAuthClientSecret, CallbackURL, and you first need to create an app linked to your Oracle Eloqua account.

### Create and Configure a Custom OAuth App

You can follow the procedure below to obtain the client credentials, the client Id (app Id) and client secret:

1. Log in to Oracle Eloqua.
2. Click Settings -> AppCloud Developer (under Platform Extensions) -> Create New App.
3. Enter a name, icon, and description for the app to be displayed to users when they connect to Oracle Eloqua. The OAuthClientId and OAuthClientSecret will be displayed.

When you connect the add-in opens the OAuth endpoint in your default browser. Log in and grant permissions to the application.

## Custom Credentials

This section describes desktop authentication using the credentials for your custom OAuth app. See [Creating a Custom OAuth App](#) for more information.

### Desktop Authentication with your OAuth App

Follow the steps below to authenticate with the credentials for a custom OAuth app. See [Creating a Custom OAuth App](#).

### Get an OAuth Access Token

After setting the following, you are ready to connect:

- OAuthClientId: Set this to the Client Id in your app settings.
- OAuthClientSecret: Set this to the Client Secret in your app settings.
- CallbackURL: Set this to the Redirect URL in your app settings.
- InitiateOAuth: Set this to GETANDREFRESH. You can use InitiateOAuth to avoid repeating the OAuth exchange and manually setting the OAuthAccessToken.

When you connect the adapter opens the OAuth endpoint in your default browser. Log in



and grant permissions to the application. The adapter then completes the OAuth process:

1. Extracts the access token from the callback URL and authenticates requests.
2. Saves OAuth values in OAuthSettingsLocation to be persisted across connections.

## The Password GrantType

When setting the OAuthGrantType to password, an OAuth flow may be triggered that passes user credentials. In this case, there will be no browser window to authorize. Instead, the following connection properties will need to be set:

- User - Set this to the user name.
- User - Set this to the password.
- Company - Set the company for the user.
- OAuthClientId: Set this to the Client Id in your app settings.
- OAuthClientSecret: Set this to the Client Secret in your app settings.
- OAuthGrantType: Set this to password.
- InitiateOAuth: Set this to GETANDREFRESH. You can use InitiateOAuth to avoid repeating the OAuth exchange and manually setting the OAuthAccessToken.

## Headless Machines

### Using OAuth on a Headless Machine

To create Oracle Eloqua data sources on headless servers or other machines on which the adapter cannot open a browser, you need to authenticate from another machine. Authentication is a two-step process.

1. Instead of installing the adapter on another machine, you can follow the steps below to obtain the OAuthVerifier value. Or, you can install the adapter on another machine and transfer the OAuth authentication values, after you authenticate through the usual browser-based flow.
2. You can then configure the adapter to automatically refresh the access token from the headless machine.

## Using the Credentials for a Custom OAuth App

### Create a Custom OAuth App

See [Creating a Custom OAuth App](#) for a procedure. You can then follow the procedures below to authenticate and connect to data.

### Obtain a Verifier Code

Set the following properties on the headless machine:

- InitiateOAuth: Set this to **OFF**.
- OAuthClientId: Set this to the App Id in your app settings.
- OAuthClientSecret: Set this to the App Secret in your app settings.

You can then follow the steps below to authenticate from another machine and obtain the OAuthVerifier connection property.

1. Call the [GetOAuthAuthorizationUrl](#) stored procedure with the CallbackURL input parameter set to the exact Redirect URI you specified in your app settings.
2. Open the returned URL in a browser. Log in and grant permissions to the adapter. You are then redirected to the callback URL, which contains the verifier code.
3. Save the value of the verifier code. You will set this in the OAuthVerifier connection property.

On the headless machine, set the following connection properties to obtain the OAuth authentication values:

- OAuthClientId: Set this to the consumer key in your app settings.
- OAuthClientSecret: Set this to the consumer secret in your app settings.
- OAuthVerifier: Set this to the verifier code.
- OAuthSettingsLocation: Set this to persist the encrypted OAuth authentication values to the specified file.
- InitiateOAuth: Set this to **REFRESH**.

### Connect to Data

After the OAuth settings file is generated, set the following properties to connect to data:

- OAuthSettingsLocation: Set this to the file containing the encrypted OAuth

authentication values. Make sure this file gives read and write permissions to the provider to enable the automatic refreshing of the access token.

- InitiateOAuth: Set this to **REFRESH**.

### Transfer OAuth Settings

Follow the steps below to install the adapter on another machine, authenticate, and then transfer the resulting OAuth values.

On a second machine, install the adapter and connect with the following properties set:

- OAuthSettingsLocation: Set this to a writable text file.
- InitiateOAuth: Set this to **GETANDREFRESH**.
- OAuthClientId: Set this to the Client Id in your app settings.
- OAuthClientSecret: Set this to the Client Secret in your app settings.
- CallbackURL: Set this to the Callback URL in your app settings.

Test the connection to authenticate. The resulting authentication values are written, encrypted, to the path specified by OAuthSettingsLocation. Once you have successfully tested the connection, copy the OAuth settings file to your headless machine. On the headless machine, set the following connection properties to connect to data:

- InitiateOAuth: Set this to **REFRESH**.
- OAuthSettingsLocation: Set this to the path to your OAuth settings file. Make sure this file gives read and write permissions to the adapter to enable the automatic refreshing of the access token.

## Fine-Tuning Data Access

### Fine Tuning Data Access

You can use the following properties to gain more granular control over how the adapter surfaces the functionality of the underlying Oracle Eloqua APIs.

The adapter uses the bulk API when possible; you can fine-tune the connectivity to the bulk API with the following connection properties:

- UseBulkAPI
- BulkPollingInterval

- BulkQueryTimeout
- DataRetentionDuration

Additionally, the following properties are useful to circumvent failed bulk API requests:

- RetryCount
- RetryInterval

Oracle Eloqua accepts characters for table and column names that must be escaped in SQL. You can set UseSimpleNames to true to report nonalphanumeric characters as underscores.

## Changelog

### General Changes

Date	Build Number	Change Type	Description
12/14/2022	2383	General	<b>Changed</b> <ul style="list-style-type: none"> <li>Added the Default column to the sys_procedureparameters table.</li> </ul>
09/30/2022	2308	General	<b>Changed</b> <ul style="list-style-type: none"> <li>Added the IsPath column to the sys_procedureparameters table.</li> </ul>
08/17/2022	2264	General	<b>Changed</b> <ul style="list-style-type: none"> <li>We now support handling the keyword "COLLATE" as standard function name as well.</li> </ul>
04/13/2021	2138	Oracle Eloqua	<b>Added</b> <ul style="list-style-type: none"> <li>Added columns for the table Activity_PageView, including PageViewSavedID,CampaignResponseDate,CampaignResponseM</li> </ul>

		emberStatus.	<ul style="list-style-type: none"> <li>Added columns for the table Campaign, included IsEmailMarketingCampaign,isIncludedInROI,isSynchedWithCRM,runAsUserId,type,FirstActivation,description,campaignCategory.</li> </ul>
09/02/2021	General	<b>Added</b>	<ul style="list-style-type: none"> <li>Added support for the STRING_SPLIT table-valued function in the CROSS APPLY clause.</li> </ul>
08/07/2021	General	<b>Changed</b>	<ul style="list-style-type: none"> <li>Added the KeySeq column to the sys_foreignkeys table.</li> </ul>
08/06/2021	General	<b>Changed</b>	<ul style="list-style-type: none"> <li>Added the new sys_primarykeys system table.</li> </ul>
07/23/2021	General	<b>Changed</b>	<ul style="list-style-type: none"> <li>Updated the Literal Function Names for relative date/datetime functions. Previously relative date/datetime functions resolved to a different value when used in the projection vs te predicate. Ie: SELECT LAST_MONTH() AS lm, Col FROM Table WHERE Col &gt; LAST_MONTH(). Formerly the two LAST_MONTH() methods would resolve to different datetimes. Now they will match.</li> <li>As a replacement for the previous behavior, the relative date/datetime functions in the criteria may have an 'L' appended to them. Ie: WHERE col &gt; L_LAST_MONTH(). This will continue to resolve to the same values that previously were calculated in the criteria. Note that the "L_" prefix will only work in the predicate - it not available for the projection.</li> </ul>
07/08/2021	General	<b>Added</b>	<ul style="list-style-type: none"> <li>Added the TCP Logging Module for the logging information happening on the TCP wire protocol. The transport bytes that are incoming and ongoing will be logged at verbosity=5.</li> </ul>
04/23/2021	General	<b>Added</b>	<ul style="list-style-type: none"> <li>Added support for handling client side formulas during insert /</li> </ul>

			update. For example: UPDATE Table SET Col1 = Concat(Col1, " - ", Col2) WHERE Col2 LIKE 'A%'
04/23/2021	General	<b>Changed</b>	<ul style="list-style-type: none"> <li>Updated how display sizes are determined for varchar primary key and foreign key columns so they will match the reported length of the column.</li> </ul>
04/16/2021	General	<b>Added</b>	<ul style="list-style-type: none"> <li>Non-conditional updates between two columns is now available to all drivers. For example: UPDATE Table SET Col1=Col2</li> </ul>
		<b>Changed</b>	<ul style="list-style-type: none"> <li>Reduced the length to 255 for varchar primary key and foreign key columns.</li> <li>Updated implicit and metadata caching to improve performance and support for multiple connections. Old metadata caches are not compatible - you would need to generate new metadata caches if you are currently using CacheMetadata.</li> <li>Updated index naming convention to avoid duplicates</li> <li>Updated and standardized Getting Started connection help.</li> <li>Added the Advanced Features section to the help of all drivers.</li> <li>Categorized connection property listings in the help for all editions.</li> </ul>
04/15/2021	General	<b>Changed</b>	<ul style="list-style-type: none"> <li>Kerberos authentication is updated to use TCP by default, but will fall back to UDP if a TCP connection cannot be established</li> </ul>

## Advanced Features

This section details a selection of advanced features of the Oracle Eloqua adapter.

## User Defined Views

The adapter allows you to define virtual tables, called *user defined views*, whose contents are decided by a pre-configured query. These views are useful when you cannot directly control queries being issued to the drivers. See [User Defined Views](#) for an overview of creating and configuring custom views.

## SSL Configuration

Use [SSL Configuration](#) to adjust how adapter handles TLS/SSL certificate negotiations. You can choose from various certificate formats; see the [SSLServerCert](#) property under "Connection String Options" for more information.

## Firewall and Proxy

Configure the adapter for compliance with [Firewall and Proxy](#), including Windows proxies and HTTP proxies. You can also set up tunnel connections.

## Query Processing

The adapter offloads as much of the SELECT statement processing as possible to Oracle Eloqua and then processes the rest of the query in memory (client-side).

See [Query Processing](#) for more information.

## Logging

See [Logging](#) for an overview of configuration settings that can be used to refine CData logging. For basic logging, you only need to set two connection properties, but there are numerous features that support more refined logging, where you can select subsets of information to be logged using the [LogModules](#) connection property.

## User Defined Views

The Oracle Eloqua Adapter allows you to define a virtual table whose contents are decided by a pre-configured query. These are called *User Defined Views*, which are useful in situations where you cannot directly control the query being issued to the driver, e.g. when using the driver from a tool. The User Defined Views can be used to define predicates that are always applied. If you specify additional predicates in the query to the view, they are

combined with the query already defined as part of the view.

There are two ways to create user defined views:

- Create a JSON-formatted configuration file defining the views you want.
- DDL statements.

## Defining Views Using a Configuration File

User Defined Views are defined in a JSON-formatted configuration file called *UserDefinedViews.json*. The adapter automatically detects the views specified in this file.

You can also have multiple view definitions and control them using the UserDefinedViews connection property. When you use this property, only the specified views are seen by the adapter.

This User Defined View configuration file is formatted as follows:

- Each root element defines the name of a view.
- Each root element contains a child element, called **query**, which contains the custom SQL query for the view.

For example:

```
{
  "MyView": {
    "query": "SELECT * FROM Contact WHERE MyColumn = 'value'"
  },
  "MyView2": {
    "query": "SELECT * FROM MyTable WHERE Id IN (1,2,3)"
  }
}
```

Use the UserDefinedViews connection property to specify the location of your JSON configuration file. For example:

```
"UserDefinedViews",
"C:\\Users\\yourusername\\Desktop\\tmp\\UserDefinedViews.json"
```



## Defining Views Using DDL Statements

The adapter is also capable of creating and altering the schema via DDL Statements such as CREATE LOCAL VIEW, ALTER LOCAL VIEW, and DROP LOCAL VIEW.

### Create a View

To create a new view using DDL statements, provide the view name and query as follows:

```
CREATE LOCAL VIEW [MyViewName] AS SELECT * FROM Customers LIMIT 20;
```

If no JSON file exists, the above code creates one. The view is then created in the JSON configuration file and is now discoverable. The JSON file location is specified by the UserDefinedViews connection property.

### Alter a View

To alter an existing view, provide the name of an existing view alongside the new query you would like to use instead:

```
ALTER LOCAL VIEW [MyViewName] AS SELECT * FROM Customers WHERE  
TimeModified > '3/1/2020';
```

The view is then updated in the JSON configuration file.

### Drop a View

To drop an existing view, provide the name of an existing schema alongside the new query you would like to use instead.

```
DROP LOCAL VIEW [MyViewName]
```

This removes the view from the JSON configuration file. It can no longer be queried.

## Schema for User Defined Views

User Defined Views are exposed in the **UserViews** schema by default. This is done to avoid the view's name clashing with an actual entity in the data model. You can change the name of the schema used for UserViews by setting the UserViewsSchemaName property.

## Working with User Defined Views

For example, a SQL statement with a User Defined View called *UserViews.RCustomers* only lists customers in Raleigh:

```
SELECT * FROM Customers WHERE City = 'Raleigh';
```

An example of a query to the driver:

```
SELECT * FROM UserViews.RCustomers WHERE Status = 'Active';
```

Resulting in the effective query to the source:

```
SELECT * FROM Customers WHERE City = 'Raleigh' AND Status = 'Active';
```

That is a very simple example of a query to a User Defined View that is effectively a combination of the view query and the view definition. It is possible to compose these queries in much more complex patterns. All SQL operations are allowed in both queries and are combined when appropriate.

## SSL Configuration

### Customizing the SSL Configuration

By default, the adapter attempts to negotiate SSL/TLS by checking the server's certificate against the system's trusted certificate store.

To specify another certificate, see the [SSLServerCert](#) property for the available formats to do so.

## Firewall and Proxy

### Connecting Through a Firewall or Proxy

#### HTTP Proxies

To connect through the Windows system proxy, you do not need to set any additional connection properties. To connect to other proxies, set [ProxyAutoDetect](#) to false.

In addition, to authenticate to an HTTP proxy, set ProxyAuthScheme, ProxyUser, and ProxyPassword, in addition to ProxyServer and ProxyPort.

## Other Proxies

Set the following properties:

- To use a proxy-based firewall, set FirewallType, FirewallServer, and FirewallPort.
- To tunnel the connection, set FirewallType to TUNNEL.
- To authenticate, specify FirewallUser and FirewallPassword.
- To authenticate to a SOCKS proxy, additionally set FirewallType to SOCKS5.

# Query Processing

## Query Processing

CData has a client-side SQL engine built into the adapter library. This enables support for the full capabilities that SQL-92 offers, including filters, aggregations, functions, etc.

For sources that do not support SQL-92, the adapter offloads as much of SQL statement processing as possible to Oracle Eloqua and then processes the rest of the query in memory (client-side). This results in optimal performance.

For data sources with limited query capabilities, the adapter handles transformations of the SQL query to make it simpler for the adapter. The goal is to make smart decisions based on the query capabilities of the data source to push down as much of the computation as possible. The Oracle Eloqua Query Evaluation component examines SQL queries and returns information indicating what parts of the query the adapter is not capable of executing natively.

The Oracle Eloqua Query Slicer component is used in more specific cases to separate a single query into multiple independent queries. The client-side Query Engine makes decisions about simplifying queries, breaking queries into multiple queries, and pushing down or computing aggregations on the client-side while minimizing the size of the result set.

There's a significant trade-off in evaluating queries, even partially, client-side. There are always queries that are impossible to execute efficiently in this model, and some can be particularly expensive to compute in this manner. CData always pushes down as much of

the query as is feasible for the data source to generate the most efficient query possible and provide the most flexible query capabilities.

## More Information

For a full discussion of how CData handles query processing, see [CData Architecture: Query Execution](#).

# Logging

Capturing adapter logging can be very helpful when diagnosing error messages or other unexpected behavior.

## Basic Logging

You will simply need to set two connection properties to begin capturing adapter logging.

- Logfile: A filepath which designates the name and location of the log file.
- Verbosity: This is a numerical value (1-5) that determines the amount of detail in the log. See the page in the Connection Properties section for an explanation of the five levels.
- MaxLogFileSize: When the limit is hit, a new log is created in the same folder with the date and time appended to the end. The default limit is 100 MB. Values lower than 100 kB will use 100 kB as the value instead.
- MaxLogFileCount: A string specifying the maximum file count of log files. When the limit is hit, a new log is created in the same folder with the date and time appended to the end and the oldest log file will be deleted. Minimum supported value is 2. A value of 0 or a negative value indicates no limit on the count.

Once this property is set, the adapter will populate the log file as it carries out various tasks, such as when authentication is performed or queries are executed. If the specified file doesn't already exist, it will be created.

## Log Verbosity

The verbosity level determines the amount of detail that the adapter reports to the Logfile. Verbosity levels from 1 to 5 are supported. These are described in the following list:

- 
- |   |   |
|---|---|
| 1 | Setting <u>Verbosity</u> to 1 will log the query, the number of rows returned by it, the start of execution and the time taken, and any errors. |
|---|---|
- 
- |   |  |
|---|--|
| 2 | Setting <u>Verbosity</u> to 2 will log everything included in <u>Verbosity</u> 1 and additional information about the request. |
|---|--|
- 
- |   |  |
|---|--|
| 3 | Setting <u>Verbosity</u> to 3 will additionally log HTTP headers, as well as the body of the request and the response. |
|---|--|
- 
- |   |  |
|---|--|
| 4 | Setting <u>Verbosity</u> to 4 will additionally log transport-level communication with the data source. This includes SSL negotiation. |
|---|--|
- 
- |   |  |
|---|--|
| 5 | Setting <u>Verbosity</u> to 5 will additionally log communication with the data source and additional details that may be helpful in troubleshooting problems. This includes interface commands. |
|---|--|
- 

The Verbosity should not be set to greater than 1 for normal operation. Substantial amounts of data can be logged at higher verbositys, which can delay execution times.

To refine the logged content further by showing/hiding specific categories of information, see LogModules.

## Sensitive Data

Verbosity levels 3 and higher may capture information that you do not want shared outside of your organization. The following lists information of concern for each level:

- Verbosity 3: The full body of the request and the response, which includes all the data returned by the adapter
- Verbosity 4: SSL certificates
- Verbosity 5: Any extra transfer data not included at Verbosity 3, such as non human-readable binary transfer data

## Best Practices for Data Security

Although we mask sensitive values, such as passwords, in the connection string and any request in the log, it is always best practice to review the logs for any sensitive information before sharing outside your organization.

## Java Logging

When Java logging is enabled in Logfile, the Verbosity will instead map to the following logging levels.

- 0: Level.WARNING
- 1: Level.INFO
- 2: Level.CONFIG
- 3: Level.FINE
- 4: Level.FINER
- 5: Level.FINEST

## Advanced Logging

You may want to refine the exact information that is recorded to the log file. This can be accomplished using the LogModules property.

This property allows you to filter the logging using a semicolon-separated list of logging modules.

All modules are four characters long. **Please note that modules containing three letters have a required trailing blank space.** The available modules are:

- **EXEC:** Query Execution. Includes execution messages for original SQL queries, parsed SQL queries, and normalized SQL queries. Query and page success/failure messages appear here as well.
- **INFO:** General Information. Includes the connection string, driver version (build number), and initial connection messages.
- **HTTP:** HTTP Protocol messages. Includes HTTP requests/responses (including POST messages), as well as Kerberos related messages.
- **SSL :** SSL certificate messages.
- **OAUT:** OAuth related failure/success messages.
- **SQL :** Includes SQL transactions, SQL bulk transfer messages, and SQL result set messages.
- **META:** Metadata cache and schema messages.

- **TCP** : Incoming and Ongoing raw bytes on TCP transport layer messages. An example value for this property would be.

```
LogModules=INFO;EXEC;SSL ;SQL ;META;
```

Note that these modules refine the information as it is pulled after taking the [Verbosity](#) into account.

## SQL Compliance

The Oracle Eloqua Adapter supports several operations on data, including querying, deleting, modifying, and inserting.

### SELECT Statements

See [SELECT Statements](#) for a syntax reference and examples.

See [Data Model](#) for information on the capabilities of the Oracle Eloqua API.

### INSERT Statements

See [INSERT Statements](#) for a syntax reference and examples, as well as retrieving the new records' Ids.

### UPDATE Statements

The primary key Id is required to update a record. See [UPDATE Statements](#) for a syntax reference and examples.

### DELETE Statements

The primary key Id is required to delete a record. See [DELETE Statements](#) for a syntax reference and examples.

### EXECUTE Statements

Use EXECUTE or EXEC statements to execute stored procedures. See [EXECUTE Statements](#) for a syntax reference and examples.

## Names and Quoting

- Table and column names are considered identifier names; as such, they are restricted to the following characters: [A-Z, a-z, 0-9, \_:@].
- To use a table or column name with characters not listed above, the name must be quoted using double quotes ("name") in any SQL statement.
- Strings must be quoted using single quotes (e.g., 'John Doe').

## SELECT Statements

A SELECT statement can consist of the following basic clauses.

- SELECT
- INTO
- FROM
- JOIN
- WHERE
- GROUP BY
- HAVING
- UNION
- ORDER BY
- LIMIT

## SELECT Syntax

The following syntax diagram outlines the syntax supported by the Oracle Eloqua adapter:

```
SELECT {
  [ TOP <numeric_literal> ]
  {
    *
    | {
      <expression> [ [ AS ] <column_reference> ]
      | { <table_name> | <correlation_name> } .*
    } [ , ... ]
  }
}
```



```

}
[ INTO csv:// [ filename= ] <file_path> [ ;delimiter=tab ] ]
{
  FROM <table_reference> [ [ AS ] <identifier> ]
}
[ WHERE <search_condition> ]
[
  ORDER BY
    <column_reference> [ ASC | DESC ] [ NULLS FIRST | NULLS LAST ]
]
} | SCOPE_IDENTITY()
<expression> ::=
  | <column_reference>
  | @ <parameter>
  | ?
  | COUNT( * | { [ DISTINCT ] <expression> } )
  | { AVG | MAX | MIN | SUM | COUNT } ( <expression> )
  | NULLIF ( <expression> , <expression> )
  | COALESCE ( <expression> , ... )
  | CASE <expression>
    WHEN { <expression> | <search_condition> } THEN { <expression> |
NULL } [ ... ]
  [ ELSE { <expression> | NULL } ]
  END
  | <literal>
  | <sql_function>
<search_condition> ::=
  {
    <expression> { = | != | < | > | >= | <= | AND } [ <expression> ]
  } [ { AND | OR } ... ]

```

## Examples

1. Return all columns:

```
SELECT * FROM Contact
```

2. Rename a column:

```
SELECT "Name" AS MY_Name FROM Contact
```

3. Cast a column's data as a different data type:

```
SELECT CAST(AnnualRevenue AS VARCHAR) AS Str_AnnualRevenue FROM
Contact
```

4. Search data:

```
SELECT * FROM Contact WHERE Country = 'U.S.A.'
```

5. The Oracle Eloqua APIs support the following operators in the WHERE clause: =, !=, <, >, >=, <=, AND.

```
SELECT * FROM Contact WHERE Country = 'U.S.A.';
```

6. Sort a result set in ascending order:

```
SELECT SalesPerson, Name FROM Contact ORDER BY Name ASC
```

## SELECT INTO Statements

You can use the SELECT INTO statement to export formatted data to a file.

### Data Export with an SQL Query

The following query exports data into a file formatted in comma-separated values (CSV):

```
boolean ret = stat.execute("SELECT SalesPerson, Name INTO
'csv://c:/Contact.txt' FROM 'Contact' WHERE Country = 'U.S.A.'");
System.out.println(stat.getUpdateCount()+" rows affected");
```

You can specify other file formats in the URI. The following example exports tab-separated values:

```
Statement stat = conn.createStatement();
boolean ret = stat.execute("SELECT * INTO 'Contact' IN
'csv://filename=c:/Contact.csv;delimiter=tab' FROM 'Contact' WHERE
Country = 'U.S.A.'");
System.out.println(stat.getUpdateCount()+" rows affected");
```

## INSERT Statements

To create new records, use INSERT statements.

## INSERT Syntax

The INSERT statement specifies the columns to be inserted and the new column values. You can specify the column values in a comma-separated list in the VALUES clause, as shown in the following example:

```
INSERT INTO <table_name>
( <column_reference> [ , ... ] )
VALUES
( { <expression> | NULL } [ , ... ] )

<expression> ::=
    | @ <parameter>
    | ?
    | <literal>
```

You can use the executeUpdate method of the Statement and PreparedStatement classes to execute data manipulation commands and retrieve the rows affected. To retrieve the Id of the last inserted record use getGeneratedKeys. Additionally, set the **RETURN\_GENERATED\_KEYS** flag of the Statement class when you call prepareStatement.

```
String cmd = "INSERT INTO Contact (Name) VALUES (?)";
PreparedStatement pstmt = connection.prepareStatement
(cmd,Statement.RETURN_GENERATED_KEYS);
pstmt.setString(1, "John");
int count = pstmt.executeUpdate();
System.out.println(count+" rows were affected");
ResultSet rs = pstmt.getGeneratedKeys();
while(rs.next()){
    System.out.println(rs.getString("Id"));
}
connection.close();
```

## UPDATE Statements

To modify existing records, use UPDATE statements.

### Update Syntax

The UPDATE statement takes as input a comma-separated list of columns and new column values as name-value pairs in the SET clause, as shown in the following example:

```

UPDATE <table_name> SET { <column_reference> = <expression> } [ , ... ]
WHERE { Id = <expression> } [ { AND | OR } ... ]
<expression> ::=
    | @ <parameter>
    | ?
    | <literal>

```

You can use the `executeUpdate` method of the `Statement` or `PreparedStatement` classes to execute data manipulation commands and retrieve the rows affected, as shown in the following example:

```

String cmd = "UPDATE Contact SET Name='John' WHERE Id = ?";
PreparedStatement pstmt = connection.prepareStatement(cmd);
pstmt.setString(1, "1");
int count = pstmt.executeUpdate();
System.out.println(count + " rows were affected");
connection.close();

```

## DELETE Statements

To delete information from a table, use DELETE statements.

### DELETE Syntax

The DELETE statement requires the table name in the FROM clause and the row's primary key in the WHERE clause, as shown in the following example:

```

<delete_statement> ::= DELETE FROM <table_name> WHERE { Id =
<expression> } [ { AND | OR } ... ]
<expression> ::=
    | @ <parameter>
    | ?
    | <literal>

```

You can use the `executeUpdate` method of the `Statement` or `PreparedStatement` classes to execute data manipulation commands and retrieve the number of affected rows, as shown in the following example:

```

Connection connection = DriverManager.getConnection
("jdbc:oracleloqua:User=user;Password=password;Company=MyCompany",);
String cmd = "DELETE FROM Contact WHERE Id = ?";
PreparedStatement pstmt = connection.prepareStatement(cmd);

```

```
pstmt.setString(1, "1");
int count=pstmt.executeUpdate();
connection.close();
```

## EXECUTE Statements

To execute stored procedures, you can use EXECUTE or EXEC statements.

EXEC and EXECUTE assign stored procedure inputs, referenced by name, to values or parameter names.

### Stored Procedure Syntax

To execute a stored procedure as an SQL statement, use the following syntax:

```
{ EXECUTE | EXEC } <stored_proc_name>
{
  [ @ ] <input_name> = <expression>
} [ , ... ]
<expression> ::=
  | @ <parameter>
  | ?
  | <literal>
```

### Example Statements

Reference stored procedure inputs by name:

```
EXECUTE my_proc @second = 2, @first = 1, @third = 3;
```

Execute a parameterized stored procedure statement:

```
EXECUTE my_proc second = @p1, first = @p2, third = @p3;
```

## PIVOT and UNPIVOT

**PIVOT** and **UNPIVOT** can be used to change a table-valued expression into another table.

## PIVOT

PIVOT rotates a table-value expression by turning unique values from one column into multiple columns in the output. PIVOT can run aggregations where required on any column value.

## PIVOT Syntax

```
"SELECT 'AverageCost' AS Cost_Sorted_By_Production_Days, [0], [1], [2],
[3], [4]
FROM
(
SELECT DaysToManufacture, StandardCost
FROM Production.Product
) AS SourceTable
PIVOT
(
AVG(StandardCost)
FOR DaysToManufacture IN ([0], [1], [2], [3], [4])
) AS PivotTable;"
```

## UNPIVOT

UNPIVOT carries out nearly the opposite to PIVOT by rotating columns of a table-valued expressions into column values.

## UNPIVOT Syntax

```
"SELECT VendorID, Employee, Orders
FROM
(SELECT VendorID, Emp1, Emp2, Emp3, Emp4, Emp5
FROM pvt) p
UNPIVOT
(Orders FOR Employee IN
(Emp1, Emp2, Emp3, Emp4, Emp5)
)AS unpvt;"
```

For further information on PIVOT and UNPIVOT, see [FROM clause plus JOIN, APPLY, PIVOT \(Transact-SQL\)](#)

# Data Model

The Oracle Eloqua Adapter models the Bulk API and the REST API as relational tables, views, and stored procedures. For example, Oracle Eloqua activity types are represented by the corresponding views. Views are tables that cannot be modified.

## Working with Oracle Eloqua APIs as Tables

All tables are accessible with the REST API, including custom tables. A limited number of tables and views can be used with the Bulk API. The adapter supports version 2.0 of the Bulk API and version 2.0 of the REST API.

## Custom Tables and Columns

The adapter can expose custom tables, views, and columns from Oracle Eloqua that are not mentioned in the [Tables](#) and [Views](#). The data model illustrates a sample of what your Oracle Eloqua data model might look like. The actual data model will be obtained dynamically based on your user credentials and Oracle Eloqua account. For example, the Account, Campaign, and Contact tables are dynamic: These tables contain the columns described and your custom columns.

## Tables Available in the Bulk API

The following tables and views, including custom tables, are supported with the Bulk API:

- Account
- Activity\_Bounceback
- Activity\_EmailClickThrough
- Activity\_EmailOpen
- Activity\_EmailSend
- Activity\_EmailSubscribe
- Activity\_EmailUnsubscribe
- Activity\_FormSubmit
- Activity\_PageView

- Activity\_WebVisit
- Contact
- Custom tables
- CampaignResponses

## Stored Procedures

[Stored Procedures](#) are function-like interfaces to Oracle Eloqua that complement the data available from Oracle Eloqua tables. Stored procedures implement actions available in the Bulk API and other API functionality that cannot be expressed as SELECT, INSERT, UPDATE, or DELETE statements.

## API SQL Limitations

Table-specific API limitations and requirements are documented in [Tables](#) and [Views](#). For both the Bulk and REST APIs, the following limitation exists for filtering on datetime columns: The "=" operator cannot be used with datetime columns in the WHERE clause. However, the "<", ">", ">=", and "<=" operators can be used to filter on datetime columns.

Additionally, the UpdatedAt column only supports filtering with the ">" operator.

Oracle Eloqua also has limited support on multiple conditions. In many tables the AND condition between filters is not supported. If AND is included the server might not return any results, nor will it return an error.

## Bulk API Limitations

DELETE and UPDATE are not supported by the Bulk API. Additionally, the column count must be <250 for SELECT and must be <= 100 for INSERT.

## REST API Limitations

Bulk updates are only supported in the Bulk API.

## Custom Table Limitations

Custom tables are read-only in the REST API. In the Bulk API, SELECT and INSERT are supported.



# Tables

The adapter models the data in Oracle Eloqua into a list of tables that can be queried using standard SQL statements.

Generally, querying Oracle Eloqua tables is the same as querying a table in a relational database. Sometimes there are special cases, for example, including a certain column in the WHERE clause might be required to get data for certain columns in the table. This is typically needed for situations where a separate request must be made for each row to get certain columns. These types of situations are clearly documented at the top of the table page linked below.

## Oracle Eloqua Adapter Tables

Name	Description
<a href="#">Account</a>	Create, update, delete, and query Accounts. This table is supported in bulk mode.
<a href="#">AccountGroup</a>	Retrieve Account group.
<a href="#">Campaign</a>	Create, update, delete, and query Campaigns.
<a href="#">CampaignResponses</a>	Create, delete, and query Campaign Responses. This table is only supported in bulk mode.
<a href="#">Contact</a>	Create, update, delete, and query Contacts. This table is supported in bulk mode.
<a href="#">ContactEmailSubscription</a>	Retrieve all email group subscription status for the specific contact.
<a href="#">ContactSegment</a>	Create, update, delete, and query Contact Segments.
<a href="#">ContentSection</a>	Create, update, delete, and query Content Sections.
<a href="#">Custom</a>	Create and delete custom.

Email	Create, update, delete, and query Emails.
EmailFooter	Create, update, delete, and query Email Footers.
EmailGroup	Create, update, delete, and query Email Groups.
EmailHeader	Create, update, delete, and query Email Headers.
Event	Create and delete Event.
ExternalActivity	Create and query External Activities. This table is <b>not</b> supported in bulk mode.
ExternalAsset	Create, update, delete, and query External Assets.
Folder	Create, update, delete, and query Folders.
Form	Create, update, delete, and query Forms.
Hyperlink	Create, update, delete, and query Hyperlinks.
LandingPage	Create, update, delete, and query Landing Pages.
Microsite	Create, update, delete, and query Microsites.
OptionList	Create, update, delete, and query Option Lists.

The adapter can expose custom tables, views, and columns from Oracle Eloqua that are not mentioned in the [Tables](#) and [Views](#). The data model illustrates a sample of what your Oracle Eloqua data model might look like. The actual data model will be obtained dynamically based on your user credentials and Oracle Eloqua account.

## Account

Create, update, delete, and query Accounts. This table is supported in bulk mode.

## Columns

<b>Name</b>	<b>Type</b>	<b>ReadOnly</b>	<b>Description</b>
ID [KEY]	<i>Long</i>	True	The unique identifier of the account.
Company Name	<i>String</i>	False	The Name of the account.
Country	<i>String</i>	False	The country of the account.
Address	<i>String</i>	False	The address of the account.
City	<i>String</i>	False	The city of the address for the account.
State or Province	<i>String</i>	False	The state or province of the address for the account.
Zip or Postal Code	<i>String</i>	False	The zip or postal code of the address for the account.
Business Phone	<i>String</i>	False	The business phone number of the account.
Eloqua Company ID	<i>String</i>	True	The Eloqua company Id of the account
Date Created	<i>Datetime</i>	True	Date and time the account was created.
Date Modified	<i>Datetime</i>	True	Date and time the account was last updated.
Company Category	<i>Double</i>	False	The category of the company
SFDC Account ID	<i>String</i>	False	The Salesforce Id of the

			account
Last Modified by CRM System	<i>Datetime</i>	True	Date and time the account was last updated by the CRM system.
Address 2	<i>String</i>	False	The second address of the account.
Address 3	<i>String</i>	False	The third address of the account.
Industry	<i>String</i>	False	The industry of the account.
PURL Name (Default Hypersite)	<i>String</i>	True	The URL of the account.
Annual Revenue	<i>String</i>	False	The annual revenue of the account.
Fax	<i>String</i>	False	The fax number of the account.
Employees	<i>Double</i>	False	The number of employees of the account
Website	<i>String</i>	False	The website of the account.
Account Rating	<i>String</i>	False	The account rating of the account.

## AccountGroup

Retrieve Account group.

## Columns

<b>Name</b>	<b>Type</b>	<b>ReadOnly</b>	<b>Description</b>
ID [KEY]	<i>Long</i>	True	The unique identifier for the account group.
Name	<i>String</i>	False	The name of the account group.
Type	<i>String</i>	False	The type of the account group.
Permissions	<i>String</i>	False	The permissions of the account group.
CreatedAt	<i>Datetime</i>	True	Date the account group was created.
IsArchived	<i>String</i>	False	The Id of the user who created the account group.
FolderId	<i>String</i>	True	The folderId of the account group.
UpdatedAt	<i>Datetime</i>	True	Date the account group was updated.
Description	<i>String</i>	True	The description of the accountgroup.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

<b>Name</b>	<b>Type</b>	<b>Description</b>
-------------	-------------	--------------------

Depth	<i>String</i>	The depth of information requested from Eloqua.
-------	---------------	---

## Campaign

Create, update, delete, and query Campaigns.

### Columns

Name	Type	ReadOnly	Description
ID [KEY]	<i>Long</i>	True	The unique identifier of the campaign.
Name	<i>String</i>	False	The name of the campaign.
ActualCost	<i>Double</i>	False	The actual cost of the campaign.
BudgetedCost	<i>Double</i>	False	The budgeted cost for the Campaign.
CrmlId	<i>String</i>	False	The unique Id of the campaign in your CRM.
EndAt	<i>Datetime</i>	False	The campaign's end date.
EndValues	<i>String</i>	False	The end values of the campaign.
IsMemberAllowedReEntry	<i>Boolean</i>	False	Determines whether a member can reenter the campaign.
IsReadOnly	<i>Boolean</i>	False	Whether the campaign is read-

			only.
StartAt	<i>Datetime</i>	False	The campaign's start date.
CreatedAt	<i>Datetime</i>	False	The date and time the campaign was created.
CreatedBy	<i>String</i>	False	The Id of the user who created the Campaign.
AccessedAt	<i>Datetime</i>	False	Date the campaign was accessed.
CurrentStatus	<i>String</i>	False	The current status of the campaign.
Depth	<i>String</i>	False	The depth of the campaign.
UpdatedAt	<i>Datetime</i>	False	Date the campaign was updated.
UpdatedBy	<i>String</i>	False	The Id of the user who last updated the Campaign.
Permissions	<i>String</i>	False	The permissions of the campaign.
ScheduledFor	<i>Datetime</i>	False	The date and time scheduled for the campaign.
SourceTemplatedId	<i>String</i>	False	The Id of the source template for the campaign.
FolderId	<i>Long</i>	False	The Id of the folder for the campaign.
Campaign Type	<i>String</i>	False	The type of campaign.

Product	<i>String</i>	False	The product of the campaign.
Region	<i>String</i>	False	The region of the campaign.
Five	<i>String</i>	False	The fifth value for the campaign.
Four	<i>String</i>	False	The fourth value for the campaign.
Three	<i>String</i>	False	The third value for the campaign.
Two	<i>String</i>	False	The second value for the campaign.
CLR End Date	<i>Datetime</i>	False	The CLR end date.

## CampaignResponses

Create, delete, and query Campaign Responses. This table is only supported in bulk mode.

### Columns

<b>Name</b>	<b>Type</b>	<b>ReadOnly</b>	<b>Description</b>
ID	<i>Long</i>	False	The unique identifier for the campaign responses.
Member Status	<i>String</i>	False	The status of the campaign member.
Integration Return Value	<i>String</i>	False	The return value of the campaign responses.



Created At	<i>Long</i>	False	The created time of the campaign responses.
Add At	<i>String</i>	False	The added time of the campaign responses.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
ServiceInstanceId	<i>String</i>	The AppCloud action service instance ID. It is necessary to insert or select this table
ExecutionId	<i>String</i>	The execution ID of service instance. It is necessary to select this table

## Contact

Create, update, delete, and query Contacts. This table is supported in bulk mode.

### Table Specific Information

The EmailAddress column is required for inserts and updates.

### Columns

Name	Type	ReadOnly	Description
ID	<i>Long</i>	True	The unique

[KEY]			identifier of the contact.
Email Address [KEY]	<i>String</i>	False	The Email Address of the contact.
First Name	<i>String</i>	False	The First Name of the contact.
Last Name	<i>String</i>	False	The Last Name of the contact.
Company	<i>String</i>	False	The Company of the contact.
Email Display Name	<i>String</i>	False	The Email Display Name of the contact.
Address 1	<i>String</i>	False	The Address 1 of the contact.
Address 2	<i>String</i>	False	The Address 2 of the contact.
Address 3	<i>String</i>	False	The Address 3 of the contact.
City	<i>String</i>	False	The City of the contact.
State or Province	<i>String</i>	False	The State or Province of the contact.
Zip or Postal Code	<i>String</i>	False	The Zip or Postal Code of the contact.

Country	<i>String</i>	False	The Country of the contact.
Business Phone	<i>String</i>	False	The Business Phone of the contact.
Mobile Phone	<i>String</i>	False	The Mobile Phone of the contact.
Fax	<i>String</i>	False	The Fax of the contact.
Title	<i>String</i>	False	The Title of the contact.
Salutation	<i>String</i>	False	The Salutation of the contact.
Salesperson	<i>String</i>	False	The Salesperson of the contact.
SFDC Contact ID	<i>String</i>	False	The SFDC Contact ID of the contact.
SFDC Lead ID	<i>String</i>	False	The SFDC Lead ID of the contact.
Date Created	<i>Date</i>	True	The Date Created of the contact.
Date Modified	<i>Date</i>	True	The Date Modified of the contact.
Eloqua Contact ID	<i>String</i>	True	The Eloqua Contact ID of the contact.
SFDC Account ID	<i>String</i>	False	The SFDC Account ID of the contact.

Last Modified by CRM System	<i>Date</i>	True	The Last Modified by CRM System of the contact.
Last SFDC Campaign ID	<i>String</i>	False	The Last SFDC Campaign ID of the contact.
Last SFDC Campaign Status	<i>String</i>	False	The Last SFDC Campaign Status of the contact.
Company Revenue	<i>String</i>	False	The Company Revenue of the contact.
SFDC Email Opt Out	<i>String</i>	False	The SFDC Email Opt Out of the contact.
Lead Source - Most Recent	<i>String</i>	False	The Lead Source - Most Recent of the contact.
Lead Source - Original	<i>String</i>	False	The Lead Source - Original of the contact.
Industry	<i>String</i>	False	The Industry of the contact.
Annual Revenue	<i>String</i>	False	The Annual Revenue of the contact.
Lead Status	<i>String</i>	False	The Lead Status of the contact.
Job Role	<i>String</i>	False	The Job Role of the contact.

Lead Score - High Value Website Content	<i>Double</i>	False	The Lead Score - High Value Website Content of the contact.
Lead Score Date - Engagement - Most Recent	<i>Date</i>	False	The Lead Score Date - Engagement - Most Recent of the contact.
Integrated Marketing and Sales Funnel Stage	<i>String</i>	False	The Integrated Marketing and Sales Funnel Stage of the contact.
Product/Solution of Interest	<i>String</i>	False	The Product/Solution of Interest of the contact.
Region	<i>String</i>	False	The Region of the contact.
PURL Name (Default Hypersite)	<i>String</i>	True	The PURL Name (Default Hypersite) of the contact.
Lead Rating - Combined	<i>String</i>	False	The Lead Rating - Combined of the contact.
Email Address Domain	<i>String</i>	True	The Email Address Domain of the contact.
First and Last Name	<i>String</i>	True	The First and Last Name of the contact.
Company Size	<i>String</i>	False	The Company Size

			of the contact.
Lead Score - Last High Touch Event Date	<i>Date</i>	False	The Lead Score - Last High Touch Event Date of the contact.
Lead Rating - Profile (Explicit)	<i>String</i>	False	The Lead Rating - Profile (Explicit) of the contact.
Lead Rating - Engagement (Implicit)	<i>String</i>	False	The Lead Rating - Engagement (Implicit) of the contact.
Lead Score - Profile (Explicit)	<i>Double</i>	False	The Lead Score - Profile (Explicit) of the contact.
Lead Score - Engagement (Implicit)	<i>Double</i>	False	The Lead Score - Engagement (Implicit) of the contact.
Lead Score Date - Profile - Most Recent	<i>Date</i>	False	The Lead Score Date - Profile - Most Recent of the contact.
Employees	<i>Double</i>	False	The Employees of the contact.
Territory	<i>String</i>	False	The Territory of the contact.
Lead Score	<i>String</i>	False	The Lead Score of the contact.
PURL Name	<i>String</i>	True	The PURL Name of

			the contact.
MD5 Hashed Email Address	<i>String</i>	True	The MD5 Hashed Email Address of the contact.
SHA256 Hashed Email Address	<i>String</i>	True	The SHA256 Hashed Email Address of the contact.
MD5 Hashed Business Phone	<i>String</i>	True	The MD5 Hashed Business Phone of the contact.
SHA256 Hashed Business Phone	<i>String</i>	True	The SHA256 Hashed Business Phone of the contact.
MD5 Hashed Mobile Phone	<i>String</i>	True	The MD5 Hashed Mobile Phone of the contact.
SHA256 Hashed Mobile Phone	<i>String</i>	True	The SHA256 Hashed Mobile Phone of the contact of the contact.

## ContactEmailSubscription

Retrieve all email group subscription status for the specific contact.

### Columns

Name	Type	ReadOnly	Description
------	------	----------	-------------

ID [KEY]	<i>Long</i>	True	The unique identifier for Email Group.
ContactID [KEY]	<i>Long</i>	False	The unique identifier for the contact.
Name	<i>String</i>	True	The name of the email group.
Description	<i>String</i>	True	The description of the email group.
IsSubScribed	<i>Boolean</i>	False	The subscribed status. The default value is <i>false</i> .
Permissions	<i>String</i>	True	The permissions of the email group.

## ContactSegment

Create, update, delete, and query Contact Segments.

### Columns

Name	Type	ReadOnly	Description
ID [KEY]	<i>Long</i>	True	The unique identifier for the contact segment.
Name	<i>String</i>	False	The name of the contact segment.
Description	<i>String</i>	False	The description of the contact



			segment.
FolderId	<i>Long</i>	False	The Id of the folder for the contact segment.
ScheduledFor	<i>Datetime</i>	False	The date and time scheduled for the contact segment.
SourceTemplatedId	<i>Long</i>	False	The Id of the source template of the contact segment.
Permissions	<i>String</i>	False	The permissions of the contact segment.
CreatedAt	<i>Datetime</i>	True	Date the contact segment was created.
CreatedBy	<i>String</i>	True	The Id of the user who created the contact segment.
AccessedAt	<i>Datetime</i>	False	Date the contact segment was accessed.
CurrentStatus	<i>String</i>	False	The current status of the contact segment.
UpdatedAt	<i>Datetime</i>	True	Date the contact segment was last updated.
UpdatedBy	<i>String</i>	True	The Id of the user who last updated the contact segment.
LastCalculatedAt	<i>Datetime</i>	False	Date the contact segment was last calculated.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
Depth	<i>String</i>	The depth of information requested from Eloqua.

## ContentSection

Create, update, delete, and query Content Sections.

### Columns

Name	Type	ReadOnly	Description
ID [KEY]	<i>Long</i>	True	The unique identifier for the content section.
Name	<i>String</i>	False	The name of the content section.
Description	<i>String</i>	False	The description of the content section.
FolderId	<i>Long</i>	False	The Id of the folder for the content section.
ScheduledFor	<i>Datetime</i>	False	The date and time scheduled for the content section.
SourceTemplatedId	<i>Long</i>	False	The Id of the source template of the content section.
Permissions	<i>String</i>	False	The permissions of the content section.

CreatedAt	<i>Datetime</i>	True	Date the content section was created.
CreatedBy	<i>String</i>	True	The Id of the user who created the content section.
AccessedAt	<i>Datetime</i>	False	Date the content section was accessed.
CurrentStatus	<i>String</i>	False	The current status of the content section.
UpdatedAt	<i>Datetime</i>	True	Date the content section was updated.
UpdatedBy	<i>String</i>	True	The Id of the user who last updated the content section.
ContentHtml	<i>String</i>	False	The content HTML of the content section.
ContentText	<i>String</i>	False	The content text of the data lookup for the content section.
Scope	<i>String</i>	False	A list of contact Ids to be removed from the content section.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

<b>Name</b>	<b>Type</b>	<b>Description</b>
Depth	<i>String</i>	The depth of information requested from Eloqua.

## Custom

Create and delete custom.

### Columns

Name	Type	ReadOnly	Description
Id [KEY]	<i>String</i>	False	The id of the custom.
Name	<i>String</i>	False	The name of the custom.
EmailAddressFieldId	<i>String</i>	False	The field id that contains the Email Address. Use the desired custom object field's negative id as the value for this parameter.
EventGroupByFieldId	<i>String</i>	False	The id of the field used to organize multiple sessions. Use the desired custom object field's negative id as the value for this parameter.
FolderId	<i>String</i>	True	The folder id of the folder which contains the custom.
Fields	<i>String</i>	False	Array consisting of custom field properties
Description	<i>String</i>	False	The description of the custom.
RecordCount	<i>Integer</i>	True	The amount of records within the custom.
CreatedAt	<i>Datetime</i>	True	Date the custom was created.

---

CreatedBy	<i>String</i>	True	The Id of the user who created the custom.
UpdatedAt	<i>Datetime</i>	True	Date the custom was last updated.
UpdatedBy	<i>String</i>	True	The Id of the user who last updated the custom.

---

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

---

<b>Name</b>	<b>Type</b>	<b>Description</b>
Depth	<i>String</i>	The depth of information requested from Eloqua.

---

## Email

Create, update, delete, and query Emails.

## Table Specific Information

### Select

The following limitation exists for filtering on datetime columns: The "=" operator cannot be used with datetime columns in the WHERE clause. However, the "<", ">", ">=", and "<=" operators can be used to filter on datetime columns.

Additionally, the UpdatedAt column only supports filtering with the '>' operator.

---

Oracle Eloqua also has limited support on multiple conditions. In many tables the AND condition between filters is not supported. If AND is included the server might not return any results, nor will it return an error.

## Columns

Name	Type	ReadOnly	Description
ID [KEY]	<i>Long</i>	True	The unique identifier of the email.
Name	<i>String</i>	False	The name of the email.
EmailGroupId	<i>Long</i>	False	The Id of the email group.
HtmlContent_ContentSource	<i>String</i>	False	The raw HTML contentSource for the content of the email.
HtmlContent_CssHeader	<i>String</i>	False	The raw HTML cssHeader for the content of the email.
HtmlContent_DocType	<i>String</i>	False	The raw HTML docType for the content of the email.
HtmlContent_DocumentDescription	<i>String</i>	False	The raw HTML documentDescription for the content of the email.
HtmlContent_HtmlBody	<i>String</i>	False	The raw HTML Body for the content of the email.

HtmlContent_MetaTags	<i>String</i>	True	The raw HTML metaTags for the content of the email.
HtmlContent_Root	<i>String</i>	False	The raw HTML root for the content of the email.
HtmlContent_SystemHeader	<i>String</i>	False	The raw HTML systemHeader for the content of the email.
HtmlContent_Type	<i>String</i>	False	The raw HTML type for the content of the email.
IsPlainTextEditable	<i>Boolean</i>	False	Identifies whether the email is plaintext editable.
IsTracked	<i>Boolean</i>	False	Identifies whether the email is tracked.
PlainText	<i>String</i>	False	The plaintext version of the email.
BouncebackEmail	<i>String</i>	False	The bounce-back email address.
ReplyToEmail	<i>String</i>	False	The email address in the reply-to line of the email.
ReplyToName	<i>String</i>	False	The name in the reply-to line of the email.
SenderEmail	<i>String</i>	False	The email address of the sender.
SenderName	<i>String</i>	False	The name of the

			sender.
SenderPlainTextOnly	<i>Boolean</i>	False	Identifies whether the plaintext version should be sent.
Subject	<i>String</i>	False	The subject of the email.
Description	<i>String</i>	False	The description of the email.
FolderId	<i>Long</i>	False	The Id of the folder for the email.
EmailHeaderId	<i>Long</i>	False	The Id of the email header.
EmailFooterId	<i>Long</i>	False	The Id of the email footer.
Permissions	<i>String</i>	False	The permissions of the email.
CreatedAt	<i>Datetime</i>	True	Date the email was created.
CreatedBy	<i>String</i>	True	The Id of the user who created the email.
AccessedAt	<i>Datetime</i>	False	Date the email was accessed.
CurrentStatus	<i>String</i>	False	The current status of the email.
Depth	<i>String</i>	False	The depth of the email.
UpdatedAt	<i>Datetime</i>	True	Date the email was



			updated.
UpdatedBy	<i>String</i>	True	The Id of the user who last updated the email.

## EmailFooter

Create, update, delete, and query Email Footers.

### Columns

Name	Type	ReadOnly	Description
ID [KEY]	<i>Long</i>	True	The unique identifier for the email footer.
Name	<i>String</i>	False	The name of the email footer.
Body	<i>String</i>	False	The body of the email footer.
Hyperlinks	<i>String</i>	True	A list of hyperlinks.
FieldMerges	<i>String</i>	True	A list of field merges.
Text	<i>String</i>	False	The text of the email footer.
Permissions	<i>String</i>	False	The permissions of the email footer.
Description	<i>String</i>	False	The description of the email footer.

FolderId	<i>Long</i>	False	The Id of the folder for the email footer.
CreatedAt	<i>Datetime</i>	True	Date the email footer was created.
CreatedBy	<i>String</i>	True	The Id of the user who created the email footer.
AccessedAt	<i>Datetime</i>	False	Date the email footer was accessed.
CurrentStatus	<i>String</i>	False	The current status of the email footer.
Depth	<i>String</i>	False	The depth of the email footer.
UpdatedAt	<i>Datetime</i>	True	Date the email footer was updated.
UpdatedBy	<i>String</i>	True	The Id of the user who last updated the email footer.
PlainText	<i>String</i>	True	The plain text of the email footer.

## EmailGroup

Create, update, delete, and query Email Groups.

### Table Specific Information

#### Select

The following limitation exists for filtering on datetime columns: The "=" operator cannot be used with datetime columns in the WHERE clause. However, the "<", ">", ">=", and "<=" operators can be used to filter on datetime columns.

Additionally, the UpdatedAt column only supports filtering with the '>' operator.

Oracle Eloqua also has limited support on multiple conditions. In many tables the AND condition between filters is not supported. If AND is included the server might not return any results, nor will it return an error.

## Columns

Name	Type	ReadOnly	Description
ID [KEY]	Long	True	The unique identifier of the email group.
Name	String	False	The name of the email.
DisplayName	String	False	The display name of the email group.
EmailFooterId	Long	False	The Id of the email footer of the email group.
EmailIds	String	False	The Ids of the emails in the email group.
IsVisibleInOutlookPlugin	Boolean	False	Identifies whether the email group is visible in the Outlook plugin.
IsVisibleInPublicSubscriptionList	Boolean	False	Identifies whether the email group is visible in public subscription list.
SubscriptionLandingPageId	Long	False	The Id of the landing page used to subscribe to the email group.
SubscriptionListId	Long	False	The Id of the subscription

			list for the email group.
UnsubscriptionLandingPageId	<i>Long</i>	False	The Id of the landing page used to unsubscribe for the email group.
UnsubscriptionListId	<i>Long</i>	False	The Id of the unsubscription list for the email group.
UnsubscriptionListDataLookupId	<i>String</i>	False	The Id of the data lookup for the unsubscription list for the email group.
Description	<i>String</i>	False	The description of the email group.
FolderId	<i>Long</i>	False	The folder Id for the email group.
Permissions	<i>String</i>	False	The permissions list of the email group.
CreatedAt	<i>Datetime</i>	True	Date the email group was created.
CreatedBy	<i>String</i>	True	The Id of the user who created the email group.
AccessedAt	<i>Datetime</i>	False	Date the email group was accessed.
CurrentStatus	<i>String</i>	False	The current status of the email group.
Depth	<i>String</i>	False	The depth of the email group.
UpdatedAt	<i>Datetime</i>	True	Date the email group was updated.

UpdatedBy	<i>String</i>	True	The Id of the user who last updated the email group.
-----------	---------------	------	--

## EmailHeader

Create, update, delete, and query Email Headers.

### Table Specific Information

#### Select

The following limitation exists for filtering on datetime columns: The "=" operator cannot be used with datetime columns in the WHERE clause. However, the "<", ">", ">=", and "<=" operators can be used to filter on datetime columns.

Additionally, the UpdatedAt column only supports filtering with the '>' operator.

Oracle Eloqua also has limited support on multiple conditions. In many tables the AND condition between filters is not supported. If AND is included the server might not return any results, nor will it return an error.

#### Columns

Name	Type	ReadOnly	Description
ID [KEY]	<i>Long</i>	True	The unique identifier for the email header.
Name	<i>String</i>	False	The name of the email header.
Body	<i>String</i>	False	The body of the email header.

Hyperlinks	<i>String</i>	True	A list of hyperlinks.
FieldMerges	<i>String</i>	True	A list of field merges.
Text	<i>String</i>	False	The text of the email header.
Permissions	<i>String</i>	False	The permissions of the email header.
Description	<i>String</i>	False	The description of the email header.
FolderId	<i>Long</i>	False	The Id of the folder for the email header.
CreatedAt	<i>Datetime</i>	True	Date the email header was created.
CreatedBy	<i>String</i>	True	The Id of the user who created the email header.
AccessedAt	<i>Datetime</i>	False	Date the email header was accessed.
CurrentStatus	<i>String</i>	False	The current status of the email header.
Depth	<i>String</i>	False	The depth of the email header.
UpdatedAt	<i>Datetime</i>	True	Date the email header was updated.
UpdatedBy	<i>String</i>	True	The Id of the user who last updated the email header.

## Event

Create and delete Event.

## Columns

Name	Type	ReadOnly	Description
Id [KEY]	<i>String</i>	False	The Id of the event.
Name	<i>String</i>	False	The name of the event.
EmailAddressFieldId	<i>String</i>	False	The field Id that contains the Email Address. Use the negative id of the desired custom object field as the value for this parameter.
EventGroupByFieldId	<i>String</i>	False	The Id of the field used to organize multiple sessions. Use the negative Id of the desired custom object field as the value for this parameter.
FolderId	<i>String</i>	True	The folder Id of the folder which contains the event.
Fields	<i>String</i>	False	Array consisting of event field properties.
Description	<i>String</i>	False	The description of the event.
RecordCount	<i>Integer</i>	True	The amount of records within the event.
CreatedAt	<i>Datetime</i>	True	Date the event was created.

CreatedBy	<i>String</i>	True	The Id of the user who created the event.
UpdatedAt	<i>Datetime</i>	True	Date the event was last updated.
UpdatedBy	<i>String</i>	True	The Id of the user who last updated the event.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
Depth	<i>String</i>	The depth of information requested from Eloqua.

## ExternalActivity

Create and query External Activities. This table is **not** supported in bulk mode.

### Table Specific Information

#### Select

This table requires that either **Id** or **ContactId** be specified to retrieve data:

```
SELECT * FROM ExternalActivity WHERE Id= '1111'
SELECT * FROM ExternalActivity WHERE ContactId = '11'
```



## Columns

Name	Type	ReadOnly	Description
ID [KEY]	<i>Long</i>	True	The unique identifier of the external activity.
Type	<i>String</i>	False	The type of the external activity.
Name	<i>String</i>	False	The Name of the external activity.
ActivityType	<i>String</i>	False	The Type of external activity.
ContactId	<i>Long</i>	False	The Id of the contact who performed the external activity.
CampaignId	<i>Long</i>	False	The Id of the campaign associated with this external activity.
AssetName	<i>String</i>	False	The AssetName of the external activity.
AssetType	<i>String</i>	False	The AssetType of the external activity.
AssetId	<i>Long</i>	False	The AssetId of the external activity.
ActivityDate	<i>Datetime</i>	False	The Date of the external activity.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
Depth	<i>String</i>	The depth of information requested from Eloqua.

## ExternalAsset

Create, update, delete, and query External Assets.

### Table Specific Information

#### Select

The following limitation exists for filtering on datetime columns: The "=" operator cannot be used with datetime columns in the WHERE clause. However, the "<", ">", ">=", and "<=" operators can be used to filter on datetime columns.

Additionally, the UpdatedAt column only supports filtering with the '>' operator.

Oracle Eloqua also has limited support on multiple conditions. In many tables the AND condition between filters is not supported. If AND is included the server might not return any results, nor will it return an error.

#### Columns

Name	Type	ReadOnly	Description
ID [KEY]	<i>Long</i>	True	The unique identifier of the external asset.

Name	<i>String</i>	False	The name of the external asset.
CreatedAt	<i>Datetime</i>	True	Date the external asset was created.
CreatedBy	<i>String</i>	True	The Id of the user who created the external asset.
Depth	<i>String</i>	False	The depth of the external asset.
UpdatedAt	<i>Datetime</i>	True	Date the external asset was updated.
UpdatedBy	<i>String</i>	True	The Id of the user who last updated the external asset.

## Folder

Create, update, delete, and query Folders.

### Columns

<b>Name</b>	<b>Type</b>	<b>ReadOnly</b>	<b>Description</b>
ID [KEY]	<i>Long</i>	True	The unique identifier for the folder.
Name	<i>String</i>	False	The name of the folder.
IsSystem	<i>Boolean</i>	False	Identifies a system-generated folder.
Description	<i>String</i>	False	The description of the folder.

CreatedAt	<i>Datetime</i>	True	Date the folder was created.
UpdatedAt	<i>Datetime</i>	True	Date the folder was last updated.
UpdatedBy	<i>String</i>	True	The Id of the user who last updated the folder.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
Depth	<i>String</i>	The depth of information requested from Eloqua.

## Form

Create, update, delete, and query Forms.

## Columns

Name	Type	ReadOnly	Description
ID [KEY]	<i>Long</i>	True	The unique identifier for the form.

Name	<i>String</i>	False	The name of the form.
FolderId	<i>Long</i>	False	The Id of the folder for the form.
SourceTemplatedId	<i>Long</i>	False	The Id of the source template of the form.
Permissions	<i>String</i>	False	The permissions of the form.
CreatedAt	<i>Datetime</i>	True	Date the form was created.
CreatedBy	<i>String</i>	True	The Id of the user who created the form.
CurrentStatus	<i>String</i>	False	The current status of the form.
UpdatedAt	<i>Datetime</i>	True	Date the form was last updated.
UpdatedBy	<i>String</i>	True	The Id of the user who last updated the form.
HtmlName	<i>String</i>	False	The HTML name of the form.
ProcessingType	<i>String</i>	False	The processing type of the data lookup for the form.
SubmitFailedLandingPageId	<i>Long</i>	False	The submit failed landing page Id of the form.
Html	<i>String</i>	False	The HTML of the form.
Style	<i>String</i>	False	The style of the form.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
Depth	<i>String</i>	The depth of information requested from Eloqua.

## Hyperlink

Create, update, delete, and query Hyperlinks.

## Columns

Name	Type	ReadOnly	Description
ID [KEY]	<i>Long</i>	True	The unique identifier for the hyperlink.
Name	<i>String</i>	False	The name of the hyperlink.
IsSystem	<i>Boolean</i>	False	Identifies a system-generated hyperlink.
UpdatedAt	<i>Datetime</i>	True	Date the hyperlink was updated.
UpdatedBy	<i>String</i>	True	The Id of the user who last updated the hyperlink.

FolderId	<i>Long</i>	False	The folder Id of the hyperlink.
CreatedAt	<i>Datetime</i>	True	Date the hyperlink was created.
CreatedBy	<i>String</i>	True	The Id of the user who created the hyperlink.
Href	<i>String</i>	False	The href of the hyperlink.
HyperlinkType	<i>String</i>	False	The type of the hyperlink.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

<b>Name</b>	<b>Type</b>	<b>Description</b>
Depth	<i>String</i>	The depth of information requested from Eloqua.

## LandingPage

Create, update, delete, and query Landing Pages.

## Table Specific Information

## Select

The following limitation exists for filtering on datetime columns: The "=" operator cannot be used with datetime columns in the WHERE clause. However, the "<", ">", ">=", and "<=" operators can be used to filter on datetime columns.

Additionally, the UpdatedAt column only supports filtering with the '>' operator.

Oracle Eloqua also has limited support on multiple conditions. In many tables the AND condition between filters is not supported. If AND is included the server might not return any results, nor will it return an error.

## Columns

Name	Type	ReadOnly	Description
ID [KEY]	Long	True	The unique identifier for the landing page.
Name	String	False	The name of the landing page.
DeployedAt	Datetime	False	Date the landing page was deployed.
HtmlContent	String	False	The raw HTML for the content of the landing page.
MicrositeId	Long	False	The unique identifier for this page's microsite.
RefreshedAt	Datetime	False	Date the landing page was refreshed.
RelativePath	String	False	The relative path to the landing page.
Style	String	False	The style of the landing page.



Description	<i>String</i>	False	The description of the landing page.
FolderId	<i>Long</i>	False	The Id of the folder for the landing page
Permissions	<i>String</i>	False	The permissions list for the landing page.
CreatedAt	<i>Datetime</i>	True	Date the landing page was created.
CreatedBy	<i>String</i>	True	The Id of the user who created the landing page.
AccessedAt	<i>Datetime</i>	False	Date the landing page was accessed.
CurrentStatus	<i>String</i>	False	The current status of the landing page.
AutoRedirectUrl	<i>String</i>	False	The URL to redirect window used with autoRedirectWaitFor.
Depth	<i>String</i>	False	The depth of the landing page.
UpdatedAt	<i>Datetime</i>	True	Date the landing page was updated.
UpdatedBy	<i>String</i>	True	The Id of the user who last updated the landing page.

## Microsite

Create, update, delete, and query Microsites.

### Table Specific Information

## Select

The following limitation exists for filtering on datetime columns: The "=" operator cannot be used with datetime columns in the WHERE clause. However, the "<", ">", ">=", and "<=" operators can be used to filter on datetime columns.

Additionally, the UpdatedAt column only supports filtering with the '>' operator.

Oracle Eloqua also has limited support on multiple conditions. In many tables the AND condition between filters is not supported. If AND is included the server might not return any results, nor will it return an error.

## Columns

Name	Type	ReadOnly	Description
ID [KEY]	Long	True	The unique identifier of the microsite.
Name	String	False	The name of the microsite.
Domains	String	False	A list of domains for this microsite.
ScheduledFor	Datetime	False	The date and time scheduled for the microsite.
SourceTemplatedId	Long	False	The Id of the source template for the microsite.
DefaultLandingPageId	Long	False	The Id of the default landing page for the microsite.
Description	String	False	The description of the microsite.
FolderId	Long	False	The Id of the folder for the microsite

Permissions	<i>String</i>	False	The permissions of the microsite.
CreatedAt	<i>Datetime</i>	True	Date the microsite was created.
CreatedBy	<i>String</i>	True	The Id of the user who created the microsite.
AccessedAt	<i>Datetime</i>	False	Date the microsite was accessed.
CurrentStatus	<i>String</i>	False	The current status of the microsite.
Depth	<i>String</i>	False	The depth of the microsite.
UpdatedAt	<i>Datetime</i>	True	Date the microsite was updated.
UpdatedBy	<i>String</i>	True	The Id of the user who last updated the microsite.

## OptionList

Create, update, delete, and query Option Lists.

### Table Specific Information

#### Select

The following limitation exists for filtering on datetime columns: The "=" operator cannot be used with datetime columns in the WHERE clause. However, the "<", ">", ">=", and "<=" operators can be used to filter on datetime columns.

Additionally, the UpdatedAt column only supports filtering with the '>' operator.

Oracle Eloqua also has limited support on multiple conditions. In many tables the AND condition between filters is not supported. If AND is included the server might not return any results, nor will it return an error.

## Columns

Name	Type	ReadOnly	Description
ID [KEY]	Long	True	The unique Id of the option list.
Name	String	False	The name of the option list item.
Depth	String	False	The depth of the option list item.
Permissions	String	False	The permissions of the option list item.

## Views

Views are composed of columns and pseudo columns. Views are similar to tables in the way that data is represented; however, views do not support updates. Entities that are represented as views are typically read-only entities. Often, a stored procedure is available to update the data if such functionality is applicable to the data source.

Queries can be executed against a view as if it were a normal table, and the data that comes back is similar in that regard.

Dynamic views, such as queries exposed as views, and views for looking up specific combinations of project\_team work items are supported.

## Oracle Eloqua Adapter Views

Name	Description
<a href="#">AccountField</a>	Retrieve Account Fields.
<a href="#">AccountView</a>	Retrieve Account Views.
<a href="#">Activity_Bounceback</a>	Retrieve Bounceback Activities. This table is only supported in bulk mode.
<a href="#">Activity_CampaignMembership</a>	Retrieve Campaign Membership Activities.
<a href="#">Activity_EmailClickThrough</a>	Retrieve Email Click Through Activities.
<a href="#">Activity_EmailOpen</a>	Retrieve Email Open Activities.
<a href="#">Activity_EmailSend</a>	Retrieve Email Send Activities.
<a href="#">Activity_EmailSubscribe</a>	Retrieve Email Subscribe Activities.
<a href="#">Activity_EmailUnsubscribe</a>	Retrieve Email Unsubscribe Activities.
<a href="#">Activity_FormSubmit</a>	Retrieve Form Submit Activities.
<a href="#">Activity_PageView</a>	Retrieve Page View Activities. This table is only supported in bulk mode.
<a href="#">Activity_WebVisit</a>	Retrieve Web Visit Activities.
<a href="#">CampaignElement</a>	Retrieve Campaign Elements.
<a href="#">CampaignField</a>	Retrieve Campaign Fields.
<a href="#">CampaignFolder</a>	Retrieve Campaign Folders.
<a href="#">ContactField</a>	Retrieve Contact Fields.
<a href="#">ContactFilter</a>	Retrieve Contact Filters.
<a href="#">ContactFilterFolder</a>	Retrieve Contact Filter Folders.

<a href="#">ContactList</a>	Create, update, delete, and query Contact Lists.
<a href="#">ContactListFolder</a>	Retrieve Contact List Folders.
<a href="#">ContactScoringModelFolder</a>	Retrieve Contact Scoring Model Folders.
<a href="#">ContactSegmentData</a>	Retrieve Contact Segment Data. Note: The pseudo column SegmentID must be specified to retrieve the data of this table. This SegmentID can be retrieved from the ContactSegment table's ID column.
<a href="#">ContactSegmentFolder</a>	Retrieve Contact Segment Folders.
<a href="#">ContactView</a>	Retrieve Contact Views.
<a href="#">ContentSectionFolder</a>	Retrieve Content Section Folders.
<a href="#">Dependencies</a>	Retrieve Dependencies.
<a href="#">DynamicContent</a>	Retrieve Dynamic Contents.
<a href="#">DynamicContentFolder</a>	Retrieve Dynamic Content Folders.
<a href="#">EmailDeployment</a>	Retrieve email deployments.
<a href="#">EmailFolder</a>	Retrieve Email Folders.
<a href="#">EmailFooterFolder</a>	Retrieve Email Footer Folders.
<a href="#">EmailHeaderFolder</a>	Retrieve Email Header Folders.
<a href="#">ExternalType</a>	Retrieve External Types.
<a href="#">FieldMerge</a>	Retrieve Field Merges.
<a href="#">FieldMergeFolder</a>	Retrieve Field Merge Folders.
<a href="#">FormElement</a>	Retrieve Form Elements.
<a href="#">FormFolder</a>	Retrieve Form Folders.

<a href="#">FormProcessingStep</a>	Retrieve Form Processing Steps.
<a href="#">HyperlinkFolder</a>	Retrieve Hyperlink Folders.
<a href="#">Image</a>	Retrieve Image.
<a href="#">ImageFolder</a>	Retrieve Image Folders.
<a href="#">ImportedFile</a>	Retrieve Imported Files.
<a href="#">ImportedFileFolder</a>	Retrieve Imported File Folders.
<a href="#">LandingPageFolder</a>	Retrieve Landing Page Folders.
<a href="#">LeadScoringModel</a>	Query Lead Scoring Models.
<a href="#">PageTag</a>	Retrieve Page Tags.
<a href="#">PageTagGroup</a>	Retrieve Page Tag Groups.
<a href="#">Program</a>	Retrieve Programs.
<a href="#">Style</a>	Retrieve Styles.
<a href="#">Template</a>	Retrieve Templates.
<a href="#">TemplateCategory</a>	Retrieve Template Categories.
<a href="#">TrackedUrl</a>	Retrieve Tracked Urls.
<a href="#">User</a>	Retrieve Users.
<a href="#">Visitor</a>	Retrieve Visitors.
<a href="#">VisitorProfileField</a>	Retrieve all of the visitor profile fields.

## AccountField

Retrieve Account Fields.

## Columns

Name	Type	Description
ID [KEY]	Long	The unique identifier for the account field.
Name	String	The name of the account field.
DataType	String	The data type of the account field.
DisplayType	String	The display type of the account field.
InternalName	String	The country of the account field.
IsReadOnly	Boolean	Identifies whether the account field is read-only.
IsRequired	Boolean	Identifies whether the account field is required.
IsStandard	Boolean	Identifies whether the account field is standard.
OptionListId	Long	The Id of the option list for the account field.
OutputFormatId	Long	The Id of the output format for the account field.
CreatedAt	Datetime	Date the account field was created.
CreatedBy	String	The Id of the user who created the account field.



Depth	<i>String</i>	The depth of the account field.
UpdatedAt	<i>Datetime</i>	Date the account field was updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the account field.

## AccountView

Retrieve Account Views.

### Columns

<b>Name</b>	<b>Type</b>	<b>Description</b>
ID [KEY]	<i>Long</i>	The unique identifier for the account view.
Name	<i>String</i>	The name of the account view.
Description	<i>String</i>	The description of the account view.
Permissions	<i>String</i>	The permissions of the account view.
CreatedAt	<i>Datetime</i>	Date the account view was created.
CreatedBy	<i>String</i>	The Id of the user who created the account view.
AccessedAt	<i>Datetime</i>	Date the account view was accessed.

---

UpdatedAt	<i>Datetime</i>	Date the account view was updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the account view.

---

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
Depth	<i>String</i>	The depth of information requested from Eloqua.

---

## Activity\_Bounceback

Retrieve Bounceback Activities. This table is only supported in bulk mode.

### Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier of the bounceback activity.
CreatedAt	<i>Datetime</i>	The date and time the bounceback activity was created.

---

Type	<i>String</i>	The type of the activity associated with the bounceback activity.
AssetName	<i>String</i>	The name of the asset associated with the bounceback activity.
AssetId	<i>Long</i>	The Id of the asset associated with the bounceback activity.
AssetType	<i>String</i>	The type of the asset associated with the bounceback activity.
CampaignId	<i>Long</i>	The Id of the campaign associated with the bounceback activity.
EmailAddress	<i>String</i>	The address of the email.
ExternalId	<i>String</i>	The ExternalId associated with the email activity. Only supported in bulk mode.
EmailRecipientId	<i>String</i>	The Email Recipient Id associated with the email activity. Only supported in bulk mode.
SmtpErrorCode	<i>String</i>	The Smtp Error Code associated with the email activity. Only supported in bulk mode.
SmtpStatusCode	<i>String</i>	The Smtp Status Code associated with the email activity. Only supported in bulk mode.
SmtpMessage	<i>String</i>	The Smtp Message associated with the email activity. Only supported in bulk mode.
DeploymentId	<i>String</i>	The email deployment Id associated with the email activity. Only supported in bulk mode.
ContactId	<i>Long</i>	The Id of the contact associated with the bounceback activity.
FirstName	<i>String</i>	The FirstName of the contact associated with the

		bounceback activity.
LastName	<i>String</i>	The LastName of the contact associated with the bounceback activity.
Company	<i>String</i>	The Company of the contact associated with the bounceback activity.
City	<i>String</i>	The City of the contact associated with the bounceback activity.
ContactCreatedAt	<i>Datetime</i>	The date and time the contact associated with the bounceback activity was created.
ContactModifiedAt	<i>Datetime</i>	The date and time the contact associated with the bounceback activity was created.
Address1	<i>String</i>	The first line address of the contact associated with the bounceback activity.
Address2	<i>String</i>	The second line address of the contact associated with the bounceback activity.
Title	<i>String</i>	The Title of the contact associated with the bounceback activity.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

<b>Name</b>	<b>Type</b>	<b>Description</b>
ExportToken	<i>String</i>	The token returned from CreateExportDefinition. If Eloqua has not finished processing the request, the query will wait until the server has finished before returning data.

---

Depth	<i>String</i>	The depth of information requested from Eloqua.
-------	---------------	---

---

## Activity\_CampaignMembership

Retrieve Campaign Membership Activities.

### Table Specific Information

This table is the only activity table not available in the Bulk API.

### Select

ContactId is required to query Activity tables.

### Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier of the campaign membership activity.
Name	<i>String</i>	The name of the campaign membership activity.
CreatedAt	<i>Datetime</i>	The date and time the campaign membership activity was created.
Type	<i>String</i>	The type of campaign membership activity.
AssetId	<i>Long</i>	The Id of the asset associated with the campaign membership activity.
AssetType	<i>String</i>	The type of the asset associated with the campaign

		membership activity.
AssetName	<i>String</i>	The name of the asset associated with the email campaign membership activity.
ContactId	<i>Long</i>	The Id of the contact who performed the campaign membership activity.
Responded	<i>String</i>	Whether the campaign membership activity has been responded to.
CampaignName	<i>String</i>	The campaign name of the campaign membership activity.
LeadStage	<i>String</i>	The lead stage of the campaign membership activity.
CampaignId	<i>Long</i>	The campaignId of the campaign membership activity.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
ExportToken	<i>String</i>	The token returned from CreateExportDefinition. If Eloqua has not finished processing the request, the query will wait until the server has finished before returning data.
Depth	<i>String</i>	The depth of information requested from Eloqua.

## Activity\_EmailClickThrough

Retrieve Email Click Through Activities.

## Table Specific Information

### Select

If you are not using the Bulk API, ContactId is required to query Activity tables.

Additionally, the COUNT function is supported for this table in bulk mode.

### Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier of the email click-through activity.
Name	<i>String</i>	The name of the email click-through activity. Not supported in bulk mode.
CreatedAt	<i>Datetime</i>	The date and time the email click-through activity was created.
Type	<i>String</i>	The type of the email click-through activity.
AssetId	<i>Long</i>	The Id of the asset associated with the email click-through activity.
AssetName	<i>String</i>	The name of the asset associated with the email click-through activity.
AssetType	<i>String</i>	The type of the asset associated with the email click-through activity.
ContactId	<i>Long</i>	The Id of the contact who performed the email click-through activity.

EmailClickedThruLink	<i>String</i>	The click-through link in the email.
EmailName	<i>String</i>	The name of the email. Not supported in bulk mode.
EmailWebLink	<i>String</i>	The Web link of the email.
SubjectLine	<i>String</i>	The subject line of the email.
EmailRecipientId	<i>String</i>	The Id of the recipient of the email.
EmailSendType	<i>String</i>	The send type of the email. Only supported in bulk mode.
EmailAddress	<i>String</i>	The address of the email. Only supported in bulk mode.
IpAddress	<i>String</i>	The IP address of the email activity. Only supported in bulk mode.
CampaignId	<i>Long</i>	The Id of the campaign associated with the email click-through activity. Only supported in bulk mode.
DeploymentId	<i>Long</i>	The deployment Id associated with the email open activity. Only supported in bulk mode.
VisitorId	<i>Long</i>	The Id of the visitor associated with the email open activity. Only supported in bulk mode.
VisitorExternalId	<i>String</i>	The ExternalId of the visitor associated with the email open activity. Only supported in bulk mode.
ExternalId	<i>String</i>	The ExternalId associated with the email open activity. Only supported in bulk mode.
FirstName	<i>String</i>	The FirstName of the contact associated with the



		email click-through activity.
LastName	<i>String</i>	The LastName of the contact associated with the email click-through activity.
Company	<i>String</i>	The Company of the contact associated with the email click-through activity.
City	<i>String</i>	The City of the contact associated with the email click-through activity.
ContactCreatedAt	<i>Datetime</i>	The date and time the contact associated with the email click-through activity was created.
ContactModifiedAt	<i>Datetime</i>	The date and time the contact associated with the email click-through activity was created.
Address1	<i>String</i>	The first line address of the contact associated with the email click-through activity.
Address2	<i>String</i>	The second line address of the contact associated with the email click-through activity.
Title	<i>String</i>	The Title of the contact associated with the email click-through activity.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

<b>Name</b>	<b>Type</b>	<b>Description</b>
ExportToken	<i>String</i>	The token returned from CreateExportDefinition. If Eloqua has not finished processing the request, the query will wait until the server has finished before returning data.

---

Depth	<i>String</i>	The depth of information requested from Eloqua.
-------	---------------	---

---

## Activity\_EmailOpen

Retrieve Email Open Activities.

### Table Specific Information

#### Select

If you are not using the Bulk API, ContactId is required to query Activity tables.

#### Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier of the email open activity.
Name	<i>String</i>	The name of the email open activity. Not supported in bulk mode.
EmailAddress	<i>String</i>	The email address of the contact.
CreatedAt	<i>Datetime</i>	The date and time the email open activity was created.
Type	<i>String</i>	The type of the activity associated with the email open activity.
AssetName	<i>String</i>	The name of the asset associated with the email open activity.

AssetId	<i>Long</i>	The Id of the asset associated with the email open activity.
AssetType	<i>String</i>	The type of the asset associated with the email open activity.
ContactId	<i>Long</i>	The Id of the contact who performed the email open activity.
EmailName	<i>String</i>	The name of the email. Not supported in bulk mode.
EmailWebLink	<i>String</i>	The Web link of the email.
EmailRecipientId	<i>String</i>	The Id of the recipient of the email.
EmailSendType	<i>String</i>	The send type of the email. Only supported in bulk mode.
IPAddress	<i>String</i>	The IP address of the sender.
SubjectLine	<i>String</i>	The subject line of the email.
CampaignId	<i>Long</i>	The Id of the campaign associated with the email open activity. Only supported in bulk mode.
DeploymentId	<i>Long</i>	The deployment Id associated with the email open activity. Only supported in bulk mode.
VisitorId	<i>Long</i>	The Id of the visitor associated with the email open activity. Only supported in bulk mode.
VisitorExternalId	<i>String</i>	The ExternalId of the visitor associated with the email open activity. Only supported in bulk mode.
ExternalId	<i>String</i>	The ExternalId associated with the email open activity.

Only supported in bulk mode.		
FirstName	<i>String</i>	The FirstName of the contact associated with the email open activity.
LastName	<i>String</i>	The LastName of the contact associated with the email open activity.
Company	<i>String</i>	The Company of the contact associated with the email open activity.
City	<i>String</i>	The City of the contact associated with the email open activity.
ContactCreatedAt	<i>Datetime</i>	The date and time the contact associated with the email open activity was created.
ContactModifiedAt	<i>Datetime</i>	The date and time the contact associated with the email open activity was created.
Address1	<i>String</i>	The first line address of the contact associated with the email open activity.
Address2	<i>String</i>	The second line address of the contact associated with the email open activity.
Title	<i>String</i>	The Title of the contact associated with the email open activity.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

<b>Name</b>	<b>Type</b>	<b>Description</b>
ExportToken	<i>String</i>	The token returned from CreateExportDefinition. If Eloqua has

---

not finished processing the request, the query will wait until the server has finished before returning data.

---

Depth	<i>String</i>	The depth of information requested from Eloqua.
-------	---------------	---

---

## Activity\_EmailSend

Retrieve Email Send Activities.

### Table Specific Information

#### Select

If you are not using the Bulk API, ContactId is required to query Activity tables.

Additionally, the COUNT function is supported for this table in bulk mode.

#### Columns

Name	Type	Description
ExternalId [KEY]	<i>String</i>	The ExternalId associated with the email send activity. Only supported in bulk mode.
ID	<i>Long</i>	The identifier of the email send activity. This is not unique.
Name	<i>String</i>	The name of the email send activity. Not supported in bulk mode.
CreatedAt	<i>Datetime</i>	The date and time when the email send activity was created.

---

Type	<i>String</i>	The activity type of the email send activity.
AssetName	<i>String</i>	The name of the asset associated with the email send activity.
AssetId	<i>Long</i>	The Id of the asset associated with the email send activity.
AssetType	<i>String</i>	The type of the asset associated with the email send activity.
ContactId	<i>Long</i>	The Id of the contact who performed the email send activity.
EmailName	<i>String</i>	The name of the email. Not supported in bulk mode.
EmailWebLink	<i>String</i>	The Web link of the email.
EmailRecipientId	<i>String</i>	The Id of the recipient of the email.
SubjectLine	<i>String</i>	The subject line of the email.
EmailAddress	<i>String</i>	The address of the email. Only supported in bulk mode.
EmailSendType	<i>String</i>	The send type of the email. Only supported in bulk mode.
CampaignId	<i>Long</i>	The Id of the campaign associated with the email send activity. Only supported in bulk mode.
DeploymentId	<i>Long</i>	The deployment Id associated with the email send activity. Only supported in bulk mode.
FirstName	<i>String</i>	The FirstName of the contact associated with the email

		send activity.
LastName	<i>String</i>	The LastName of the contact associated with the email send activity.
Company	<i>String</i>	The Company of the contact associated with the email send activity.
City	<i>String</i>	The City of the contact associated with the email send activity.
ContactCreatedAt	<i>Datetime</i>	The date and time the contact associated with the email send activity was created.
ContactModifiedAt	<i>Datetime</i>	The date and time the contact associated with the email send activity was created.
Address1	<i>String</i>	The first line address of the contact associated with the email send activity.
Address2	<i>String</i>	The second line address of the contact associated with the email send activity.
Title	<i>String</i>	The Title of the contact associated with the email send activity.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

<b>Name</b>	<b>Type</b>	<b>Description</b>
ExportToken	<i>String</i>	The token returned from CreateExportDefinition. If Eloqua has not finished processing the request, the query will wait until the server has finished before returning data.

---

Depth	<i>String</i>	The depth of information requested from Eloqua.
-------	---------------	---

---

## Activity\_EmailSubscribe

Retrieve Email Subscribe Activities.

### Table Specific Information

#### Select

If you are not using the Bulk API, ContactId is required to query Activity tables.

Additionally, the COUNT function is supported for this table in bulk mode.

#### Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier of the email subscribe activity.
Name	<i>String</i>	The name of the email subscribe activity. Not supported in bulk mode.
CreatedAt	<i>Datetime</i>	The date and time the email subscribe activity was created.
Type	<i>String</i>	The activity type of the email subscribe activity.
AssetName	<i>String</i>	The name of the asset associated with the email subscribe activity.



AssetId	<i>Long</i>	The Id of the asset associated with the email subscribe activity.
AssetType	<i>String</i>	The type of the asset associated with the email subscribe activity.
CampaignName	<i>String</i>	The name of the campaign associated with the email subscribe activity. Not supported in bulk mode.
EmailCampaignId	<i>String</i>	The Id of the email campaign associated with the email subscribe activity. Not supported in bulk mode.
EmailAddress	<i>String</i>	The address of the email. Only supported in bulk mode.
EmailRecipientId	<i>String</i>	The Id of the recipient of the email.
CampaignId	<i>Long</i>	The Id of the campaign associated with the email subscribe activity. Only supported in bulk mode.
ExternalId	<i>String</i>	The ExternalId associated with the email subscribe activity. Only supported in bulk mode.
ContactId	<i>Long</i>	The Id of the contact associated with the bounceback activity.
FirstName	<i>String</i>	The FirstName of the contact associated with the email subscribe activity.
LastName	<i>String</i>	The LastName of the contact associated with the email subscribe activity.
Company	<i>String</i>	The Company of the contact associated with the email subscribe activity.
City	<i>String</i>	The City of the contact associated with the email subscribe activity.
ContactCreatedAt	<i>Datetime</i>	The date and time the contact associated with the

		email subscribe activity was created.
ContactModifiedAt	<i>Datetime</i>	The date and time the contact associated with the email subscribe activity was created.
Address1	<i>String</i>	The first line address of the contact associated with the email subscribe activity.
Address2	<i>String</i>	The second line address of the contact associated with the email subscribe activity.
Title	<i>String</i>	The Title of the contact associated with the email subscribe activity.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

<b>Name</b>	<b>Type</b>	<b>Description</b>
ExportToken	<i>String</i>	The token returned from CreateExportDefinition. If Eloqua has not finished processing the request, the query will wait until the server has finished before returning data.
Depth	<i>String</i>	The depth of information requested from Eloqua.

## Activity\_EmailUnsubscribe

Retrieve Email Unsubscribe Activities.

### Table Specific Information

## Select

If you are not using the Bulk API, ContactId is required to query Activity tables.

Additionally, the COUNT function is supported for this table in bulk mode.

## Columns

Name	Type	Description
ID [KEY]	Long	The unique identifier of the email unsubscribe activity.
Name	String	The name of the email unsubscribe activity. Not supported in bulk mode.
CreatedAt	Datetime	The date and time the email unsubscribe activity was created.
Type	String	The type of email unsubscribe activity.
AssetName	String	The name of the asset associated with the email unsubscribe activity.
AssetId	Long	The Id of the asset associated with the email unsubscribe activity.
AssetType	String	The type of the asset associated with the email unsubscribe activity.
CampaignName	String	The name of the campaign associated with the email unsubscribe activity. Not supported in bulk mode.
EmailCampaignId	String	The Id of the email campaign associated with the email unsubscribe activity. Not supported in bulk mode.
EmailRecipientId	String	The Id of the recipient of the email.

EmailAddress	<i>String</i>	The address of the email. Only supported in bulk mode.
CampaignId	<i>Long</i>	The Id of the campaign associated with the email unsubscribe activity. Only supported in bulk mode.
ExternalId	<i>String</i>	The ExternalId associated with the email unsubscribe activity. Only supported in bulk mode.
ContactId	<i>Long</i>	The Id of the contact associated with the bounceback activity.
FirstName	<i>String</i>	The FirstName of the contact associated with the email unsubscribe activity.
LastName	<i>String</i>	The LastName of the contact associated with the email unsubscribe activity.
Company	<i>String</i>	The Company of the contact associated with the email unsubscribe activity.
City	<i>String</i>	The City of the contact associated with the email unsubscribe activity.
ContactCreatedAt	<i>Datetime</i>	The date and time the contact associated with the email unsubscribe activity was created.
ContactModifiedAt	<i>Datetime</i>	The date and time the contact associated with the email unsubscribe activity was created.
Address1	<i>String</i>	The first line address of the contact associated with the email unsubscribe activity.
Address2	<i>String</i>	The second line address of the contact associated with the email unsubscribe activity.
Title	<i>String</i>	The Title of the contact associated with the email unsubscribe activity.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
ExportToken	<i>String</i>	The token returned from CreateExportDefinition. If Eloqua has not finished processing the request, the query will wait until the server has finished before returning data.
Depth	<i>String</i>	The depth of information requested from Eloqua.

## Activity\_FormSubmit

Retrieve Form Submit Activities.

### Table Specific Information

#### Select

If you are not using the Bulk API, ContactId is required to query Activity tables.

Additionally, the COUNT function is supported for this table in bulk mode.

#### Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier of the form submit activity.

Name	<i>String</i>	The name of the form submit activity. Not supported in bulk mode.
CreatedAt	<i>Datetime</i>	The date and time the form submit activity was created.
Type	<i>String</i>	The activity type of the form submit activity.
AssetName	<i>String</i>	The name of the asset associated with the form submit activity.
AssetId	<i>Long</i>	The Id of the asset associated with the form submit activity.
AssetType	<i>String</i>	The type of the asset associated with the form submit activity.
ContactId	<i>Long</i>	The Id of the contact who performed the form submit activity.
Collection	<i>String</i>	The collection of the form submit activity. Not supported in bulk mode.
FormName	<i>String</i>	The name of the form associated with the form submit activity. Not supported in bulk mode.
FormData	<i>String</i>	The data of the form associated with the form submit activity. Not supported in bulk mode.
RawData	<i>String</i>	The raw data of the form submit activity. Only supported in bulk mode.
CampaignId	<i>Long</i>	The Id of the campaign associated with the form submit activity. Only supported in bulk mode.
VisitorId	<i>Long</i>	The Id of the visitor associated with the form submit activity. Only supported in bulk mode.
VisitorExternalId	<i>String</i>	The ExternalId of the visitor associated with the form

		submit activity. Only supported in bulk mode.
ExternalId	<i>String</i>	The ExternalId associated with the form submit activity. Only supported in bulk mode.
FirstName	<i>String</i>	The FirstName of the contact associated with the form submit activity.
LastName	<i>String</i>	The LastName of the contact associated with the form submit activity.
Company	<i>String</i>	The Company of the contact associated with the form submit activity.
City	<i>String</i>	The City of the contact associated with the bounceback activity.
ContactCreatedAt	<i>Datetime</i>	The date and time the contact associated with the form submit activity was created.
ContactModifiedAt	<i>Datetime</i>	The date and time the contact associated with the form submit activity was created.
Address1	<i>String</i>	The first line address of the contact associated with the form submit activity.
Address2	<i>String</i>	The second line address of the contact associated with the form submit activity.
Title	<i>String</i>	The Title of the contact associated with the form submit activity.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
ExportToken	<i>String</i>	The token returned from CreateExportDefinition. If Eloqua has not finished processing the request, the query will wait until the server has finished before returning data.
Depth	<i>String</i>	The depth of information requested from Eloqua.

## Activity\_PageView

Retrieve Page View Activities. This table is only supported in bulk mode.

### Table Specific Information

#### Select

COUNT is supported for this table.

#### Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier of the page view activity.
CreatedAt	<i>Datetime</i>	The date and time the page view activityactivity was created.
Type	<i>String</i>	The type of the activity associated with the page view activity.
ContactId	<i>Long</i>	The Id of the contact who performed the page view activity.



IPAddress	<i>String</i>	The IP address of the sender.
Url	<i>String</i>	The URL of the page view activity.
CampaignId	<i>Long</i>	The Id of the campaign associated with the page view activity.
ReferrerUrl	<i>String</i>	The referrer URL of the page view activity.
VisitorId	<i>Long</i>	The visitor Id of the page view activity.
VisitorExternalId	<i>String</i>	The visitor external Id of the page view activity.
WebVisitId	<i>Long</i>	The Web visit Id of the page view activity.
IsWebTrackingOptedIn	<i>Boolean</i>	Whether the visitor who performed the page view activity has opted into Web tracking.
ExternalId	<i>String</i>	The ExternalId associated with the page activity. Only supported in bulk mode.
FirstName	<i>String</i>	The FirstName of the contact associated with the page view activity.
LastName	<i>String</i>	The LastName of the contact associated with the page view activity.
Company	<i>String</i>	The Company of the contact associated with the page view activity.
City	<i>String</i>	The City of the contact associated with the page view activity.
ContactCreatedAt	<i>Datetime</i>	The date and time the contact associated with the

		page view activity was created.
ContactModifiedAt	<i>Datetime</i>	The date and time the contact associated with the page view activity was created.
Address1	<i>String</i>	The first line address of the contact associated with the page view activity.
Address2	<i>String</i>	The second line address of the contact associated with the page view activity.
Title	<i>String</i>	The Title of the contact associated with the page view activity.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

<b>Name</b>	<b>Type</b>	<b>Description</b>
ExportToken	<i>String</i>	The token returned from CreateExportDefinition. If Eloqua has not finished processing the request, the query will wait until the server has finished before returning data.
Depth	<i>String</i>	The depth of information requested from Eloqua.

## Activity\_WebVisit

Retrieve Web Visit Activities.

### Table Specific Information

## Select

If you are not using the Bulk API, ContactId is required to query Activity tables.

Additionally, the COUNT function is supported for this table in bulk mode.

## Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The Id of the Web visit activity.
Name	<i>String</i>	The name of the Web visit activity. Not supported in bulk mode.
CreatedAt	<i>Datetime</i>	The date and time the Web visit activity was created.
Type	<i>String</i>	The type of Web visit activity.
AssetId	<i>Long</i>	The Id of the asset associated with the Web visit activity. Not supported in bulk mode.
AssetType	<i>String</i>	The type of the asset associated with the Web visit activity. Not supported in bulk mode.
ContactId	<i>Long</i>	The Id of the contact who performed the Web visit activity. Not supported in bulk mode.
Duration	<i>String</i>	The duration of the Web visit activity.
QueryString	<i>String</i>	The query string of the Web visit activity. Not supported in bulk mode.
QueryStringDisplay	<i>String</i>	The query string display of the Web visit activity. Not supported in bulk mode.

Thread	<i>String</i>	The thread of the Web visit activity. Not supported in bulk mode.
URL	<i>String</i>	The URL of the Web visit activity. Not supported in bulk mode.
ReferrerUrl	<i>String</i>	The Referrer URL of the Web visit activity. Only supported in bulk mode.
VisitorId	<i>Long</i>	The Id of the visitor associated with the Web visit activity. Only supported in bulk mode.
VisitorExternalId	<i>String</i>	The visitor external Id of the Web visit activity. Only supported in bulk mode.
ExternalId	<i>String</i>	The ExternalId associated with the Web visit activity. Only supported in bulk mode.
NumberOfPages	<i>String</i>	The number of pages of the Web visit activity. Only supported in bulk mode.
IpAddress	<i>String</i>	The IP address of the Web visit activity. Only supported in bulk mode.
FirstPageViewUrl	<i>String</i>	The first page view URL of the Web visit activity. Only supported in bulk mode.
FirstName	<i>String</i>	The FirstName of the contact associated with the Web visit activity.
LastName	<i>String</i>	The LastName of the contact associated with the Web visit activity.
Company	<i>String</i>	The Company of the contact associated with the Web visit activity.
City	<i>String</i>	The City of the contact associated with the Web visit activity.
ContactCreatedAt	<i>Datetime</i>	The date and time the contact associated with the Web

		visit activity was created.
ContactModifiedAt	<i>Datetime</i>	The date and time the contact associated with the Web visit activity was created.
Address1	<i>String</i>	The first line address of the contact associated with the Web visit activity.
Address2	<i>String</i>	The second line address of the contact associated with the Web visit activity.
Title	<i>String</i>	The Title of the contact associated with the Web visit activity.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

<b>Name</b>	<b>Type</b>	<b>Description</b>
ExportToken	<i>String</i>	The token returned from CreateExportDefinition. If Eloqua has not finished processing the request, the query will wait until the server has finished before returning data.
Depth	<i>String</i>	The depth of information requested from Eloqua.

## CampaignElement

Retrieve Campaign Elements.

## Columns

<b>Name</b>	<b>Type</b>	<b>Description</b>
ID [KEY]	<i>Long</i>	The unique identifier for the campaign element.
Name	<i>String</i>	The name of the campaign element.
Type	<i>String</i>	The type of the campaign element.
FolderId	<i>Long</i>	The folder Id of the campaign element.
FormId	<i>String</i>	The form Id of the campaign element.
EmailId	<i>String</i>	The email Id of the campaign element.
LandingPageId	<i>String</i>	The landing page Id of the campaign element.
SegmentId	<i>String</i>	The segment Id of the campaign element.
ListId	<i>String</i>	The list Id of the campaign element.
CampaignId	<i>String</i>	The campaign Id of the campaign element.
memberCount	<i>Integer</i>	The member count of the campaign element.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
Depth	<i>String</i>	The depth of information requested from Eloqua.

## CampaignField

Retrieve Campaign Fields.

### Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier for the campaign field.
Name	<i>String</i>	The name of the campaign field.
DataType	<i>String</i>	The data type of the campaign field.
Description	<i>String</i>	The description of the campaign field.
DisplayType	<i>String</i>	The display type of the campaign field.
DisplayIndex	<i>String</i>	The display index of the campaign field.

FolderId	<i>Long</i>	The folder Id of the campaign field.
IsReadOnly	<i>Boolean</i>	Identifies whether the campaign field is read-only.
IsRequired	<i>Boolean</i>	Identifies whether the campaign field is required.
CreatedAt	<i>Datetime</i>	Date the campaign field was created.
CreatedBy	<i>String</i>	The Id of the user who created the campaign field.
Depth	<i>String</i>	The depth of the campaign field.

## CampaignFolder

Retrieve Campaign Folders.

### Columns

<b>Name</b>	<b>Type</b>	<b>Description</b>
ID [KEY]	<i>Long</i>	The unique identifier for the folder.
Name	<i>String</i>	The name of the folder.
IsSystem	<i>Boolean</i>	Identifies a system-generated folder.
Description	<i>String</i>	The description of the folder.



Permissions	<i>String</i>	The permissions of the folder.
CreatedAt	<i>Datetime</i>	Date the folder was created.
CreatedBy	<i>String</i>	The Id of the user who created the folder.
AccessedAt	<i>Datetime</i>	Date the folder was accessed.
CurrentStatus	<i>String</i>	The current status of the folder.
UpdatedAt	<i>Datetime</i>	Date the folder was last updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the folder.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

<b>Name</b>	<b>Type</b>	<b>Description</b>
Depth	<i>String</i>	The depth of information requested from Eloqua.

## ContactField

Retrieve Contact Fields.

## Table Specific Information

The EmailAddress column is required to insert into or to update the Contact table. For example,

```
UPDATE [Contact] SET emailAddress='example@test.com',firstName='test'
WHERE ID = ?
```

## Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier for the contact field.
Name	<i>String</i>	The name of the contact field.
UpdateType	<i>String</i>	The update rule for when to update this field.
CheckedValue	<i>String</i>	The checked value of the contact field.
DataType	<i>String</i>	The data type of the contact field.
DefaultValue	<i>String</i>	The default value for the contact field.
Description	<i>String</i>	The description of the contact field.
DisplayType	<i>String</i>	The display type of the contact field.

FolderId	<i>Long</i>	The folder Id of the contact field.
InternalName	<i>String</i>	The country of the contact field.
IsReadOnly	<i>Boolean</i>	Identifies whether the contact field is read-only.
IsRequired	<i>Boolean</i>	Identifies whether the contact field is required.
IsStandard	<i>Boolean</i>	Identifies whether the contact field is standard.
OptionListId	<i>Long</i>	The Id of the option list for the contact field.
OutputFormatId	<i>Long</i>	The Id of the output format for the contact field.
ScheduledFor	<i>String</i>	The date and time scheduled for the contact field.
SourceTemplatedId	<i>Long</i>	The Id of the source template for the contact field
UncheckedValue	<i>String</i>	The unchecked value of the contact field.
Permissions	<i>String</i>	The permissions of the contact field.
CreatedAt	<i>Datetime</i>	Date the contact field was created.
CreatedBy	<i>String</i>	The Id of the user who created the contact field.
AccessedAt	<i>Datetime</i>	Date the contact field was accessed.

CurrentStatus	<i>String</i>	The current status of the contact field.
Depth	<i>String</i>	The depth of the contact field.
UpdatedAt	<i>Datetime</i>	Date the contact field was updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the contact field.

## ContactFilter

Retrieve Contact Filters.

### Columns

<b>Name</b>	<b>Type</b>	<b>Description</b>
ID [KEY]	<i>Long</i>	The unique identifier for the contact filter.
Name	<i>String</i>	The name of the contact filter.
Type	<i>String</i>	The type of the contact filter.
FolderId	<i>Long</i>	The folder Id of the contact filter.
Description	<i>String</i>	The description of the contact filter.

ScheduledFor	<i>String</i>	The date and time scheduled for the contact filter.
Scope	<i>String</i>	The scope of the contact filter.
Statement	<i>String</i>	The statement of the contact filter.
Permissions	<i>String</i>	The permissions of the contact filter.
CreatedAt	<i>Datetime</i>	Date the contact filter was created.
CreatedBy	<i>String</i>	The Id of the user who created the contact filter.
AccessedAt	<i>Datetime</i>	Date the contact filter field was accessed.
CurrentStatus	<i>String</i>	The current status of the contact filter.
Depth	<i>String</i>	The depth of the contact filter.
UpdatedAt	<i>Datetime</i>	Date the contact filter was updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the contact filter.

## ContactFilterFolder

Retrieve Contact Filter Folders.

### Columns

<b>Name</b>	<b>Type</b>	<b>Description</b>
ID [KEY]	<i>Long</i>	The unique identifier for the folder.
Name	<i>String</i>	The name of the folder.
IsSystem	<i>Boolean</i>	Identifies a system-generated folder.
Description	<i>String</i>	The description of the folder.
Permissions	<i>String</i>	The permissions of the folder.
CreatedAt	<i>Datetime</i>	Date the folder was created.
CreatedBy	<i>String</i>	The Id of the user who created the folder.
AccessedAt	<i>Datetime</i>	Date the folder was accessed.
CurrentStatus	<i>String</i>	The current status of the folder.
UpdatedAt	<i>Datetime</i>	Date the folder was last updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the folder.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
Depth	<i>String</i>	The depth of information requested from Eloqua.

## ContactList

Create, update, delete, and query Contact Lists.

### Table Specific Information

#### Select

The following limitation exists for filtering on datetime columns: The "=" operator cannot be used with datetime columns in the WHERE clause. However, the "<", ">", ">=", and "<=" operators can be used to filter on datetime columns.

Additionally, the UpdatedAt column only supports filtering with the '>' operator.

Oracle Eloqua also has limited support on multiple conditions. In many tables the AND condition between filters is not supported. If AND is included the server might not return any results, nor will it return an error.

#### Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier for the contact list.
Name	<i>String</i>	The name of the contact list.

Count	<i>String</i>	The count of the contact list.
DataLookupId	<i>String</i>	The Id of the data lookup for the contact list.
MembershipAdditions	<i>String</i>	The number of contact membership additions in the contact list.
Scope	<i>String</i>	The contact list's scope: either local or global.
Description	<i>String</i>	The description of the contact list.
FolderId	<i>Long</i>	The Id of the folder for the contact list.
ScheduledFor	<i>Datetime</i>	The date and time scheduled for the contact list.
SourceTemplatedId	<i>Long</i>	The Id of the source template of the contact list.
Permissions	<i>String</i>	The permissions of the contact list.
CreatedAt	<i>Datetime</i>	Date the contact list was created.
CreatedBy	<i>String</i>	The Id of the user who created the contact list.
AccessedAt	<i>Datetime</i>	Date the contact list was accessed.
CurrentStatus	<i>String</i>	The current status of the contact list.



Depth	<i>String</i>	The depth of the contact list.
UpdatedAt	<i>Datetime</i>	Date the contact list was updated.
UpdatedBy	<i>String</i>	The Id of the user who created the contact list.

## ContactListFolder

Retrieve Contact List Folders.

### Columns

<b>Name</b>	<b>Type</b>	<b>Description</b>
ID [KEY]	<i>Long</i>	The unique identifier for the folder.
Name	<i>String</i>	The name of the folder.
IsSystem	<i>Boolean</i>	Identifies a system-generated folder.
Description	<i>String</i>	The description of the folder.
Permissions	<i>String</i>	The permissions of the folder.
CreatedAt	<i>Datetime</i>	Date the folder was created.
CreatedBy	<i>String</i>	The Id of the user who created the folder.

---

AccessedAt	<i>Datetime</i>	Date the folder was accessed.
CurrentStatus	<i>String</i>	The current status of the folder.
UpdatedAt	<i>Datetime</i>	Date the folder was last updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the folder.

---

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

---

<b>Name</b>	<b>Type</b>	<b>Description</b>
Depth	<i>String</i>	The depth of information requested from Eloqua.

---

## ContactScoringModelFolder

Retrieve Contact Scoring Model Folders.

### Columns

---

<b>Name</b>	<b>Type</b>	<b>Description</b>
-------------	-------------	--------------------

---

ID [KEY]	<i>Long</i>	The unique identifier for the folder.
Name	<i>String</i>	The name of the folder.
IsSystem	<i>Boolean</i>	Identifies a system-generated folder.
Description	<i>String</i>	The description of the folder.
Permissions	<i>String</i>	The permissions of the folder.
CreatedAt	<i>Datetime</i>	Date the folder was created.
CreatedBy	<i>String</i>	The Id of the user who created the folder.
AccessedAt	<i>Datetime</i>	Date the folder was accessed.
CurrentStatus	<i>String</i>	The current status of the folder.
UpdatedAt	<i>Datetime</i>	Date the folder was last updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the folder.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
Depth	<i>String</i>	The depth of information requested from Eloqua.

## ContactSegmentData

Retrieve Contact Segment Data. Note: The pseudo column SegmentID must be specified to retrieve the data of this table. This SegmentID can be retrieved from the ContactSegment table's ID column.

### Columns

Name	Type	Description
ID [KEY]	<i>Int64</i>	The unique identifier of the content segment data.
AccountName	<i>String</i>	This field is generated dynamically and corresponds to the AccountName field in the API.
BouncebackDate	<i>String</i>	This field is generated dynamically and corresponds to the BouncebackDate field in the API.
IsBounceback	<i>Boolean</i>	This field is generated dynamically and corresponds to the IsBounceback field in the API.
IsSubscribed	<i>Boolean</i>	This field is generated dynamically and corresponds to the IsSubscribed field in

		the API.
SubscriptionDate	<i>Datetime</i>	This field is generated dynamically and corresponds to the SubscriptionDate field in the API.
UnsubscriptionDate	<i>Datetime</i>	This field is generated dynamically and corresponds to the UnsubscriptionDate field in the API.
CreatedAt	<i>Datetime</i>	This field is generated dynamically and corresponds to the CreatedAt field in the API.
CreatedBy	<i>String</i>	This field is generated dynamically and corresponds to the CreatedBy field in the API.
AccessedAt	<i>Datetime</i>	This field is generated dynamically and corresponds to the AccessedAt field in the API.
CurrentStatus	<i>String</i>	This field is generated dynamically and corresponds to the CurrentStatus field in the API.
Depth	<i>String</i>	This field is generated dynamically and corresponds to the Depth field in the API.
UpdatedAt	<i>Datetime</i>	This field is generated dynamically and corresponds to the UpdatedAt field in the API.

UpdatedBy	<i>String</i>	This field is generated dynamically and corresponds to the UpdatedBy field in the API.
Email Address	<i>String</i>	This field is generated dynamically and corresponds to the Email Address field in the API.
First Name	<i>String</i>	This field is generated dynamically and corresponds to the First Name field in the API.
Last Name	<i>String</i>	This field is generated dynamically and corresponds to the Last Name field in the API.
Company	<i>String</i>	This field is generated dynamically and corresponds to the Company field in the API.
Email Display Name	<i>String</i>	This field is generated dynamically and corresponds to the Email Display Name field in the API.
Address 1	<i>String</i>	This field is generated dynamically and corresponds to the Address 1 field in the API.
Address 2	<i>String</i>	This field is generated dynamically and corresponds to the Address 2 field in the API.

Address 3	<i>String</i>	This field is generated dynamically and corresponds to the Address 3 field in the API.
City	<i>String</i>	This field is generated dynamically and corresponds to the City field in the API.
State or Province	<i>String</i>	This field is generated dynamically and corresponds to the State or Province field in the API.
Zip or Postal Code	<i>String</i>	This field is generated dynamically and corresponds to the Zip or Postal Code field in the API.
Country	<i>String</i>	This field is generated dynamically and corresponds to the Country field in the API.
Business Phone	<i>String</i>	This field is generated dynamically and corresponds to the Business Phone field in the API.
Mobile Phone	<i>String</i>	This field is generated dynamically and corresponds to the Mobile Phone field in the API.
Fax	<i>String</i>	This field is generated dynamically and corresponds to the Fax field in the API.
Title	<i>String</i>	This field is generated dynamically and corresponds

		to the Title field in the API.
Salutation	<i>String</i>	This field is generated dynamically and corresponds to the Salutation field in the API.
Salesperson	<i>String</i>	This field is generated dynamically and corresponds to the Salesperson field in the API.
SFDC Contact ID	<i>String</i>	This field is generated dynamically and corresponds to the SFDC Contact ID field in the API.
SFDC Lead ID	<i>String</i>	This field is generated dynamically and corresponds to the SFDC Lead ID field in the API.
Date Created	<i>Datetime</i>	This field is generated dynamically and corresponds to the Date Created field in the API.
Date Modified	<i>Datetime</i>	This field is generated dynamically and corresponds to the Date Modified field in the API.
Eloqua Contact ID	<i>String</i>	This field is generated dynamically and corresponds to the Eloqua Contact ID field in the API.
SFDC Account ID	<i>String</i>	This field is generated dynamically and corresponds to the SFDC Account ID field



		in the API.
Last Modified by CRM System	<i>Datetime</i>	This field is generated dynamically and corresponds to the Last Modified by CRM System field in the API.
Last SFDC Campaign ID	<i>String</i>	This field is generated dynamically and corresponds to the Last SFDC Campaign ID field in the API.
Last SFDC Campaign Status	<i>String</i>	This field is generated dynamically and corresponds to the Last SFDC Campaign Status field in the API.
Company Revenue	<i>String</i>	This field is generated dynamically and corresponds to the Company Revenue field in the API.
SFDC Email Opt Out	<i>String</i>	This field is generated dynamically and corresponds to the SFDC Email Opt Out field in the API.
Lead Source - Most Recent	<i>String</i>	This field is generated dynamically and corresponds to the Lead Source - Most Recent field in the API.
Lead Source - Original	<i>String</i>	This field is generated dynamically and corresponds to the Lead Source - Original field in the API.
Industry	<i>String</i>	This field is generated dynamically and corresponds to the Industry field in the

		API.
Annual Revenue	<i>String</i>	This field is generated dynamically and corresponds to the Annual Revenue field in the API.
Lead Status	<i>String</i>	This field is generated dynamically and corresponds to the Lead Status field in the API.
Job Role	<i>String</i>	This field is generated dynamically and corresponds to the Job Role field in the API.
Lead Score - High Value Website Content	<i>Double</i>	This field is generated dynamically and corresponds to the Lead Score - High Value Website Content field in the API.
Lead Score Date - Engagement - Most Recent	<i>Datetime</i>	This field is generated dynamically and corresponds to the Lead Score Date - Engagement - Most Recent field in the API.
Integrated Marketing and Sales Funnel Stage	<i>String</i>	This field is generated dynamically and corresponds to the Integrated Marketing and Sales Funnel Stage field in the API.
Product/Solution of Interest	<i>String</i>	This field is generated dynamically and corresponds to the Product/Solution of Interest field in the API.

Region	<i>String</i>	This field is generated dynamically and corresponds to the Region field in the API.
PURL Name (Default Hypersite)	<i>String</i>	This field is generated dynamically and corresponds to the PURL Name (Default Hypersite) field in the API.
Lead Rating - Combined	<i>String</i>	This field is generated dynamically and corresponds to the Lead Rating - Combined field in the API.
Email Address Domain	<i>String</i>	This field is generated dynamically and corresponds to the Email Address Domain field in the API.
First and Last Name	<i>String</i>	This field is generated dynamically and corresponds to the First and Last Name field in the API.
Company Size	<i>String</i>	This field is generated dynamically and corresponds to the Company Size field in the API.
Lead Score - Last High Touch Event Date	<i>Datetime</i>	This field is generated dynamically and corresponds to the Lead Score - Last High Touch Event Date field in the API.
Lead Rating - Profile (Explicit)	<i>String</i>	This field is generated dynamically and corresponds to the Lead Rating - Profile (Explicit) field in the API.

Lead Rating - Engagement (Implicit)	<i>String</i>	This field is generated dynamically and corresponds to the Lead Rating - Engagement (Implicit) field in the API.
Lead Score - Profile (Explicit)	<i>Double</i>	This field is generated dynamically and corresponds to the Lead Score - Profile (Explicit) field in the API.
Lead Score - Engagement (Implicit)	<i>Double</i>	This field is generated dynamically and corresponds to the Lead Score - Engagement (Implicit) field in the API.
Lead Score Date - Profile - Most Recent	<i>Datetime</i>	This field is generated dynamically and corresponds to the Lead Score Date - Profile - Most Recent field in the API.
Employees	<i>Double</i>	This field is generated dynamically and corresponds to the Employees field in the API.
Territory	<i>String</i>	This field is generated dynamically and corresponds to the Territory field in the API.
Lead Score	<i>String</i>	This field is generated dynamically and corresponds to the Lead Score field in the API.
PURL Name	<i>String</i>	This field is generated dynamically and corresponds

		to the PURL Name field in the API.
MD5 Hashed Email Address	<i>String</i>	This field is generated dynamically and corresponds to the MD5 Hashed Email Address field in the API.
SHA256 Hashed Email Address	<i>String</i>	This field is generated dynamically and corresponds to the SHA256 Hashed Email Address field in the API.
MD5 Hashed Business Phone	<i>String</i>	This field is generated dynamically and corresponds to the MD5 Hashed Business Phone field in the API.
SHA256 Hashed Business Phone	<i>String</i>	This field is generated dynamically and corresponds to the SHA256 Hashed Business Phone field in the API.
MD5 Hashed Mobile Phone	<i>String</i>	This field is generated dynamically and corresponds to the MD5 Hashed Mobile Phone field in the API.
SHA256 Hashed Mobile Phone	<i>String</i>	This field is generated dynamically and corresponds to the SHA256 Hashed Mobile Phone field in the API.
onlineInsertTest	<i>String</i>	This field is generated dynamically and corresponds to the onlineInsertTest field in the API.
Survey Initial Meeting Date & Time	<i>Datetime</i>	This field is generated

		dynamically and corresponds to the Survey Initial Meeting Date & Time field in the API.
TestColumn1	<i>String</i>	This field is generated dynamically and corresponds to the TestColumn1 field in the API.
TestColumn2	<i>String</i>	This field is generated dynamically and corresponds to the TestColumn2 field in the API.
TestColumn3	<i>String</i>	This field is generated dynamically and corresponds to the TestColumn3 field in the API.
TestColumn4	<i>String</i>	This field is generated dynamically and corresponds to the TestColumn4 field in the API.
TestColumn5	<i>String</i>	This field is generated dynamically and corresponds to the TestColumn5 field in the API.

## ContactSegmentFolder

Retrieve Contact Segment Folders.

### Columns

Name	Type	Description
------	------	-------------

ID [KEY]	<i>Long</i>	The unique identifier for the folder.
Name	<i>String</i>	The name of the folder.
IsSystem	<i>Boolean</i>	Identifies a system-generated folder.
Description	<i>String</i>	The description of the folder.
Permissions	<i>String</i>	The permissions of the folder.
CreatedAt	<i>Datetime</i>	Date the folder was created.
CreatedBy	<i>String</i>	The Id of the user who created the folder.
AccessedAt	<i>Datetime</i>	Date the folder was accessed.
CurrentStatus	<i>String</i>	The current status of the folder.
UpdatedAt	<i>Datetime</i>	Date the folder was last updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the folder.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
Depth	<i>String</i>	The depth of information requested from Eloqua.

## ContactView

Retrieve Contact Views.

### Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier for the contact view.
Name	<i>String</i>	The name of the contact view.
Description	<i>String</i>	The description of the contact view.
Permissions	<i>String</i>	The permissions of the contact view.
CreatedAt	<i>Datetime</i>	Date the contact view was created.
CreatedBy	<i>String</i>	The Id of the user who created the contact view.
AccessedAt	<i>Datetime</i>	Date the contact view was accessed.
UpdatedAt	<i>Datetime</i>	Date the contact view was updated.



---

UpdatedBy	<i>String</i>	The Id of the user who last updated the contact view.
-----------	---------------	---

---

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
Depth	<i>String</i>	The depth of information requested from Eloqua.

## ContentSectionFolder

Retrieve Content Section Folders.

### Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier for the folder.
Name	<i>String</i>	The name of the folder.
IsSystem	<i>Boolean</i>	Identifies a system-generated folder.
Description	<i>String</i>	The description of the folder.

Permissions	<i>String</i>	The permissions of the folder.
CreatedAt	<i>Datetime</i>	Date the folder was created.
CreatedBy	<i>String</i>	The Id of the user who created the folder.
AccessedAt	<i>Datetime</i>	Date the folder was accessed.
CurrentStatus	<i>String</i>	The current status of the folder.
UpdatedAt	<i>Datetime</i>	Date the folder was last updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the folder.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

<b>Name</b>	<b>Type</b>	<b>Description</b>
Depth	<i>String</i>	The depth of information requested from Eloqua.

## Dependencies

Retrieve Dependencies.

## Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	TBD.
Name	<i>String</i>	TBD.
Permissions	<i>String</i>	The permissions of the contact list.
Depth	<i>String</i>	The depth of the field.
Type	<i>String</i>	The type of the field.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
ObjectName	<i>String</i>	TDB
ObjectId	<i>String</i>	TBD

# DynamicContent

Retrieve Dynamic Contents.

## Columns

Name	Type	Description
ID [KEY]	Long	The unique identifier for the dynamic content.
Name	String	The name of the dynamic content.
Description	String	The description of the dynamic content.
FolderId	Long	The Id of the folder for the dynamic content.
ScheduledFor	Datetime	The date and time scheduled for the dynamic content.
SourceTemplatedId	Long	The Id of the source template of the dynamic content.
Permissions	String	The permissions of the dynamic content.
CreatedAt	Datetime	Date the dynamic content was created.
CreatedBy	String	The Id of the user who created the dynamic content.
AccessedAt	Datetime	Date the dynamic content was accessed.

CurrentStatus	<i>String</i>	The current status of the dynamic content.
UpdatedAt	<i>Datetime</i>	Date the dynamic content was last updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the dynamic content.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
Depth	<i>String</i>	The depth of information requested from Eloqua.

## DynamicContentFolder

Retrieve Dynamic Content Folders.

### Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier for the folder.
Name	<i>String</i>	The name of the folder.

IsSystem	<i>Boolean</i>	Identifies a system-generated folder.
Description	<i>String</i>	The description of the folder.
Permissions	<i>String</i>	The permissions of the folder.
CreatedAt	<i>Datetime</i>	Date the folder was created.
CreatedBy	<i>String</i>	The Id of the user who created the folder.
AccessedAt	<i>Datetime</i>	Date the folder was accessed.
CurrentStatus	<i>String</i>	The current status of the folder.
UpdatedAt	<i>Datetime</i>	Date the folder was last updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the folder.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

<b>Name</b>	<b>Type</b>	<b>Description</b>
Depth	<i>String</i>	The depth of information requested from Eloqua.

## EmailDeployment

Retrieve email deployments.

### Columns

Name	Type	Description
ID [KEY]	Long	The unique identifier for the email deployment.
Name	String	The name of the email deployment.
CurrentStatus	String	The current status of the email deployment.
SuccessfulSendCount	Int	The successful send count of the email deployment.
FailedSendCount	Int	The failed send count of the email deployment.
ContactId	String	The contact ID of the email deployment.
SendFromUserId	String	The sent user ID of the email deployment.
EndAt	Datetime	Date the email deployment ended.
Permissions	String	The permissions of the email deployment.
Email_ID	Long	The unique identifier of the email.

Email_Name	<i>String</i>	The name of the email.
Email_EmailGroupId	<i>Long</i>	The Id of the email group.
Email_IsPlainTextEditable	<i>Boolean</i>	Identifies whether the email is plaintext editable.
Email_IsTracked	<i>Boolean</i>	Identifies whether the email is tracked.
Email_PlainText	<i>String</i>	The plaintext version of the email.
Email_BouncebackEmail	<i>String</i>	The bounce-back email address.
Email_ReplyToEmail	<i>String</i>	The email address in the reply-to line of the email.
Email_ReplyToName	<i>String</i>	The name in the reply-to line of the email.
Email_SenderEmail	<i>String</i>	The email address of the sender.
Email_SenderName	<i>String</i>	The name of the sender.
Email_SendPlainTextOnly	<i>Boolean</i>	Identifies whether the plaintext version should be sent.
Email_Subject	<i>String</i>	The subject of the email.
Email_Description	<i>String</i>	The description of the email.
Email_FolderId	<i>Long</i>	The Id of the folder for the email.



Email_Permissions	<i>String</i>	The permissions of the email.
Email_CreatedAt	<i>Datetime</i>	Date the email was created.
Email_CreatedBy	<i>String</i>	The Id of the user who created the email.
Email_CurrentStatus	<i>String</i>	The current status of the email.
Email_UpdatedAt	<i>Datetime</i>	Date the email was updated.
Email_UpdatedBy	<i>String</i>	The Id of the user who last updated the email.
Email_HtmlContent_Type	<i>String</i>	The type for the content of the email.
Email_HtmlContent_ContentSource	<i>String</i>	The content source for the content of the email.
Email_HtmlContent_DocType	<i>String</i>	The doc type for the content of the email.
Email_HtmlContent_HTMLBody	<i>String</i>	The HTML body for the content of the email.
Email_HtmlContent_Root	<i>String</i>	The root for the content of the email.
Email_HtmlContent_MetaTags	<i>String</i>	The meta tags for the content of the email.
Email_HtmlContent_SystemHeader	<i>String</i>	The system header for the content of the email.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
Depth	<i>String</i>	The depth of information requested from Eloqua.

## EmailFolder

Retrieve Email Folders.

### Table Specific Information

#### Select

The following limitation exists for filtering on datetime columns: The "=" operator cannot be used with datetime columns in the WHERE clause. However, the "<", ">", ">=", and "<=" operators can be used to filter on datetime columns.

Additionally, the UpdatedAt column only supports filtering with the '>' operator.

Oracle Eloqua also has limited support on multiple conditions. In many tables the AND condition between filters is not supported. If AND is included the server might not return any results, nor will it return an error.

#### Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier for the folder.

---

Name	<i>String</i>	The name of the folder.
IsSystem	<i>Boolean</i>	Identifies a system-generated folder.
Description	<i>String</i>	The description of the folder.
Permissions	<i>String</i>	The permissions of the folder.
CreatedAt	<i>Datetime</i>	Date the folder was created.
CreatedBy	<i>String</i>	The Id of the user who created the folder.
AccessedAt	<i>Datetime</i>	Date the folder was accessed.
CurrentStatus	<i>String</i>	The current status of the folder.
UpdatedAt	<i>Datetime</i>	Date the folder was last updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the folder.

---

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
Depth	<i>String</i>	The depth of information requested from Eloqua.

## EmailFooterFolder

Retrieve Email Footer Folders.

### Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier for the folder.
Name	<i>String</i>	The name of the folder.
IsSystem	<i>Boolean</i>	Identifies a system-generated folder.
Description	<i>String</i>	The description of the folder.
Permissions	<i>String</i>	The permissions of the folder.
CreatedAt	<i>Datetime</i>	Date the folder was created.
CreatedBy	<i>String</i>	The Id of the user who created the folder.
AccessedAt	<i>Datetime</i>	Date the folder was accessed.

CurrentStatus	<i>String</i>	The current status of the folder.
UpdatedAt	<i>Datetime</i>	Date the folder was last updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the folder.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
Depth	<i>String</i>	The depth of information requested from Eloqua.

## EmailHeaderFolder

Retrieve Email Header Folders.

### Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier for the folder.
Name	<i>String</i>	The name of the folder.

IsSystem	<i>Boolean</i>	Identifies a system-generated folder.
Description	<i>String</i>	The description of the folder.
Permissions	<i>String</i>	The permissions of the folder.
CreatedAt	<i>Datetime</i>	Date the folder was created.
CreatedBy	<i>String</i>	The Id of the user who created the folder.
AccessedAt	<i>Datetime</i>	Date the folder was accessed.
CurrentStatus	<i>String</i>	The current status of the folder.
UpdatedAt	<i>Datetime</i>	Date the folder was last updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the folder.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

<b>Name</b>	<b>Type</b>	<b>Description</b>
Depth	<i>String</i>	The depth of information requested from Eloqua.

## ExternalType

Retrieve External Types.

### Columns

Name	Type	Description
ID [KEY]	Long	The unique identifier for the external type.
Name	String	The name of the external type.
Description	String	The description of the external type.
FolderId	Long	The Id of the folder for the external type.
ScheduledFor	Datetime	The date and time scheduled for the external type.
SourceTemplatedId	Long	The Id of the source template of the external type.
Permissions	String	The permissions of the external type.
CreatedAt	Datetime	Date the external type was created.
CreatedBy	String	The Id of the user who created the external type.
AccessedAt	Datetime	Date the external type was accessed.

CurrentStatus	<i>String</i>	The current status of the external type.
UpdatedAt	<i>Datetime</i>	Date the external type was updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the external type.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
Depth	<i>String</i>	The depth of information requested from Eloqua.

## FieldMerge

Retrieve Field Merges.

### Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier for the field merge.
Name	<i>String</i>	The name of the field merge.



Description	<i>String</i>	The description of the field merge.
FolderId	<i>Long</i>	The Id of the folder for the field merge.
ScheduledFor	<i>Datetime</i>	The date and time scheduled for the field merge.
SourceTemplatedId	<i>Long</i>	The Id of the source template of the field merge.
Permissions	<i>String</i>	The permissions of the field merge.
CreatedAt	<i>Datetime</i>	Date the field merge was created.
CreatedBy	<i>String</i>	The Id of the user who created the field merge.
AccessedAt	<i>Datetime</i>	Date the field merge was accessed.
CurrentStatus	<i>String</i>	The current status of the field merge.
UpdatedAt	<i>Datetime</i>	Date the field merge was updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the field merge.
Syntax	<i>String</i>	The syntax of the field merge.
DefaultValue	<i>String</i>	The default value of the data lookup for the field merge.

ContactFieldId	<i>Long</i>	The contact field Id of the field merge.
AccountFieldId	<i>Long</i>	The account field Id of the field merge.
EventId	<i>Long</i>	The event Id of the field merge.
EventFieldId	<i>Long</i>	The event field Id of the field merge.
EventSessionFieldId	<i>Long</i>	The event session field Id of the field merge.
MergeType	<i>String</i>	The merge type of the field merge.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

<b>Name</b>	<b>Type</b>	<b>Description</b>
Depth	<i>String</i>	The depth of information requested from Eloqua.

## FieldMergeFolder

Retrieve Field Merge Folders.

## Columns

<b>Name</b>	<b>Type</b>	<b>Description</b>
ID [KEY]	<i>Long</i>	The unique identifier for the folder.
Name	<i>String</i>	The name of the folder.
IsSystem	<i>Boolean</i>	Identifies a system-generated folder.
Description	<i>String</i>	The description of the folder.
Permissions	<i>String</i>	The permissions of the folder.
CreatedAt	<i>Datetime</i>	Date the folder was created.
CreatedBy	<i>String</i>	The Id of the user who created the folder.
AccessedAt	<i>Datetime</i>	Date the folder was accessed.
CurrentStatus	<i>String</i>	The current status of the folder.
UpdatedAt	<i>Datetime</i>	Date the folder was last updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the folder.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
Depth	<i>String</i>	The depth of information requested from Eloqua.

## FormElement

Retrieve Form Elements.

### Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier for the form element.
CreatedFromContactFieldId	<i>Long</i>	The contact field Id which created the form element.
Name	<i>String</i>	The name of the form element.
Type	<i>String</i>	The type of the form element.
DataType	<i>String</i>	The data type of the form element.
HTMLName	<i>String</i>	The HTML name of the form element.

Style	<i>String</i>	The style of the form element.
Instructions	<i>String</i>	The instructions of the form element.
Validations	<i>String</i>	The validations of the form element.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
FormId	<i>String</i>	TBD
Depth	<i>String</i>	The depth of information requested from Eloqua.

## FormFolder

Retrieve Form Folders.

## Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier for the folder.

---

Name	<i>String</i>	The name of the folder.
IsSystem	<i>Boolean</i>	Identifies a system-generated folder.
Description	<i>String</i>	The description of the folder.
Permissions	<i>String</i>	The permissions of the folder.
CreatedAt	<i>Datetime</i>	Date the folder was created.
CreatedBy	<i>String</i>	The Id of the user who created the folder.
AccessedAt	<i>Datetime</i>	Date the folder was accessed.
CurrentStatus	<i>String</i>	The current status of the folder.
UpdatedAt	<i>Datetime</i>	Date the folder was last updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the folder.

---

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
Depth	<i>String</i>	The depth of information requested from Eloqua.

## FormProcessingStep

Retrieve Form Processing Steps.

### Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier for the form processing step.
Name	<i>String</i>	The name of the form processing step.
Type	<i>String</i>	The type of the form processing step.
Condition	<i>String</i>	The condition of the form processing step.
execute	<i>String</i>	The execute of the form processing step.
Mappings	<i>String</i>	The mappings of the form processing step.
keyFieldMapping	<i>String</i>	The key field mapping of the form processing step.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
FormId	<i>String</i>	TBD
Depth	<i>String</i>	The depth of information requested from Eloqua.

## HyperlinkFolder

Retrieve Hyperlink Folders.

### Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier for the folder.
Name	<i>String</i>	The name of the folder.
IsSystem	<i>Boolean</i>	Identifies a system-generated folder.
Description	<i>String</i>	The description of the folder.
Permissions	<i>String</i>	The permissions of the folder.



---

CreatedAt	<i>Datetime</i>	Date the folder was created.
CreatedBy	<i>String</i>	The Id of the user who created the folder.
AccessedAt	<i>Datetime</i>	Date the folder was accessed.
CurrentStatus	<i>String</i>	The current status of the folder.
UpdatedAt	<i>Datetime</i>	Date the folder was last updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the folder.

---

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

---

<b>Name</b>	<b>Type</b>	<b>Description</b>
Depth	<i>String</i>	The depth of information requested from Eloqua.

---

## Image

Retrieve Image.

## Columns

Name	Type	Description
ID [KEY]	Long	The unique identifier for the image.
Name	String	The name of the image.
Permissions	String	The permissions of the image.
Description	String	The description of the image.
FolderId	Long	The folder Id of the image.
CreatedAt	Datetime	Date the image was created.
CreatedBy	String	The Id of the user who created the image.
UpdatedAt	Datetime	Date the image was last updated.
UpdatedBy	String	The Id of the user who last updated the image.
FullImageUrl	String	The full image URL of the image.
ThumbNailUrl	String	The thumb nail URL of the image.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
Depth	<i>String</i>	The depth of information requested from Eloqua.

## ImageFolder

Retrieve Image Folders.

### Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier for the folder.
Name	<i>String</i>	The name of the folder.
IsSystem	<i>Boolean</i>	Identifies a system-generated folder.
Description	<i>String</i>	The description of the folder.
Permissions	<i>String</i>	The permissions of the folder.
CreatedAt	<i>Datetime</i>	Date the folder was created.

---

CreatedBy	<i>String</i>	The Id of the user who created the folder.
AccessedAt	<i>Datetime</i>	Date the folder was accessed.
CurrentStatus	<i>String</i>	The current status of the folder.
UpdatedAt	<i>Datetime</i>	Date the folder was last updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the folder.

---

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

---

<b>Name</b>	<b>Type</b>	<b>Description</b>
Depth	<i>String</i>	The depth of information requested from Eloqua.

---

## ImportedFile

Retrieve Imported Files.

## Columns

<b>Name</b>	<b>Type</b>	<b>Description</b>
ID [KEY]	<i>Long</i>	The unique identifier for the imported file.
Name	<i>String</i>	The name of the imported file.
Type	<i>String</i>	The type of the imported file.
FolderId	<i>Long</i>	The folder Id of the imported file.
FileName	<i>String</i>	The file name of the imported file.
Link	<i>String</i>	The link of the imported file.
TrackedLink	<i>String</i>	The tracked link of the imported file.
Permissions	<i>String</i>	The permissions of the imported file.
CreatedAt	<i>Datetime</i>	Date the imported file was created.
CreatedBy	<i>String</i>	The Id of the user who created the imported file.
AccessedAt	<i>Datetime</i>	Date the imported file was accessed.
Depth	<i>String</i>	The depth of the imported file.
UpdatedAt	<i>Datetime</i>	Date the imported file was updated.

---

UpdatedBy	<i>String</i>	The Id of the user who last updated the imported file.
-----------	---------------	--

---

## ImportedFileFolder

Retrieve Imported File Folders.

### Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier for the folder.
Name	<i>String</i>	The name of the folder.
IsSystem	<i>Boolean</i>	Identifies a system-generated folder.
Description	<i>String</i>	The description of the folder.
Permissions	<i>String</i>	The permissions of the folder.
CreatedAt	<i>Datetime</i>	Date the folder was created.
CreatedBy	<i>String</i>	The Id of the user who created the folder.
AccessedAt	<i>Datetime</i>	Date the folder was accessed.

CurrentStatus	<i>String</i>	The current status of the folder.
UpdatedAt	<i>Datetime</i>	Date the folder was last updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the folder.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
Depth	<i>String</i>	The depth of information requested from Eloqua.

## LandingPageFolder

Retrieve Landing Page Folders.

### Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier for the folder.
Name	<i>String</i>	The name of the folder.

IsSystem	<i>Boolean</i>	Identifies a system-generated folder.
Description	<i>String</i>	The description of the folder.
Permissions	<i>String</i>	The permissions of the folder.
CreatedAt	<i>Datetime</i>	Date the folder was created.
CreatedBy	<i>String</i>	The Id of the user who created the folder.
AccessedAt	<i>Datetime</i>	Date the folder was accessed.
CurrentStatus	<i>String</i>	The current status of the folder.
UpdatedAt	<i>Datetime</i>	Date the folder was last updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the folder.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

<b>Name</b>	<b>Type</b>	<b>Description</b>
Depth	<i>String</i>	The depth of information requested from Eloqua.



## LeadScoringModel

Query Lead Scoring Models.

### Columns

Name	Type	Description
ID [KEY]	Integer	The unique identifier for the landing page.
Name	String	The name of the landing page.
Status	String	The status of the lead scoring model.
Rating	String	The rating of the lead scoring model.
ProfileScore	String	The profiles core of the lead scoring model.
EngagementScore	String	The engagement score of the lead scoring model.
Uri	String	The URI of the lead scoring model.
CreatedBy	String	The Id of the user who created the lead scoring model.
UpdatedBy	String	The Id of the user who updated the lead scoring model.
CreatedAt	Timestamp	Date the lead scoring model was created.

UpdatedAt	<i>Timestamp</i>	Date the lead scoring model was updated.
-----------	------------------	--

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
Depth	<i>String</i>	The depth of information requested from Eloqua.

## PageTag

Retrieve Page Tags.

## Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier for the page tag.
Name	<i>String</i>	The name of the page tag.
Description	<i>String</i>	The description of the page tag.
Permissions	<i>String</i>	The permissions of the page tag.

---

CreatedAt	<i>Datetime</i>	Date the page tag was created.
CreatedBy	<i>String</i>	The Id of the user who created the page tag.
AccessedAt	<i>Datetime</i>	Date the page tag was accessed.
CurrentStatus	<i>String</i>	The current status of the page tag.
UpdatedAt	<i>Datetime</i>	Date the page tag was updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the page tag.

---

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

---

<b>Name</b>	<b>Type</b>	<b>Description</b>
Depth	<i>String</i>	The depth of information requested from Eloqua.

---

## PageTagGroup

Retrieve Page Tag Groups.

## Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier for the page tag group.
Name	<i>String</i>	The name of the page tag group.
Description	<i>String</i>	The description of the page tag group.
Permissions	<i>String</i>	The permissions of the page tag group.
CreatedAt	<i>Datetime</i>	Date the page tag group was created.
CreatedBy	<i>String</i>	The Id of the user who created the page tag group.
AccessedAt	<i>Datetime</i>	Date the page tag group was accessed.
CurrentStatus	<i>String</i>	The current status of the page tag group.
UpdatedAt	<i>Datetime</i>	Date the page tag group was updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the page tag group.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
Depth	<i>String</i>	The depth of information requested from Eloqua.

## Program

Retrieve Programs.

### Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier for the program.
Name	<i>String</i>	The name of the program.
Description	<i>String</i>	The description of the program.
FolderId	<i>Long</i>	The Id of the folder for the program.
ScheduledFor	<i>Datetime</i>	The date and time scheduled for the program.
SourceTemplatedId	<i>Long</i>	The Id of the source template of the program.

Permissions	<i>String</i>	The permissions of the program.
CreatedAt	<i>Datetime</i>	Date the program list was created.
CreatedBy	<i>String</i>	The Id of the user who created the program.
AccessedAt	<i>Datetime</i>	Date the program was accessed.
UpdatedAt	<i>Datetime</i>	Date the program was updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the program.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

<b>Name</b>	<b>Type</b>	<b>Description</b>
Depth	<i>String</i>	The depth of information requested from Eloqua.

## Style

Retrieve Styles.

## Columns

<b>Name</b>	<b>Type</b>	<b>Description</b>
ID [KEY]	<i>Long</i>	The unique identifier for the style.
Name	<i>String</i>	The name of the style.
Description	<i>String</i>	The description of the style.
UpdatedAt	<i>Datetime</i>	Date the style was last updated.
CreatedAt	<i>Datetime</i>	Date the style was created.
Body	<i>String</i>	The body of the style.
Column1	<i>String</i>	Column 1 of the style.
Column2	<i>String</i>	Column 2 of the style.
Column3	<i>String</i>	Column 3 of the style.
Footer	<i>String</i>	The footer of the style.
Header	<i>String</i>	The header of the style.
Headline	<i>String</i>	The headline of the style.
Icon	<i>String</i>	The icon of the style.

Page	<i>String</i>	The page of the style.
------	---------------	------------------------

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
Depth	<i>String</i>	The depth of information requested from Eloqua.

## Template

Retrieve Templates.

### Columns

Name	Type	Description
ID [KEY]	<i>Long</i>	The unique identifier for the template.
Name	<i>String</i>	The name of the template.
Description	<i>String</i>	The description of the template.



FolderId	<i>Long</i>	The Id of the folder for the template.
ScheduledFor	<i>Datetime</i>	The date and time scheduled for the template.
SourceTemplateId	<i>Long</i>	The source template Id of the template.
Permissions	<i>String</i>	The permissions of the template.
CreatedAt	<i>Datetime</i>	Date the template was created.
CreatedBy	<i>String</i>	The Id of the user who created the template.
AccessedAt	<i>Datetime</i>	Date the template was accessed.
CurrentStatus	<i>String</i>	The current status of the template.
UpdatedAt	<i>Datetime</i>	Date the template was last updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the template.
AssetType	<i>String</i>	The asset type of the template.
AssetId	<i>Long</i>	The asset Id of the template.
CategoryId	<i>Long</i>	The category Id of the template.
DocumentActionTitle	<i>String</i>	The documentation title of the template.

DocumentAionUrl	<i>String</i>	The documentation URL of the template.
IsContentProtected	<i>Boolean</i>	Identifies whether the template is content protected.
UseThumbnail	<i>Boolean</i>	Identifies whether the template uses a thumbnail.
IconUrl	<i>String</i>	The icon URL of the template.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

<b>Name</b>	<b>Type</b>	<b>Description</b>
Depth	<i>String</i>	The depth of information requested from Eloqua.

## TemplateCategory

Retrieve Template Categories.

## Columns

<b>Name</b>	<b>Type</b>	<b>Description</b>
-------------	-------------	--------------------

ID [KEY]	<i>Long</i>	The unique identifier for the template category.
Name	<i>String</i>	The name of the template.
Description	<i>String</i>	The description of the template category.
FolderId	<i>Long</i>	The Id of the folder for the template category.
ScheduledFor	<i>Datetime</i>	The date and time scheduled for the template category.
SourceTemplateId	<i>Long</i>	The source template Id of the template category.
Permissions	<i>String</i>	The permissions of the template category.
CreatedAt	<i>Datetime</i>	Date the template category was created.
CreatedBy	<i>String</i>	The Id of the user who created the template category.
AccessedAt	<i>Datetime</i>	Date the template category was accessed.
CurrentStatus	<i>String</i>	The current status of the template category.
UpdatedAt	<i>Datetime</i>	Date the template category was updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the template category.

Siteld	<i>Long</i>	The site Id of the template category.
DefaultTemplateIconUrl	<i>String</i>	The default template icon URL of the template category.
IconUrl	<i>String</i>	The icon URL of the template category.
Publisher	<i>String</i>	The publisher of the template category.
Campaigns	<i>Int</i>	The campaigns of the template category.
ContactSegments	<i>Int</i>	The contact segments of the template category.
Forms	<i>Int</i>	The forms of the template category.
LandingPages	<i>Int</i>	The landing pages of the template category.
Emails	<i>Int</i>	The emails of the template category.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

<b>Name</b>	<b>Type</b>	<b>Description</b>
Depth	<i>String</i>	The depth of information requested from Eloqua.

# TrackedUrl

Retrieve Tracked Urls.

## Columns

Name	Type	Description
ID [KEY]	Long	The unique identifier for the tracked URL.
Name	String	The name of the tracked URL.
Description	String	The description of the tracked URL.
FolderId	Long	The Id of the folder for the tracked URL.
ScheduledFor	Datetime	The date and time scheduled for the tracked URL.
SourceTemplatedId	Long	The Id of the source template of the tracked URL.
Permissions	String	The permissions of the tracked URL.
CreatedAt	Datetime	Date the tracked URL was created.
CreatedBy	String	The Id of the user who created the tracked URL.
AccessedAt	Datetime	Date the tracked URL was accessed.

CurrentStatus	<i>String</i>	The current status of the tracked URL.
Depth	<i>String</i>	The depth of the tracked URL.
UpdatedAt	<i>Datetime</i>	Date the tracked URL was last updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the tracked URL.
Url	<i>String</i>	The URL of the tracked URL.

## User

Retrieve Users.

## Table Specific Information

### Select

The following limitation exists for filtering on datetime columns: The "=" operator cannot be used with datetime columns in the WHERE clause. However, the "<", ">", ">=", and "<=" operators can be used to filter on datetime columns.

Additionally, the UpdatedAt column only supports filtering with the '>' operator.

Oracle Eloqua also has limited support on multiple conditions. In many tables the AND condition between filters is not supported. If AND is included the server might not return any results, nor will it return an error.

### Columns

<b>Name</b>	<b>Type</b>	<b>Description</b>
ID [KEY]	<i>Long</i>	The unique identifier for this user.
Name	<i>String</i>	The name of the user.
Company	<i>String</i>	The company of the user.
Capabilities	<i>String</i>	The capabilities of the user.
BetaAccess	<i>String</i>	The beta access flag for the user.
DefaultAccountViewId	<i>Long</i>	The Id of the default account view for the user.
DefaultContactViewId	<i>Long</i>	The Id of the default contact view for the user.
EmailAddress	<i>String</i>	The email address of the user.
LoggedInAt	<i>String</i>	When the user last logged in.
LoginName	<i>String</i>	The login name of the user.
Preferences	<i>String</i>	The preferences for the user.
ProductPermissions	<i>String</i>	The product permissions for the user.
TypePermissions	<i>String</i>	The type permissions for the user.

ScheduledFor	<i>Datetime</i>	The date and time scheduled for the user.
SourceTemplatedId	<i>Long</i>	The Id of the source template for the user.
Description	<i>String</i>	The description of the user.
FolderId	<i>Long</i>	The Id of the folder for the user.
Permissions	<i>String</i>	The permissions of the user.
CreatedAt	<i>Datetime</i>	Date the user was created.
CreatedBy	<i>String</i>	The Id of the user that created the user.
AccessedAt	<i>Datetime</i>	Date the user was accessed.
CurrentStatus	<i>String</i>	The current status of the user.
Depth	<i>String</i>	The depth of the user.
UpdatedAt	<i>Datetime</i>	Date the user was updated.
UpdatedBy	<i>String</i>	The Id of the user who last updated the user.

## Visitor

Retrieve Visitors.



## Table Specific Information

### Select

The following limitation exists for filtering on datetime columns: The "=" operator cannot be used with datetime columns in the WHERE clause. However, the "<", ">", ">=", and "<=" operators can be used to filter on datetime columns.

Additionally, the UpdatedAt column only supports filtering with the '>' operator.

Oracle Eloqua also has limited support on multiple conditions. In many tables the AND condition between filters is not supported. If AND is included the server might not return any results, nor will it return an error.

### Columns

Name	Type	Description
VisitorId [KEY]	Long	The Id of the visitor profile.
CreatedAt	Datetime	Date the visitor profile was created.
V_IPAddress	String	IP address of the visitor.
V_LastVisitDateAndTime	Datetime	Date of last visit to an Eloqua asset for this profile.
ExternalId	String	Customer GUID for this visitor profile.
ContactId	Long	The Id of the contact associated with this profile, if any.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
Depth	<i>String</i>	The depth of information requested from Eloqua.

## VisitorProfileField

Retrieve all of the visitor profile fields.

### Columns

Name	Type	Description
Id [KEY]	<i>Long</i>	The Id of the visitor profile field.
Type	<i>String</i>	The type of the visitor profile field.
name	<i>String</i>	The name of the visitor profile field.
DataType	<i>String</i>	The datatype of the visitor profile field.
InternalName	<i>String</i>	The internal name of the visitor profile field.
Length	<i>Long</i>	The length of the visitor profile field.

## Pseudo-Columns

Pseudo column fields are used in the WHERE clause of SELECT statements and offer a more granular control over the tuples that are returned from the data source.

Name	Type	Description
Depth	<i>String</i>	The depth of information requested from Eloqua.

## Stored Procedures

Stored procedures are function-like interfaces that extend the functionality of the adapter beyond simple SELECT/INSERT/UPDATE/DELETE operations with Oracle Eloqua.

Stored procedures accept a list of parameters, perform their intended function, and then return, if applicable, any relevant response data from Oracle Eloqua, along with an indication of whether the procedure succeeded or failed.

### Oracle Eloqua Adapter Stored Procedures

Name	Description
<a href="#">ClearImportDefinitions</a>	Clears all open import definitions on a given table.
<a href="#">CreateExportDefinition</a>	Returns the token needed to retrieve data from a bulk export request.
<a href="#">DeleteExportData</a>	Delete the data stored on the Eloqua server from a bulk query request.
<a href="#">GetExportDefinition</a>	Retrieves the current state of the bulk export process on the server.

<a href="#">GetOAuthAccessToken</a>	Gets an authentication token from Eloqua.
<a href="#">GetOAuthAuthorizationUrl</a>	Gets the authorization URL that must be opened separately by the user to grant access to your application. Only needed when developing Web apps.
<a href="#">GetSyncLogs</a>	retrieve a sync's logs.
<a href="#">RefreshOAuthAccessToken</a>	Refreshes the OAuth access token used for authentication with Eloqua.
<a href="#">SubscribeContacts</a>	Subscribes a list of contacts to Email Groups.
<a href="#">UnsubscribeContacts</a>	Unsubscribes a list of contacts to Email Groups.

## ClearImportDefinitions

Clears all open import definitions on a given table.

### Input

Name	Type	Description
TableName	<i>String</i>	The table to remove the import definitions from.

## CreateExportDefinition

Returns the token needed to retrieve data from a bulk export request.

### Input

Name	Type	Description
Query	<i>String</i>	The SQL query used to retrieve data from a bulk table.

## Result Set Columns

Name	Type	Description
ExportToken	<i>String</i>	The token used to retrieve data from a bulk export request. Supply this value in the WHERE clause when SELECTing data from a bulk API table. In the form 'DefinitionUrl SyncsUrl'.

## DeleteExportData

Delete the data stored on the Eloqua server from a bulk query request.

## Input

Name	Type	Description
ExportToken	<i>String</i>	The token returned from CreateExportDefinition

## Result Set Columns

Name	Type	Description
State	<i>String</i>	A value of 'success' indicates delete export data successfully. Any other state would indicate an error.

## GetExportDefinition

Retrieves the current state of the bulk export process on the server.

### Input

Name	Type	Description
ExportToken	<i>String</i>	The token returned from CreateExportDefinition
GetExportDefinition	<i>String</i>	Specifies whether to return extra metadata information. Set to false to return only the State.  The default value is <i>true</i> .

### Result Set Columns

Name	Type	Description
State	<i>String</i>	The current state of bulk export. A value of 'pending' or 'active' indicate Eloqua is still processing the request. A value of 'success' indicates the results are ready to be retrieved. Any other state would indicate an error.
ExportDefinitionName	<i>String</i>	The name used to identify this bulk export definition.
ExportDefinitionFields	<i>String</i>	The fields requested by this bulk export definition.
ExportDefinitionFilter	<i>String</i>	The data filter for this bulk export definition.

ExportDefinitionCreateAt	<i>String</i>	The time the bulk export definition was created.
ExportDefinitionCreateBy	<i>String</i>	The user who created the bulk export definition.
ExportDefinitionUpdateAt	<i>String</i>	The time the bulk export definition was last updated.
ExportDefinitionUpdateBy	<i>String</i>	The user who last updated the bulk export definition.

## GetOAuthAccessToken

Gets an authentication token from Eloqua.

### Input

<b>Name</b>	<b>Type</b>	<b>Description</b>
AuthMode	<i>String</i>	The type of authentication mode to use. The default value is <i>WEB</i> .
CallbackUrl	<i>String</i>	The page to return the Exact Online app after authentication has been completed.
GrantType	<i>String</i>	The type of authorization to be granted for your app. The allowed values are <i>code</i> , <i>password</i> , <i>refresh_token</i> . The default value is <i>code</i> .
State	<i>String</i>	An optional value that has meaning for your App.

## Result Set Columns

Name	Type	Description
OAuthAccessToken	<i>String</i>	The OAuth access token.
OAuthRefreshToken	<i>String</i>	A token that may be used to obtain a new access token.
ExpiresIn	<i>String</i>	The remaining lifetime on the access token.

## GetOAuthAuthorizationUrl

Gets the authorization URL that must be opened separately by the user to grant access to your application. Only needed when developing Web apps.

### Input

Name	Type	Description
CallbackUrl	<i>String</i>	The page to return the user after authorization is complete.
GrantType	<i>String</i>	The type of authorization to be granted for your app. The allowed values are <i>code</i> , <i>password</i> , <i>refresh_token</i> .
State	<i>String</i>	An optional value that has meaning for your App.

## Result Set Columns



Name	Type	Description
URL	<i>String</i>	The authorization url.

## GetSyncLogs

retrieve a sync's logs.

### Input

Name	Type	Description
SyncId	<i>String</i>	Id of the sync.

### Result Set Columns

Name	Type	Description
SyncUri	<i>String</i>	The syncUri of the sync.
Count	<i>String</i>	The index of the log message.
Severity	<i>String</i>	The severity of the sync log.
StatusCode	<i>String</i>	The statusCode of the sync log.
Message	<i>String</i>	The message of the sync log.

---

CreatedAt	<i>String</i>	The created time of the sync log.
-----------	---------------	-----------------------------------

---

## RefreshOAuthAccessToken

Refreshes the OAuth access token used for authentication with Eloqua.

### Input

Name	Type	Description
OAuthRefreshToken	<i>String</i>	The refresh token returned from the original authorization code exchange.

### Result Set Columns

Name	Type	Description
OAuthAccessToken	<i>String</i>	The authentication token returned from Eloqua. This can be used in subsequent calls to other operations for this particular service.
OAuthRefreshToken	<i>String</i>	A token that may be used to obtain a new access token.
ExpiresIn	<i>String</i>	The remaining lifetime on the access token.

## SubscribeContacts

Subscribes a list of contacts to Email Groups.

## Input

Name	Type	Description
EmailGroupID	<i>String</i>	The list of Email Groups to subscribe to. The default is GlobalSubscribe.  The default value is <i>GlobalSubscribe</i> .
ContactIDs	<i>String</i>	The list of Contacts to subscribe to the groups.

## Result Set Columns

Name	Type	Description
State	<i>String</i>	The current state of bulk export. A value of 'pending' or 'active' indicate Eloqua is still processing the request. A value of 'success' indicates the results are ready to be retrieved. Any other state would indicate an error.

## UnsubscribeContacts

Unsubscribes a list of contacts to Email Groups.

## Input

Name	Type	Description
EmailGroupID	<i>String</i>	The list of Email Groups to subscribe to. The default is

		GlobalSubscribe
		The default value is <i>GlobalSubscribe</i> .
ContactIDs	<i>String</i>	The list of Contacts to subscribe to the groups.

## Result Set Columns

Name	Type	Description
State	<i>String</i>	The current state of bulk export. A value of 'pending' or 'active' indicate Eloqua is still processing the request. A value of 'success' indicates the results are ready to be retrieved. Any other state would indicate an error.

## Connection String Options

The connection string properties are the various options that can be used to establish a connection. This section provides a complete list of the options you can configure in the connection string for this provider. Click the links for further details.

For more information on establishing a connection, see [Basic Tab](#).

## Authentication

Property	Description
<a href="#">AuthScheme</a>	The authentication scheme used. Accepted entries are Basic and OAuth.
<a href="#">Company</a>	The company of the Eloqua account. This field is used to authenticate the user.
<a href="#">User</a>	The Oracle Eloqua user account used to authenticate.

---

<b>Password</b>	The password used to authenticate the user.
-----------------	---

---

## OAuth

---

Property	Description
<a href="#">InitiateOAuth</a>	Set this property to initiate the process to obtain or refresh the OAuth access token when you connect.
<a href="#">OAuthClientId</a>	The client Id assigned when you register your application with an OAuth authorization server.
<a href="#">OAuthClientSecret</a>	The client secret assigned when you register your application with an OAuth authorization server.
<a href="#">OAuthAccessToken</a>	The access token for connecting using OAuth.
<a href="#">OAuthSettingsLocation</a>	The location of the settings file where OAuth values are saved when InitiateOAuth is set to GETANDREFRESH or REFRESH. Alternatively, this can be held in memory by specifying a value starting with memory://.
<a href="#">CallbackURL</a>	The OAuth callback URL to return to when authenticating. This value must match the callback URL you specify in your app settings.
<a href="#">OAuthVerifier</a>	The verifier code returned from the OAuth authorization URL.
<a href="#">OAuthRefreshToken</a>	The OAuth refresh token for the corresponding OAuth access token.
<a href="#">OAuthExpiresIn</a>	The lifetime in seconds of the OAuth AccessToken.
<a href="#">OAuthTokenTimestamp</a>	The Unix epoch timestamp in milliseconds when the current Access Token was created.

---

## SSL

---

Property	Description
<a href="#">SSLServerCert</a>	The certificate to be accepted from the server when connecting using TLS/SSL.

## Firewall

Property	Description
<a href="#">FirewallType</a>	The protocol used by a proxy-based firewall.
<a href="#">FirewallServer</a>	The name or IP address of a proxy-based firewall.
<a href="#">FirewallPort</a>	The TCP port for a proxy-based firewall.
<a href="#">FirewallUser</a>	The user name to use to authenticate with a proxy-based firewall.
<a href="#">FirewallPassword</a>	A password used to authenticate to a proxy-based firewall.

## Proxy

Property	Description
<a href="#">ProxyAutoDetect</a>	This indicates whether to use the system proxy settings or not. This takes precedence over other proxy settings, so you'll need to set ProxyAutoDetect to FALSE in order use custom proxy settings.
<a href="#">ProxyServer</a>	The hostname or IP address of a proxy to route HTTP traffic through.
<a href="#">ProxyPort</a>	The TCP port the ProxyServer proxy is running on.
<a href="#">ProxyAuthScheme</a>	The authentication type to use to authenticate to the ProxyServer proxy.

<a href="#">ProxyUser</a>	A user name to be used to authenticate to the ProxyServer proxy.
<a href="#">ProxyPassword</a>	A password to be used to authenticate to the ProxyServer proxy.
<a href="#">ProxySSLType</a>	The SSL type to use when connecting to the ProxyServer proxy.
<a href="#">ProxyExceptions</a>	A semicolon separated list of destination hostnames or IPs that are exempt from connecting through the ProxyServer .

## Logging

Property	Description
<a href="#">LogModules</a>	Core modules to be included in the log file.

## Schema

Property	Description
<a href="#">Location</a>	A path to the directory that contains the schema files defining tables, views, and stored procedures.

## Miscellaneous

Property	Description
<a href="#">BulkPollingInterval</a>	The time interval between requests that check the availability of the bulk query response. The default value is 200 ms.
<a href="#">BulkQueryTimeout</a>	The timeout in minutes for which the provider will wait for a bulk query response. The default value is 25

	minutes.
<a href="#">DataRetentionDuration</a>	The length of time (in hours) that bulk data is stored on the server. Valid values are from 1 hour to 2 weeks. The default value is 24 hours.
<a href="#">DisplayColumnName</a>	Specifies the display name for columns when using the Oracle Eloqua Bulk API. The possible settings are empty string, 'internal', and 'statement'.
<a href="#">MapDataCardsCaseSensitiveMatch</a>	Whether or not to use case sensitive match in data card mapping.
<a href="#">MapDataCardsRelationship</a>	Comma-separated list of the relationships between the Custom Object tables and the Entity tables.
<a href="#">MaxRows</a>	Limits the number of rows returned rows when no aggregation or group by is used in the query. This helps avoid performance issues at design time.
<a href="#">Other</a>	These hidden properties are used only in specific use cases.
<a href="#">Pagesize</a>	The maximum number of results to return per page from Oracle Eloqua.
<a href="#">Readonly</a>	You can use this property to enforce read-only access to Oracle Eloqua from the provider.
<a href="#">RetryCount</a>	The maximum number of times to retry a Bulk API request that fails due to an HTTP 500 status code (Internal Server Error).
<a href="#">RetryInterval</a>	The time interval between attempts to retry a Bulk API request that failed with an HTTP 500 status code (Internal Server Error).
<a href="#">Timeout</a>	The value in seconds until the timeout error is thrown, canceling the operation.



<a href="#">UseBulkAPI</a>	Whether or not the bulk API is used for retrieving data.
<a href="#">UserDefinedViews</a>	A filepath pointing to the JSON configuration file containing your custom views.
<a href="#">UseSimpleNames</a>	Boolean determining if simple names should be used for tables and columns.

## Authentication

This section provides a complete list of the Authentication properties you can configure in the connection string for this provider.

Property	Description
<a href="#">AuthScheme</a>	The authentication scheme used. Accepted entries are Basic and OAuth.
<a href="#">Company</a>	The company of the Eloqua account. This field is used to authenticate the user.
<a href="#">User</a>	The Oracle Eloqua user account used to authenticate.
<a href="#">Password</a>	The password used to authenticate the user.

## AuthScheme

The authentication scheme used. Accepted entries are Basic and OAuth.

### Possible Values

Basic, OAuth

### Data Type

string

## Default Value

"Basic"

## Remarks

The adapter supports the following authentication mechanisms. See the Getting Started chapter for authentication guides.

- Basic: Set this to authenticate with a Oracle Eloqua company, user, password.
- OAuth: Set this to use oauth authentication. Set [OAuthClientId](#), [OAuthClientSecret](#) to the Oracle Eloqua OAuth credentials. Additionally, set [InitiateOAuth](#) to GETANDREFRESH or REFRESH.

## Company

The company of the Eloqua account. This field is used to authenticate the user.

## Data Type

string

## Default Value

""

## Remarks

This field is used to provide authentication for the user to the Eloqua servers.

## User

The Oracle Eloqua user account used to authenticate.

## Data Type

string

## Default Value

""

## Remarks

Together with [Password](#), this field is used to authenticate against the Oracle Eloqua server.

## Password

The password used to authenticate the user.

## Data Type

string

## Default Value

""

## Remarks

The [User](#) and [Password](#) are together used to authenticate with the server.

## OAuth

This section provides a complete list of the OAuth properties you can configure in the connection string for this provider.

Property	Description
<a href="#">InitiateOAuth</a>	Set this property to initiate the process to obtain or refresh the OAuth access token when you connect.
<a href="#">OAuthClientId</a>	The client Id assigned when you register your application with an

	OAuth authorization server.
<code>OAuthClientSecret</code>	The client secret assigned when you register your application with an OAuth authorization server.
<code>OAuthAccessToken</code>	The access token for connecting using OAuth.
<code>OAuthSettingsLocation</code>	The location of the settings file where OAuth values are saved when <code>InitiateOAuth</code> is set to <code>GETANDREFRESH</code> or <code>REFRESH</code> . Alternatively, this can be held in memory by specifying a value starting with <code>memory://</code> .
<code>CallbackURL</code>	The OAuth callback URL to return to when authenticating. This value must match the callback URL you specify in your app settings.
<code>OAuthVerifier</code>	The verifier code returned from the OAuth authorization URL.
<code>OAuthRefreshToken</code>	The OAuth refresh token for the corresponding OAuth access token.
<code>OAuthExpiresIn</code>	The lifetime in seconds of the OAuth AccessToken.
<code>OAuthTokenTimestamp</code>	The Unix epoch timestamp in milliseconds when the current Access Token was created.

## InitiateOAuth

Set this property to initiate the process to obtain or refresh the OAuth access token when you connect.

### Possible Values

OFF, GETANDREFRESH, REFRESH

### Data Type

string

## Default Value

"OFF"

## Remarks

The following options are available:

1. **OFF:** Indicates that the OAuth flow will be handled entirely by the user. An OAuthAccessToken will be required to authenticate.
2. **GETANDREFRESH:** Indicates that the entire OAuth Flow will be handled by the adapter. If no token currently exists, it will be obtained by prompting the user via the browser. If a token exists, it will be refreshed when applicable.
3. **REFRESH:** Indicates that the adapter will only handle refreshing the OAuthAccessToken. The user will never be prompted by the adapter to authenticate via the browser. The user must handle obtaining the OAuthAccessToken and OAuthRefreshToken initially.

## OAuthClientId

The client Id assigned when you register your application with an OAuth authorization server.

## Data Type

string

## Default Value

""

## Remarks

As part of registering an OAuth application, you will receive the OAuthClientId value, sometimes also called a consumer key, and a client secret, the [OAuthClientSecret](#).

## OAuthClientSecret

The client secret assigned when you register your application with an OAuth authorization server.

## Data Type

string

## Default Value

""

## Remarks

As part of registering an OAuth application, you will receive the [OAuthClientId](#), also called a consumer key. You will also receive a client secret, also called a consumer secret. Set the client secret in the [OAuthClientSecret](#) property.

# OAuthAccessToken

The access token for connecting using OAuth.

## Data Type

string

## Default Value

""

## Remarks

The [OAuthAccessToken](#) property is used to connect using OAuth. The [OAuthAccessToken](#) is retrieved from the OAuth server as part of the authentication process. It has a server-dependent timeout and can be reused between requests.

The access token is used in place of your user name and password. The access token protects your credentials by keeping them on the server.

# OAuthSettingsLocation

The location of the settings file where OAuth values are saved when InitiateOAuth is set to GETANDREFRESH or REFRESH. Alternatively, this can be held in memory by specifying a value starting with memory://.

## Data Type

string

## Default Value

"%APPDATA%\\CData\\Oracle Eloqua Data Provider\\OAuthSettings.txt"

## Remarks

When [InitiateOAuth](#) is set to GETANDREFRESH or REFRESH, the adapter saves OAuth values to avoid requiring the user to manually enter OAuth connection properties and allowing the credentials to be shared across connections or processes.

Alternatively to specifying a file path, memory storage can be used instead. Memory locations are specified by using a value starting with 'memory://' followed by a unique identifier for that set of credentials (ex: memory://user1). The identifier can be anything you choose but should be unique to the user. Unlike with the file based storage, you must manually store the credentials when closing the connection with memory storage to be able to set them in the connection when the process is started again. The OAuth property values can be retrieved with a query to the sys\_connection\_props system table. If there are multiple connections using the same credentials, the properties should be read from the last connection to be closed.

If left unspecified, the default location is "%APPDATA%\\CData\\Oracle Eloqua Data Provider\\OAuthSettings.txt" with %APPDATA% being set to the user's configuration directory:

Platform	%APPDATA%
Windows	The value of the APPDATA environment variable
Mac	~/Library/Application Support
Linux	~/.config

## CallbackURL

The OAuth callback URL to return to when authenticating. This value must match the callback URL you specify in your app settings.

### Data Type

string

### Default Value

""

### Remarks

During the authentication process, the OAuth authorization server redirects the user to this URL. This value must match the callback URL you specify in your app settings.

## OAuthVerifier

The verifier code returned from the OAuth authorization URL.

### Data Type

string

### Default Value

""

### Remarks

The verifier code returned from the OAuth authorization URL. This can be used on systems where a browser cannot be launched such as headless systems.



## Authentication on Headless Machines

See to obtain the [OAuthVerifier](#) value.

Set [OAuthSettingsLocation](#) along with [OAuthVerifier](#). When you connect, the adapter exchanges the [OAuthVerifier](#) for the OAuth authentication tokens and saves them, encrypted, to the specified file. Set [InitiateOAuth](#) to GETANDREFRESH automate the exchange.

Once the OAuth settings file has been generated, you can remove [OAuthVerifier](#) from the connection properties and connect with [OAuthSettingsLocation](#) set.

To automatically refresh the OAuth token values, set [OAuthSettingsLocation](#) and additionally set [InitiateOAuth](#) to REFRESH.

## OAuthRefreshToken

The OAuth refresh token for the corresponding OAuth access token.

### Data Type

string

### Default Value

""

### Remarks

The [OAuthRefreshToken](#) property is used to refresh the [OAuthAccessToken](#) when using OAuth authentication.

## OAuthExpiresIn

The lifetime in seconds of the OAuth AccessToken.

### Data Type

string

## Default Value

""

## Remarks

Pair with OAuthTokenTimestamp to determine when the AccessToken will expire.

## OAuthTokenTimestamp

The Unix epoch timestamp in milliseconds when the current Access Token was created.

## Data Type

string

## Default Value

""

## Remarks

Pair with OAuthExpiresIn to determine when the AccessToken will expire.

## SSL

This section provides a complete list of the SSL properties you can configure in the connection string for this provider.

---

Property	Description
<a href="#">SSLServerCert</a>	The certificate to be accepted from the server when connecting using TLS/SSL.

---

## SSLServerCert

The certificate to be accepted from the server when connecting using TLS/SSL.

## Data Type

string

## Default Value

""

## Remarks

If using a TLS/SSL connection, this property can be used to specify the TLS/SSL certificate to be accepted from the server. Any other certificate that is not trusted by the machine is rejected.

This property can take the following forms:

Description	Example
A full PEM Certificate (example shortened for brevity)	-----BEGIN CERTIFICATE----- MIICHTCCAe4CAQAwDQYJKoZIhvd.....Qw == -----END CERTIFICATE-----
A path to a local file containing the certificate	C:\cert.cer
The public key (example shortened for brevity)	-----BEGIN RSA PUBLIC KEY----- MIGfMA0GCSq.....AQAB -----END RSA PUBLIC KEY-----
The MD5 Thumbprint (hex values can also be either space or colon separated)	34e92226ae0819f2ec14b4a3d904f801c
The SHA1 Thumbprint (hex values can also be either space or colon separated)	bb150d

If not specified, any certificate trusted by the machine is accepted.

Certificates are validated as trusted by the machine based on the System's trust store. The trust store used is the 'javax.net.ssl.trustStore' value specified for the system. If no value is specified for this property, Java's default trust store is used (for example, JAVA\_HOME\lib\security\cacerts).

Use '\*' to signify to accept all certificates. Note that this is not recommended due to security concerns.

## Firewall

This section provides a complete list of the Firewall properties you can configure in the connection string for this provider.

---

Property	Description
<a href="#">FirewallType</a>	The protocol used by a proxy-based firewall.
<a href="#">FirewallServer</a>	The name or IP address of a proxy-based firewall.
<a href="#">FirewallPort</a>	The TCP port for a proxy-based firewall.
<a href="#">FirewallUser</a>	The user name to use to authenticate with a proxy-based firewall.
<a href="#">FirewallPassword</a>	A password used to authenticate to a proxy-based firewall.

---

## FirewallType

The protocol used by a proxy-based firewall.

### Possible Values

NONE, TUNNEL, SOCKS4, SOCKS5

### Data Type

string

## Default Value

"NONE"

## Remarks

This property specifies the protocol that the adapter will use to tunnel traffic through the [FirewallServer](#) proxy. Note that by default, the adapter connects to the system proxy; to disable this behavior and connect to one of the following proxy types, set [ProxyAutoDetect](#) to false.

Type	Default Port	Description
TUNNEL	80	When this is set, the adapter opens a connection to Oracle Eloqua and traffic flows back and forth through the proxy.
SOCKS4	1080	When this is set, the adapter sends data through the SOCKS 4 proxy specified by <a href="#">FirewallServer</a> and <a href="#">FirewallPort</a> and passes the <a href="#">FirewallUser</a> value to the proxy, which determines if the connection request should be granted.
SOCKS5	1080	When this is set, the adapter sends data through the SOCKS 5 proxy specified by <a href="#">FirewallServer</a> and <a href="#">FirewallPort</a> . If your proxy requires authentication, set <a href="#">FirewallUser</a> and <a href="#">FirewallPassword</a> to credentials the proxy recognizes.

To connect to HTTP proxies, use [ProxyServer](#) and [ProxyPort](#). To authenticate to HTTP proxies, use [ProxyAuthScheme](#), [ProxyUser](#), and [ProxyPassword](#).

## FirewallServer

The name or IP address of a proxy-based firewall.

## Data Type

string

## Default Value

""

## Remarks

This property specifies the IP address, DNS name, or host name of a proxy allowing traversal of a firewall. The protocol is specified by [FirewallType](#): Use [FirewallServer](#) with this property to connect through SOCKS or do tunneling. Use [ProxyServer](#) to connect to an HTTP proxy.

Note that the adapter uses the system proxy by default. To use a different proxy, set [ProxyAutoDetect](#) to false.

## FirewallPort

The TCP port for a proxy-based firewall.

## Data Type

int

## Default Value

0

## Remarks

This specifies the TCP port for a proxy allowing traversal of a firewall. Use [FirewallServer](#) to specify the name or IP address. Specify the protocol with [FirewallType](#).

## FirewallUser

The user name to use to authenticate with a proxy-based firewall.

## Data Type

string

## Default Value

""

## Remarks

The [FirewallUser](#) and [FirewallPassword](#) properties are used to authenticate against the proxy specified in [FirewallServer](#) and [FirewallPort](#), following the authentication method specified in [FirewallType](#).

## FirewallPassword

A password used to authenticate to a proxy-based firewall.

## Data Type

string

## Default Value

""

## Remarks

This property is passed to the proxy specified by [FirewallServer](#) and [FirewallPort](#), following the authentication method specified by [FirewallType](#).

## Proxy

This section provides a complete list of the Proxy properties you can configure in the connection string for this provider.

Property	Description
<a href="#">ProxyAutoDetect</a>	This indicates whether to use the system proxy settings or not. This takes precedence over other proxy settings, so you'll need to set

	ProxyAutoDetect to FALSE in order use custom proxy settings.
<a href="#">ProxyServer</a>	The hostname or IP address of a proxy to route HTTP traffic through.
<a href="#">ProxyPort</a>	The TCP port the ProxyServer proxy is running on.
<a href="#">ProxyAuthScheme</a>	The authentication type to use to authenticate to the ProxyServer proxy.
<a href="#">ProxyUser</a>	A user name to be used to authenticate to the ProxyServer proxy.
<a href="#">ProxyPassword</a>	A password to be used to authenticate to the ProxyServer proxy.
<a href="#">ProxySSLType</a>	The SSL type to use when connecting to the ProxyServer proxy.
<a href="#">ProxyExceptions</a>	A semicolon separated list of destination hostnames or IPs that are exempt from connecting through the ProxyServer .

## ProxyAutoDetect

This indicates whether to use the system proxy settings or not. This takes precedence over other proxy settings, so you'll need to set ProxyAutoDetect to FALSE in order use custom proxy settings.

### Data Type

bool

### Default Value

true

### Remarks

This takes precedence over other proxy settings, so you'll need to set ProxyAutoDetect to FALSE in order use custom proxy settings.

NOTE: When this property is set to True, the proxy used is determined as follows:



- A search from the JVM properties (**http.proxy**, **https.proxy**, **socksProxy**, etc.) is performed.
- In the case that the JVM properties don't exist, a search from **java.home/lib/net.properties** is performed.
- In the case that `java.net.useSystemProxies` is set to `True`, a search from **the SystemProxy** is performed.
- In Windows only, an attempt is made to retrieve these properties from the **Internet Options** in the **registry**.

To connect to an HTTP proxy, see [ProxyServer](#). For other proxies, such as SOCKS or tunneling, see [FirewallType](#).

## ProxyServer

The hostname or IP address of a proxy to route HTTP traffic through.

### Data Type

string

### Default Value

""

### Remarks

The hostname or IP address of a proxy to route HTTP traffic through. The adapter can use the HTTP, Windows (NTLM), or Kerberos authentication types to authenticate to an HTTP proxy.

If you need to connect through a SOCKS proxy or tunnel the connection, see [FirewallType](#).

By default, the adapter uses the system proxy. If you need to use another proxy, set [ProxyAutoDetect](#) to `false`.

## ProxyPort

The TCP port the ProxyServer proxy is running on.

## Data Type

int

## Default Value

80

## Remarks

The port the HTTP proxy is running on that you want to redirect HTTP traffic through. Specify the HTTP proxy in [ProxyServer](#). For other proxy types, see [FirewallType](#).

## ProxyAuthScheme

The authentication type to use to authenticate to the ProxyServer proxy.

## Possible Values

BASIC, DIGEST, NONE, NEGOTIATE, NTLM, PROPRIETARY

## Data Type

string

## Default Value

"BASIC"

## Remarks

This value specifies the authentication type to use to authenticate to the HTTP proxy specified by [ProxyServer](#) and [ProxyPort](#).

Note that the adapter will use the system proxy settings by default, without further configuration needed; if you want to connect to another proxy, you will need to set [ProxyAutoDetect](#) to false, in addition to [ProxyServer](#) and [ProxyPort](#). To authenticate, set [ProxyAuthScheme](#) and set [ProxyUser](#) and [ProxyPassword](#), if needed.

The authentication type can be one of the following:

- **BASIC:** The adapter performs HTTP BASIC authentication.
- **DIGEST:** The adapter performs HTTP DIGEST authentication.
- **NEGOTIATE:** The adapter retrieves an NTLM or Kerberos token based on the applicable protocol for authentication.
- **PROPRIETARY:** The adapter does not generate an NTLM or Kerberos token. You must supply this token in the Authorization header of the HTTP request.

If you need to use another authentication type, such as SOCKS 5 authentication, see [FirewallType](#).

## ProxyUser

A user name to be used to authenticate to the ProxyServer proxy.

### Data Type

string

### Default Value

""

### Remarks

The [ProxyUser](#) and [ProxyPassword](#) options are used to connect and authenticate against the HTTP proxy specified in [ProxyServer](#).

You can select one of the available authentication types in [ProxyAuthScheme](#). If you are using HTTP authentication, set this to the user name of a user recognized by the HTTP proxy. If you are using Windows or Kerberos authentication, set this property to a user name in one of the following formats:

```
user@domain
domain\user
```

## ProxyPassword

A password to be used to authenticate to the ProxyServer proxy.

## Data Type

string

## Default Value

""

## Remarks

This property is used to authenticate to an HTTP proxy server that supports NTLM (Windows), Kerberos, or HTTP authentication. To specify the HTTP proxy, you can set [ProxyServer](#) and [ProxyPort](#). To specify the authentication type, set [ProxyAuthScheme](#).

If you are using HTTP authentication, additionally set [ProxyUser](#) and [ProxyPassword](#) to HTTP proxy.

If you are using NTLM authentication, set [ProxyUser](#) and [ProxyPassword](#) to your Windows password. You may also need these to complete Kerberos authentication.

For SOCKS 5 authentication or tunneling, see [FirewallType](#).

By default, the adapter uses the system proxy. If you want to connect to another proxy, set [ProxyAutoDetect](#) to false.

## ProxySSLType

The SSL type to use when connecting to the ProxyServer proxy.

## Possible Values

AUTO, ALWAYS, NEVER, TUNNEL

## Data Type

string

## Default Value

"AUTO"

## Remarks

This property determines when to use SSL for the connection to an HTTP proxy specified by [ProxyServer](#). This value can be AUTO, ALWAYS, NEVER, or TUNNEL. The applicable values are the following:

<b>AUTO</b>	Default setting. If the URL is an HTTPS URL, the adapter will use the TUNNEL option. If the URL is an HTTP URL, the component will use the NEVER option.
<b>ALWAYS</b>	The connection is always SSL enabled.
<b>NEVER</b>	The connection is not SSL enabled.
<b>TUNNEL</b>	The connection is through a tunneling proxy. The proxy server opens a connection to the remote host and traffic flows back and forth through the proxy.

## ProxyExceptions

A semicolon separated list of destination hostnames or IPs that are exempt from connecting through the ProxyServer .

### Data Type

string

### Default Value

""

## Remarks

The [ProxyServer](#) is used for all addresses, except for addresses defined in this property. Use semicolons to separate entries.

Note that the adapter uses the system proxy settings by default, without further configuration needed; if you want to explicitly configure proxy exceptions for this connection, you need to set [ProxyAutoDetect](#) = false, and configure [ProxyServer](#) and

[ProxyPort](#). To authenticate, set [ProxyAuthScheme](#) and set [ProxyUser](#) and [ProxyPassword](#), if needed.

## Logging

This section provides a complete list of the Logging properties you can configure in the connection string for this provider.

---

Property	Description
<a href="#">LogModules</a>	Core modules to be included in the log file.

---

## LogModules

Core modules to be included in the log file.

### Data Type

string

### Default Value

""

### Remarks

Only the modules specified (separated by ';') will be included in the log file. By default all modules are included.

See the [Logging](#) page for an overview.

## Schema

This section provides a complete list of the Schema properties you can configure in the connection string for this provider.

Property	Description
<a href="#">Location</a>	A path to the directory that contains the schema files defining tables, views, and stored procedures.

## Location

A path to the directory that contains the schema files defining tables, views, and stored procedures.

## Data Type

string

## Default Value

"%APPDATA%\\CData\\Oracle Eloqua Data Provider\\Schema"

## Remarks

The path to a directory which contains the schema files for the adapter (.rsd files for tables and views, .rsb files for stored procedures). The folder location can be a relative path from the location of the executable. The [Location](#) property is only needed if you want to customize definitions (for example, change a column name, ignore a column, and so on) or extend the data model with new tables, views, or stored procedures.

If left unspecified, the default location is "%APPDATA%\\CData\\Oracle Eloqua Data Provider\\Schema" with **%APPDATA%** being set to the user's configuration directory:

Platform	%APPDATA%
Windows	The value of the APPDATA environment variable
Mac	~/Library/Application Support
Linux	~/.config

## Miscellaneous

This section provides a complete list of the Miscellaneous properties you can configure in the connection string for this provider.

Property	Description
<a href="#">BulkPollingInterval</a>	The time interval between requests that check the availability of the bulk query response. The default value is 200 ms.
<a href="#">BulkQueryTimeout</a>	The timeout in minutes for which the provider will wait for a bulk query response. The default value is 25 minutes.
<a href="#">DataRetentionDuration</a>	The length of time (in hours) that bulk data is stored on the server. Valid values are from 1 hour to 2 weeks. The default value is 24 hours.
<a href="#">DisplayColumnName</a>	Specifies the display name for columns when using the Oracle Eloqua Bulk API. The possible settings are empty string, 'internal', and 'statement'.
<a href="#">MapDataCardsCaseSensitiveMatch</a>	Whether or not to use case sensitive match in data card mapping.
<a href="#">MapDataCardsRelationship</a>	Comma-separated list of the relationships between the Custom Object tables and the Entity tables.
<a href="#">MaxRows</a>	Limits the number of rows returned rows when no aggregation or group by is used in the query. This helps avoid performance issues at design time.
<a href="#">Other</a>	These hidden properties are used only in specific use cases.
<a href="#">Pagesize</a>	The maximum number of results to return per page from Oracle Eloqua.



<a href="#">Readonly</a>	You can use this property to enforce read-only access to Oracle Eloqua from the provider.
<a href="#">RetryCount</a>	The maximum number of times to retry a Bulk API request that fails due to an HTTP 500 status code (Internal Server Error).
<a href="#">RetryInterval</a>	The time interval between attempts to retry a Bulk API request that failed with an HTTP 500 status code (Internal Server Error).
<a href="#">Timeout</a>	The value in seconds until the timeout error is thrown, canceling the operation.
<a href="#">UseBulkAPI</a>	Whether or not the bulk API is used for retrieving data.
<a href="#">UserDefinedViews</a>	A filepath pointing to the JSON configuration file containing your custom views.
<a href="#">UseSimpleNames</a>	Boolean determining if simple names should be used for tables and columns.

## BulkPollingInterval

The time interval between requests that check the availability of the bulk query response. The default value is 200 ms.

### Data Type

string

### Default Value

"200"

### Remarks

The time interval between requests that check the availability of the bulk query response. When [UseBulkAPI](#) is set, the adapter requests Eloqua to prepare a response to the query. It

then waits for the response to be ready by periodically polling the server to check status. This property controls the frequency of polling.

## BulkQueryTimeout

The timeout in minutes for which the provider will wait for a bulk query response. The default value is 25 minutes.

### Data Type

string

### Default Value

"25"

### Remarks

The timeout in minutes for which the adapter will wait for a bulk query response. The default value is 25 minutes. When [UseBulkAPI](#) is set, the adapter requests Eloqua to prepare a response to the query. It then waits for the response to be ready by periodically polling the server to check status. This property controls the total time the adapter will wait for a response.

Note that this property is very different from [Timeout](#). The [Timeout](#) is an inactivity timeout that controls the time to wait for any response. This property controls the total length of time to wait for a bulk query to execute. ;

## DataRetentionDuration

The length of time (in hours) that bulk data is stored on the server. Valid values are from 1 hour to 2 weeks. The default value is 24 hours.

### Data Type

string

## Default Value

"24"

## Remarks

The length of time (in hours) that bulk data is stored on the server. Valid values are from 1 hour to 2 weeks. The default value is 24 hours.

## DisplayColumnName

Specifies the display name for columns when using the Oracle Eloqua Bulk API. The possible settings are empty string, 'internal', and 'statement'.

## Data Type

string

## Default Value

""

## Remarks

The response data Oracle Eloqua Bulk API Fields contains three kinds of displayName: "name", "statement", and "internalName" for the same field.

The display name type chosen is simply a matter of visual preference.

As an example, suppose you have the following JSON response for an EmailAddress item.

```
{
  "items": [
    {
      "name": "Email Address",
      "internalName": "C_EmailAddress",
      "dataType": "emailAddress",
      "hasReadOnlyConstraint": false,
      "hasNotNullConstraint": false,
      "hasUniquenessConstraint": true,
    }
  ]
}
```

```

        "statement": "{{Contact.Field(C_EmailAddress)}}",
        "uri": "/contacts/fields/100001",
        "createdAt": "1900-01-01T05:00:00.0000000Z",
        "updatedAt": "1900-01-01T05:00:00.0000000Z"
    },
    .....
]
}

```

The column name listed upon metadata discovery from the adapter will be one of the following.

- Empty String: The default value is empty string. This means the "name" of the field will be regarded as the display column name: **EmailAddress**.
- internal: This means the "internalName" of the field will be regarded as the display column name: **C\_EmailAddress**.
- statement: It means the "statement" of the field will be regarded as the display column name: **{{Contact.Field(C\_EmailAddress)}}**.

## MapDataCardsCaseSensitiveMatch

Whether or not to use case sensitive match in data card mapping.

### Data Type

bool

### Default Value

false

### Remarks

Whether or not to use case sensitive match in data card mapping. Only has an effect if [MapDataCardsRelationship](#) is set.

## MapDataCardsRelationship

Comma-separated list of the relationships between the Custom Object tables and the Entity tables.

## Data Type

string

## Default Value

""

## Remarks

Comma-separated list of the relationships between the Custom Object tables and the Entity tables. The format of these relationships is '<custom\_object>.<source\_field>=<entity\_type>.<entity\_field>'. For example: Custom\_MyCustomObject.Email\_Address=Contact.C\_EmailAddress

## MaxRows

Limits the number of rows returned rows when no aggregation or group by is used in the query. This helps avoid performance issues at design time.

## Data Type

int

## Default Value

-1

## Remarks

Limits the number of rows returned rows when no aggregation or group by is used in the query. This helps avoid performance issues at design time.

## Other

These hidden properties are used only in specific use cases.

## Data Type

string

## Default Value

""

## Remarks

The properties listed below are available for specific use cases. Normal driver use cases and functionality should not require these properties.

Specify multiple properties in a semicolon-separated list.

## Integration and Formatting

DefaultColumnSize	Sets the default length of string fields when the data source does not provide column length in the metadata. The default value is 2000.
ConvertDateTimeToGMT	Determines whether to convert date-time values to GMT, instead of the local time of the machine.
RecordToFile=filename	Records the underlying socket data transfer to the specified file.

## Pagesize

The maximum number of results to return per page from Oracle Eloqua.

## Data Type

int

## Default Value

500

## Remarks

The Pagesize property affects the maximum number of results to return per page from Oracle Eloqua. Setting a higher value may result in better performance at the cost of additional memory allocated per page consumed.

## Readonly

You can use this property to enforce read-only access to Oracle Eloqua from the provider.

## Data Type

bool

## Default Value

false

## Remarks

If this property is set to true, the adapter will allow only SELECT queries. INSERT, UPDATE, DELETE, and stored procedure queries will cause an error to be thrown.

## RetryCount

The maximum number of times to retry a Bulk API request that fails due to an HTTP 500 status code (Internal Server Error).

## Data Type

string

## Default Value

"5"

## Remarks

When [UseBulkAPI](#) is set to "Auto" or "True", the adapter will attempt to retry any requests that fail due to an HTTP 500 status code (Internal Server Error). This property defines maximum number of time the adapter will retry a Bulk API request that failed. By default the adapter will retry the request 5 times.

## RetryInterval

The time interval between attempts to retry a Bulk API request that failed with an HTTP 500 status code (Internal Server Error).

## Data Type

string

## Default Value

"200"

## Remarks

When [UseBulkAPI](#) is set to "Auto" or "True", the adapter will attempt to retry any requests that fail due to an HTTP 500 status code (Internal Server Error). This property defines the time interval between attempts to retry a Bulk API request that failed. The default value is 200 ms.

## Timeout

The value in seconds until the timeout error is thrown, canceling the operation.

## Data Type

int

## Default Value

60



## Remarks

If Timeout = 0, operations do not time out. The operations run until they complete successfully or until they encounter an error condition.

If Timeout expires and the operation is not yet complete, the adapter throws an exception.

## UseBulkAPI

Whether or not the bulk API is used for retrieving data.

### Possible Values

Auto, False, True

### Data Type

string

### Default Value

"Auto"

## Remarks

Whether or not the bulk API is used for retrieving data. When UseBulkAPI is set to "True", the adapter will always attempt to use the bulk API. However, there are several restrictions to accessible tables and columns. See [Data Model](#) for more information.

When set to "False", the adapter will use the REST API for all requests. When set to "Auto", the adapter will use whichever API is most appropriate for the request. For example, consider the following query:

```
SELECT * FROM Activity_EmailOpen
```

In this case, the adapter will use the Bulk API (because the ContactId is not specified), whereas the following query will use the REST API:

```
SELECT * FROM Activity_EmailOpen ContactId='...'
```

Using the Bulk API starts with the adapter sending a request to Eloqua to prepare a response to the query. It then waits for the response to be ready by periodically polling the

server to check status. [BulkPollingInterval](#) and [BulkQueryTimeout](#) control the frequency and duration of polling respectively.

## UserDefinedViews

A filepath pointing to the JSON configuration file containing your custom views.

### Data Type

string

### Default Value

""

### Remarks

User Defined Views are defined in a JSON-formatted configuration file called *UserDefinedViews.json*. The adapter automatically detects the views specified in this file.

You can also have multiple view definitions and control them using the [UserDefinedViews](#) connection property. When you use this property, only the specified views are seen by the adapter.

This User Defined View configuration file is formatted as follows:

- Each root element defines the name of a view.
- Each root element contains a child element, called **query**, which contains the custom SQL query for the view.

For example:

```
{
  "MyView": {
    "query": "SELECT * FROM Contact WHERE MyColumn = 'value'"
  },
  "MyView2": {
    "query": "SELECT * FROM MyTable WHERE Id IN (1,2,3)"
  }
}
```

Use the [UserDefinedViews](#) connection property to specify the location of your JSON configuration file. For example:

```
"UserDefinedViews",  
"C:\\Users\\yourusername\\Desktop\\tmp\\UserDefinedViews.json"
```

## UseSimpleNames

Boolean determining if simple names should be used for tables and columns.

### Data Type

bool

### Default Value

false

### Remarks

Oracle Eloqua tables and columns can use special characters in names that are normally not allowed in standard databases. UseSimpleNames makes the adapter easier to use with traditional database tools.

Setting UseSimpleNames to true will simplify the names of tables and columns returned. It will enforce a naming scheme such that only alphanumeric characters and the underscore are valid for the displayed table and column names. Any nonalphanumeric characters will be converted to an underscore.

# TIBCO Product Documentation and Support Services

---

For information about this product, you can read the documentation, contact TIBCO Support, and join the TIBCO Community.

## How to Access TIBCO Documentation

Documentation for TIBCO products is available on the [TIBCO Product Documentation](#) website, mainly in HTML and PDF formats.

The [TIBCO Product Documentation](#) website is updated frequently and is more current than any other documentation included with the product.

## Product-Specific Documentation

The following documentation for this product is available on the [TIBCO Data Virtualization](#) page.

- **Users**
  - TDV Getting Started Guide
  - TDV User Guide
  - TDV Web UI User Guide
  - TDV Client Interfaces Guide
  - TDV Tutorial Guide
  - TDV Northbay Example
- **Administration**
  - TDV Installation and Upgrade Guide
  - TDV Administration Guide
  - TDV Active Cluster Guide
  - TDV Security Features Guide

- **Data Sources**

- TDV Adapter Guides

- TDV Data Source Toolkit Guide (Formerly Extensibility Guide)

- **References**

- TDV Reference Guide

- TDV Application Programming Interface Guide

- **Other**

- TDV Business Directory Guide

- TDV Discovery Guide

- *TIBCO TDV and Business Directory Release Notes* Read the release notes for a list of new and changed features. This document also contains lists of known issues and closed issues for this release.

## How to Contact TIBCO Support

Get an overview of [TIBCO Support](#). You can contact TIBCO Support in the following ways:

- For accessing the Support Knowledge Base and getting personalized content about products you are interested in, visit the [TIBCO Support](#) website.
- For creating a Support case, you must have a valid maintenance or support contract with TIBCO. You also need a user name and password to log in to [TIBCO Support](#) website. If you do not have a user name, you can request one by clicking **Register** on the website.

## Release Version Support

TDV 8.5 is designated as a Long Term Support (LTS) version. Some release versions of TIBCO Data Virtualization products are selected to be long-term support (LTS) versions. Defect corrections will typically be delivered in a new release version and as hotfixes or service packs to one or more LTS versions. See also

[https://docs.tibco.com/pub/tdv/general/LTS/tdv\\_LTS\\_releases.htm](https://docs.tibco.com/pub/tdv/general/LTS/tdv_LTS_releases.htm).

## How to Join TIBCO Community

TIBCO Community is the official channel for TIBCO customers, partners, and employee subject matter experts to share and access their collective experience. TIBCO Community offers access to Q&A forums, product wikis, and best practices. It also offers access to extensions, adapters, solution accelerators, and tools that extend and enable customers to gain full value from TIBCO products. In addition, users can submit and vote on feature requests from within the [TIBCO Ideas Portal](#). For a free registration, visit [TIBCO Community](#).

# Legal and Third-Party Notices

---

SOME TIBCO SOFTWARE EMBEDS OR BUNDLES OTHER TIBCO SOFTWARE. USE OF SUCH EMBEDDED OR BUNDLED TIBCO SOFTWARE IS SOLELY TO ENABLE THE FUNCTIONALITY (OR PROVIDE LIMITED ADD-ON FUNCTIONALITY) OF THE LICENSED TIBCO SOFTWARE. THE EMBEDDED OR BUNDLED SOFTWARE IS NOT LICENSED TO BE USED OR ACCESSED BY ANY OTHER TIBCO SOFTWARE OR FOR ANY OTHER PURPOSE.

USE OF TIBCO SOFTWARE AND THIS DOCUMENT IS SUBJECT TO THE TERMS AND CONDITIONS OF A LICENSE AGREEMENT FOUND IN EITHER A SEPARATELY EXECUTED SOFTWARE LICENSE AGREEMENT, OR, IF THERE IS NO SUCH SEPARATE AGREEMENT, THE CLICKWRAP END USER LICENSE AGREEMENT WHICH IS DISPLAYED DURING DOWNLOAD OR INSTALLATION OF THE SOFTWARE (AND WHICH IS DUPLICATED IN THE LICENSE FILE) OR IF THERE IS NO SUCH SOFTWARE LICENSE AGREEMENT OR CLICKWRAP END USER LICENSE AGREEMENT, THE LICENSE(S) LOCATED IN THE “LICENSE” FILE(S) OF THE SOFTWARE. USE OF THIS DOCUMENT IS SUBJECT TO THOSE TERMS AND CONDITIONS, AND YOUR USE HEREOF SHALL CONSTITUTE ACCEPTANCE OF AND AN AGREEMENT TO BE BOUND BY THE SAME.

This document is subject to U.S. and international copyright laws and treaties. No part of this document may be reproduced in any form without the written authorization of TIBCO Software Inc.

TIBCO, TIBCO logo, Two-Second Advantage, TIBCO Spotfire, TIBCO ActiveSpaces, TIBCO Spotfire Developer, TIBCO EMS, TIBCO Spotfire Automation Services, TIBCO Enterprise Runtime for R, TIBCO Spotfire Server, TIBCO Spotfire Web Player, TIBCO Spotfire Statistics Services, S-PLUS, and TIBCO Spotfire S+ are either registered trademarks or trademarks of TIBCO Software Inc. in the United States and/or other countries.

Java and all Java based trademarks and logos are trademarks or registered trademarks of Oracle Corporation and/or its affiliates.

All other product and company names and marks mentioned in this document are the property of their respective owners and are mentioned for identification purposes only.

This software may be available on multiple operating systems. However, not all operating system platforms for a specific software version are released at the same time. See the readme file for the availability of this software version on a specific operating system platform.

THIS DOCUMENT IS PROVIDED “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

THIS DOCUMENT COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THIS DOCUMENT. TIBCO SOFTWARE INC. MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S) AND/OR THE PROGRAM(S) DESCRIBED IN THIS DOCUMENT AT ANY TIME.

THE CONTENTS OF THIS DOCUMENT MAY BE MODIFIED AND/OR QUALIFIED, DIRECTLY OR INDIRECTLY, BY OTHER DOCUMENTATION WHICH ACCOMPANIES THIS SOFTWARE, INCLUDING BUT NOT LIMITED TO ANY RELEASE NOTES AND "READ ME" FILES.

This and other products of TIBCO Software Inc. may be covered by registered patents. Please refer to TIBCO's Virtual Patent Marking document (<https://www.tibco.com/patents>) for details.

Copyright © 2002-2023 Cloud Software Group, Inc All Rights Reserved.