TIBCO Data Virtualization®

SharePoint Adapter Guide

Version 8.4

***Last Updated:** January 25, 2021*

Two-Second Advantage®

TIBCO®

Important Information

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 DOCUMENTATION 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 DOCUMENTATION IS SUBJECT TO THOSE TERMS AND CONDITIONS, AND YOUR USE HEREOF SHALL CONSTITUTE ACCEPTANCE OF AND AN AGREEMENT TO BE BOUND BY THE SAME.

TIBCO and the TIBCO logo are either registered trademarks or trademarks of TIBCO Software Inc. in the United States and/or other countries

TIBCO, 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.

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 DOCUMENTATION 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 DOCUMENTATION 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 DOCUMENTATION. TIBCO SOFTWARE INC. MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S) AND/OR THE PROGRAM(S) DESCRIBED IN THIS DOCUMENTATION AT ANY TIME.

Contents

Preface

Forinformationonthefollowing,seethe *TDVUserGuide*:

- AddingaDataSource
- IntrospectingaDataSource
- TestingtheConnectiontoYourDataSource

Documentation for this and other TIBCO products is available on the TIBCO Documentation site. This site is updated more frequently than any documentation that might be included with the product. To ensure that you are accessing the latest available help topics, please visit:

- https://docs.tibco.com

Product-Specific Documentation

The following documents form the TIBCO® Data Virtualization(TDV) documentation set:

- Users
  - TDV Getting Started Guide
  - TDV 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 Access TIBCO Documentation

Documentation for TIBCO products is available on the TIBCO Product Documentation website mainly in the HTML and PDF formats.

The TIBCO Product Documentation website is updated frequently and is more current than any other documentation included with the product. To access the latest documentation, visit https://docs.tibco.com.

Documentation for TIBCO Data Virtualization is available on https://docs.tibco.com/products/tibco-data-virtualization-server.

How to Contact TIBCO Support

You can contact TIBCO Support in the following ways:

- For an overview of TIBCO Support, visit https://www.tibco.com/services/support.
- For accessing the Support Knowledge Base and getting personalized content about products you are interested in, visit the TIBCO Support portal at https://support.tibco.com.
- 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 https://support.tibco.com. If you do not have a user name, you can request one by clicking **Register** on the website.

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, go to https://community.tibco.com.

TDV SharePoint Adapter

SharePoint Version Support

The adapter supports all versions of SharePoint that support the SOAP API. This includes: Windows SharePoint Services 3.0, SharePoint Server 2007+ (2010, 2013, etc.), and SharePoint Online. The adapter models the custom lists of your SharePoint site as bidirectional tables; when you connect, the adapter retrieves the metadata for these tables by calling SharePoint Web services. Supported authentication schemes are NTLM, Basic, Digest, Forms, Kerberos, SSO, STS (security token services), and SharePoint authentication cookies.

SQL Compliance

The SQL Compliance section shows the SQL syntax supported by the adapter and points out any limitations.

Getting Started

Deploying the Adapter

For instructions on deploying the adapter, refer to the *Installation Guide, section Installing the Advanced Adapters*.

Connecting to SharePoint

Basic Tab shows how to authenticate to SharePoint 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.

Basic Tab

Connecting to SharePoint

Set the URL to a Site Collection to work with all Lists and Documents in all nested Subsites. Set the URL to a specific Site to work with Lists and Documents in that Site only.

URL	Example URL
Site	https://teams.contoso.com/teamA or https://teamA.contoso.com
Site Collection	https://teams.contoso.com

In addition to providing the URL, use one of the following sets of connection properties to authenticate to SharePoint. The default values make it easy to connect in most environments, as shown below.

Authenticating to SharePoint Online

Set SharePointEdition to "SharePoint Online" and set the User and Password to the credentials you use to log onto SharePoint; for example, the credentials to your Microsoft Online Services account.

The following SSO (single sign-on) identity providers are also supported: Azure Active Directory, OneLogin, and OKTA.

- SSO

If you have enabled SSO in SharePoint Online, set UseSSO to true in addition to your User and Password.

If the user account domain is different from the domain configured with the identity provider, set SSODomain to the domain configured with the identity provider. This property may be required for AD FS, OneLogin, and OKTA.

Authenticating to SharePoint On Premises

Set SharePointEdition to "SharePoint OnPremise" to use the following authentication types.

Windows (NTLM)

This is the most common authentication type. As such, the adapter is preconfigured to use NTLM as the default; simply set the Windows User and Password to connect.

Kerberos and Kerberos Delegation

To authenticate with Kerberos, set AuthScheme to NEGOTIATE. If needed, provide the User and Password. To use Kerberos Delegation, set AuthScheme to KERBEROSDELEGATION.

KerberosKDC, KerberosSPN, and KerberosRealm enable control over the components of Kerberos authentication.

Forms

This allows authentication through a custom authentication method, instead of Active Directory. To use this authentication type, set AuthScheme to FORMS and set the User and Password.

Connect using MSI Authentication

If you are running SharePoint on an Azure VM, you can leverage Managed Service Identity (MSI) credentials to connect:

- AuthScheme: Set this to **AzureMSI**.

The MSI credentials will then be automatically obtained for authentication.

Advanced Tab

The connection string properties describe the various options that can be used to establish a connection.

Connection String Options

The following is the full list of the options you can configure in the connection string for this provider.

Auth Scheme	The scheme used for authenticating to SharePoint. For On-Premise instances, accepted entries are NTLM, Basic, Digest, Forms, None, and Negotiate. NTLM is the default. For Sharepoint Online, the only accepted entry is OAuth.

Azure Environment	The Azure Environment to use when establishing a connection.
Azure Tenant	The Microsoft Online tenant being used to access data. If not specified, your default tentant will be used.
Calculated Data Type	The data type to be used for calculated fields.
Callback URL	The OAuth callback URL to return to when authenticating. This value must match the callback URL you specify in your Add-In settings.
Continue On Error	Indicates whether or not to continue updating items in a batch after an error.
Create ID Columns	Indicates whether or not to create supplemental ID columns for SharePoint columns that use values from information stored in other Lists.
Firewall Password	A password used to authenticate to a proxy-based firewall.
Firewall Port	The TCP port for a proxy-based firewall.
Firewall Server	The name or IP address of a proxy-based firewall.
Firewall Type	The protocol used by a proxy-based firewall.
Firewall User	The user name to use to authenticate with a proxy-based firewall.
Folder Option	An option to determine how to display folders in results. Enter either FilesOnly, FilesAndFolders, Recursive, or RecursiveAll.
Include Lookup Columns	This option controls whether the driver returns the lookup columns defined on a table.
Initiate OAuth	Set this property to initiate the process to obtain or refresh the OAuth access token when you connect.
Kerberos KDC	The Kerberos Key Distribution Center (KDC) service used to authenticate the user.
Kerberos Keytab File	The Keytab file containing your pairs of Kerberos principals and encrypted keys.

Kerberos Realm	The Kerberos Realm used to authenticate the user with.
Kerberos Service Realm	The Kerberos realm of the service.
Kerberos SPN	The Service Principal Name for the Kerberos Domain Controller.
Kerberos Ticket Cache	The full file path to an MIT Kerberos credential cache file.
Location	A path to the directory that contains the schema files defining tables, views, and stored procedures.
Log Modules	Core modules to be included in the log file.
Max Rows	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.
OAuth Access Token	The access token for connecting using OAuth.
OAuth Client Id	The client ID assigned when you register your application with an OAuth authorization server.
OAuth Client Secret	The client secret assigned when you register your application with an OAuth authorization server.
OAuth Expires In	The lifetime in seconds of the OAuth AccessToken.
OAuth Grant Type	The grant type for the OAuth flow.
OAuth Settings Location	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.
OAuth Token Timestamp	The Unix epoch timestamp in milliseconds when the current Access Token was created.
OAuth Verifier	The verifier code returned from the OAuth authorization URL.
Other	These hidden properties are used only in specific use cases.

Page Size	The number of results to return per page of data retrieved from SharePoint.
Password	The password used to authenticate the user.
Proxy Auth Scheme	The authentication type to use to authenticate to the ProxyServer proxy.
Proxy Auto Detect	This indicates whether to use the system proxy settings or not. Set ProxyAutoDetect to FALSE to use custom proxy settings. This takes precedence over other proxy settings.
Proxy Exceptions	A semicolon separated list of hosts or IPs that will be exempt from connecting through the ProxyServer .
Proxy Password	A password to be used to authenticate to the ProxyServer proxy.
Proxy Port	The TCP port the ProxyServer proxy is running on.
Proxy Server	The hostname or IP address of a proxy to route HTTP traffic through.
Proxy SSL Type	The SSL type to use when connecting to the ProxyServer proxy.
Proxy User	A user name to be used to authenticate to the ProxyServer proxy.
Readonly	You can use this property to enforce read-only access to SharePoint from the provider.
Schema	The type of schema to use.
Share Point Edition	The edition of SharePoint being used. Set either SharePoint Online or SharePoint On-Premise.
Show Hidden Columns	Boolean determining if hidden columns should be shown or not. If false, all hidden columns will be removed from the column listing.
Show Predefined Columns	Boolean determining if predefined columns should be shown or not. If false, all columns derived from a base type will be removed from the column listing.
Show Version Views	Indicate whether to display the view of list versions. Such as ListA_Versions.
SSL Client Cert	The TLS/SSL client certificate store for SSL Client Authentication (2-way SSL).

SSL Client Cert Password	The password for the TLS/SSL client certificate.
SSL Client Cert Subject	The subject of the TLS/SSL client certificate.
SSL Client Cert Type	The type of key store containing the TLS/SSL client certificate.
SSL Server Cert	The certificate to be accepted from the server when connecting using TLS/SSL.
SSO Domain	The domain of the user when using single sign-on (SSO).
SSO Login URL	The identity provider's login URL.
SSO Password	The password of the SSOProvider used to authenticate the user.
SSO Properties	Additional properties required to connect to the identity provider in a semicolon-separated list.
SSO Provider	The name of the SSO provider you are trying to authenticate.
SSO User	The SSOProvider user account used to authenticate.
STSURL	The URL of the security token service (STS) when using single sign-on (SSO).
Timeout	The value in seconds until the timeout error is thrown, canceling the operation.
URL	The base URL for the site.
Use Display Names	Boolean determining if the display names for the columns should be used instead of the API names.
Use NTLMV1	Determines whether the driver will attempt to connect with NTLMv1 or NTLMv2 (default).
User	The SharePoint user account used to authenticate.
Use Simple Names	Boolean determining if simple names should be used for tables and columns.
Use SSO	Whether or not to use single sign-on (SSO) to authenticate to SharePoint Online.

Auth Scheme

The scheme used for authenticating to SharePoint On-Premise instances. Accepted entries are NTLM, BASIC, DIGEST, FORMS, NONE, NEGOTIATE, and KERBEROSDELEGATION. NTLM is the default.

Data Type

string

Default Value

"NTLM"

Remarks

Together with Password and User, this field is used to authenticate against the server. NTLM is the default option. Use the following options to select your authentication scheme:

NTLM: Set this to use your Windows credentials for authentication.

NEGOTIATE: If

AuthScheme is set to NEGOTIATE, the adapter will negotiate an authentication mechanism with the server. Set AuthScheme to NEGOTIATE if you want to use Kerberos authentication.

KERBEROSDELEGATION: Set this to use delegation through the Kerberos protocol. Set the User and Password of the account you want to impersonate.

NONE: Set this to use anonymous authentication; for example, to access a public site.

FORMS: Set this if your SharePoint instance uses a custom authentication method through a Web form.

DIGEST: Set this to use HTTP Digest authentication.

BASIC: Set this to use HTTP Basic authentication.

AzureMSI: Set this to automatically obtain Managed Service Identity credentials when running on an Azure VM.

If authenticating to Sharepoint Online, set AuthScheme to OAuth.

Azure Environment

The Azure Environment to use when establishing a connection.

Data Type

string

Default Value

"GLOBAL"

Remarks

In most cases, leaving the environment set to global will work. However, if your Azure Account has been added to a different environment, the AzureEnvironment may be used to specify which environment.

Azure Tenant

The Microsoft Online tenant being used to access data. If not specified, your default tentant will be used.

Data Type

string

Default Value

""

Remarks

The Microsoft Online tenant being used to access data. For instance, contoso.onmicrosoft.com. Alternatively, specify the tenant id. This value is the directory Id in the Azure Portal > Azure Active Directory > Properties.

Typically it is not necessary to specify the Tenant. This can be automatically determined by Microsoft when using the OAuthGrantType set to CODE (default). However, it may fail in the case that the user belongs to multiple tenants. For instance, if an Admin of domain A invites a user of domain B to be a guest user. The user will now belong to both tenants. It is a good practice to specify the Tenant, although in general things should normally work without having to specify it.

The AzureTenant is required when setting OAuthGrantType to CLIENT. When using client credentials, there is no user context. The credentials are taken from the context of the app itself. While Microsoft still allows client credentials to be obtained without specifying which Tenant, it has a much lower probability of picking the specific tenant you want to work with. For this reason, we require AzureTenant to be explicitly stated for all client credentials connections to ensure you get credentials that are applicable for the domain you intend to connect to.

Calculated Data Type

The data type to be used for calculated fields.

Data Type

string

Default Value

""

Remarks

The data type to be used for calculated fields. By default, the data type is determined by the type of calculated field in SharePoint. However, in some cases these calculated fields may return values that are not appropriate for the specified type. In these instances, you may wish to set the Calculated Data Type to String.

Callback URL

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

Continue On Error

Indicates whether or not to continue updating items in a batch after an error.

Data Type

bool

Default Value

true

Remarks

If this property is set to True (default), the adapter will continue adding, updating, or deleting items when an error is encountered on one of the items. When set to False, the adapter will stop adding, updating, or deleting items after an error is encountered (entries preceeding the problematic entry will still be added, updated, or deleted).

Create ID Columns

Indicates whether or not to create supplemental ID columns for SharePoint columns that use values from information stored in other Lists.

Data Type

bool

Default Value

true

Remarks

Indicates whether or not to create supplemental ID columns for SharePoint columns that use values from information stored in other Lists (like "Lookup" or "Person or Group" columns). The ID column that is created will contain the related entry's ID (in the context of its original List). If set to false, the ID columns will not be created, the ID will be ignored, and only the value of the referenced column will be returned.

Firewall Password

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.

Firewall Port

The TCP port for a proxy-based firewall.

Data Type

string

Default Value

""

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.

Firewall Server

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.

Firewall Type

The protocol used by a proxy-based firewall.

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 SharePoint 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 FirewallServer and FirewallPort and passes the FirewallUser 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 FirewallServer and FirewallPort. If your proxy requires authentication, set FirewallUser and FirewallPassword to credentials the proxy recognizes.

To connect to HTTP proxies, use ProxyServer and ProxyPort. To authenticate to HTTP proxies, use ProxyAuthScheme, ProxyUser, and ProxyPassword.

Firewall User

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.

Folder Option

An option to determine how to display folders in results. Enter either FilesOnly, FilesAndFolders, Recursive, or RecursiveAll.

Data Type

string

Default Value

"RecursiveAll"

Remarks

An option to determine how to display folders in results. FilesOnly will display only files in specified lists or libraries. FilesAndFolders will display files and folders in the specified list. RecursiveAll will display files in the specified list and all subfolders.

Include Lookup Columns

This option controls whether the driver returns the lookup columns defined on a table.

Data Type

bool

Default Value

true

Remarks

This option controls whether the driver returns the lookup columns defined on a table. The SharePoint server may reject the request if too many lookup columns are included in a single query.

Initiate OAuth

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

Data Type

string

Default Value

"OFF"

Remarks

The following options are available:

OFF: Indicates that the OAuth flow will be handled entirely by the user. An OAuthAccessToken will be required to authenticate.

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.

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.

Kerberos KDC

The Kerberos Key Distribution Center (KDC) service used to authenticate the user.

Data Type

string

Default Value

""

Remarks

The Kerberos properties are used when using Windows Authentication. The adapter will request session tickets and temporary session keys from the Kerberos Key Distribution Center (KDC) service. The Kerberos Key Distribution Center (KDC) service is conventionally colocated with the domain controller. If Kerberos KDC is not specified the adapter will attempt to detect these properties automatically from the following locations:

Java System Properties: Kerberos settings can be configured in Java using the config file krb5.conf, or using the system properties java.security.krb5.realm and java.security.krb5.kdc. The adapter will use the system settings if KerberosRealm and KerberosKDC are not explicitly set.

Domain Name and Host: The adapter will infer the Kerberos Realm and Kerberos KDC from the configured domain name and host as a last resort.

Note: Windows authentication is supported in JRE 1.6 and above only.

Kerberos Keytab File

The Keytab file containing your pairs of Kerberos principals and encrypted keys.

Data Type

string

Default Value

""

Remarks

The Keytab file containing your pairs of Kerberos principals and encrypted keys.

Kerberos Realm

The Kerberos Realm used to authenticate the user with.

Data Type

string

Default Value

""

Remarks

The Kerberos properties are used when using SPNEGO or Windows Authentication. The Kerberos Realm is used to authenticate the user with the Kerberos Key Distribution Service (KDC). The Kerberos Realm can be configured by an administrator to be any string, but conventionally it is based on the domain name. If Kerberos Realm is not specified the adapter will attempt to detect these properties automatically from the following locations:

Java System Properties: Kerberos settings can be configured in Java using a config file (krb5.conf) or using the system properties java.security.krb5.realm and java.security.krb5.kdc. The adapter will use the system settings if KerberosRealm and KerberosKDC

are not explicitly set.

Domain Name and Host: The adapter will infer the Kerberos Realm and Kerberos KDC from the user-configured domain name and host as a last resort. This might work in some Windows environments.

Note: Kerberos-based authentication is supported in JRE 1.6 and above only.

Kerberos Service KDC

The Kerberos KDC of the service.

Data Type

string

Default Value

""

Remarks

The KerberosServiceKDC is used to specify the service Kerberos KDC when using cross-realm Kerberos authentication.

In most cases, a single realm and KDC machine are used to perform the Kerberos authentication and this property is not required.

This property is available for complex setups where a different realm and KDC machine are used to obtain an authentication ticket (AS request) and a service ticket (TGS request).

Kerberos Service Realm

The Kerberos realm of the service.

Data Type

string

Default Value

""

Remarks

The KerberosServiceRealm is the specify the service Kerberos realm when using cross-realm Kerberos authentication.

In most cases, a single realm and KDC machine are used to perform the Kerberos authentication and this property is not required.

This property is available for complex setups where a different realm and KDC machine are used to obtain an authentication ticket (AS request) and a service ticket (TGS request).

Kerberos SPN

The Service Principal Name for the Kerberos Domain Controller.

Data Type

string

Default Value

""

Remarks

If the Service Principal Name on the Kerberos Domain Controller is not the same as the URL that you are authenticating to, set the Service Principal Name here.

Kerberos Ticket Cache

The full file path to an MIT Kerberos credential cache file.

Data Type

string

Default Value

""

Remarks

This property can be set if you wish to use a credential cache file that was created using the MIT Kerberos Ticket Manager or kinit command.

Location

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

Data Type

string

Default Value

""

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 Location property is only needed if you would like to customize definitions (e.g., change a column name, ignore a column, etc.) or extend the data model with new tables, views, or stored procedures.

The schema files are deployed alongside the adapter assemblies. You must also ensure that Location points to the folder that contains the schema files. The folder location can be a relative path from the location of the executable.

Log Modules

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.

Max Rows

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

string

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.

OAuth Access Token

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.

OAuth Client Id

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.

OAuth Client Secret

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.

OAuth Expires In

The lifetime in seconds of the OAuth AccessToken.

Data Type

string

Default Value

""

Remarks

Pair with OAuthTokenTimestamp to determine when the AccessToken will expire.

OAuth Grant Type

The grant type for the OAuth flow.

Data Type

string

Default Value

"CODE"

Remarks

The grant type for the OAuth flow. The following options are available: CODE,CLIENT,PASSWORD

OAuth Refresh Token

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.

OAuth Settings Location

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\\SharePoint 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\\SharePoint 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

OAuth Token Timestamp

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.

OAuth Verifier

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

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.

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.

Page Size

The number of results to return per page of data retrieved from SharePoint.

Data Type

string

Default Value

"1000"

Remarks

The number of results to return per page of data retrieved from SharePoint. Higher page sizes will result in fewer requests, but timeouts may occur.

Password

The password used to authenticate the user.

Data Type

string

Default Value

""

Remarks

The User, Password, and AuthScheme are together used to authenticate with the server.

Proxy Auth Scheme

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

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 Firewall Type.

Proxy Auto Detect

This indicates whether to use the system proxy settings or not. Set ProxyAutoDetect to FALSE to use custom proxy settings. This takes precedence over other proxy settings.

Data Type

bool

Default Value

true

Remarks

By default, the adapter uses the system HTTP proxy. Set this to false if you want to connect to another proxy.

To connect to an HTTP proxy, see Proxy Server.

For other proxies, such as SOCKS or tunneling, see Firewall Type.

Proxy Exceptions

A semicolon separated list of hosts or IPs that will be exempt from connecting through the ProxyServer .

Data Type

string

Default Value

""

Remarks

The ProxyServer will be used for all addresses, except for addresses defined in this property. Use semicolons to separate entries.

Note that the adapter will use the system proxy settings by default, without further configuration needed; if you want to explicitly configure proxy exceptions for this connection, you will need to set ProxyAutoDetect to false, and configure ProxyServer and ProxyPort. To authenticate, set ProxyAuthScheme and set ProxyUser and ProxyPassword, if needed.

Proxy Password

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 Firewall Type.

By default, the adapter uses the system proxy. If you want to connect to another proxy, set ProxyAutoDetect to false.

Proxy Port

The TCP port the ProxyServer proxy is running on.

Data Type

string

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 Firewall Type.

Proxy Server

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 Firewall Type.

By default, the adapter uses the system proxy. If you need to use another proxy, set ProxyAutoDetect to false.

Proxy SSL Type

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

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:

AUTO	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.

ALWAYS	The connection is always SSL enabled.
NEVER	The connection is not SSL enabled.
TUNNEL	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.

Proxy User

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 username of a user recognized by the HTTP proxy. If you are using Windows or Kerberos authentication, set this property to a username in one of the following formats:

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

Readonly

You can use this property to enforce read-only access to SharePoint 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.

Schema

The type of schema to use.

Data Type

string

Default Value

"SOAP"

Remarks

The schemas available are REST (to use Sharepoint REST API) and SOAP (to use Sharepoint SOAP API).

Share Point Edition

The edition of SharePoint being used. Set either SharePoint Online or SharePoint On-Premise.

Data Type

string

Default Value

"SharePoint OnPremise"

Remarks

The edition of SharePoint being used. Set either SharePoint Online or SharePoint On-Premise.

Show Hidden Columns

Boolean determining if hidden columns should be shown or not. If false, all hidden columns will be removed from the column listing.

Data Type

bool

Default Value

false

Remarks

Boolean determining if hidden columns should be shown or not. If false, all hidden columns will be removed from the column listing.

Show Predefined Columns

Boolean determining if predefined columns should be shown or not. If false, all columns derived from a base type will be removed from the column listing.

Data Type

bool

Default Value

true

Remarks

Boolean determining if predefined columns should be shown or not. If false, all columns derived from a base type will be removed from the column listing. These columns are normally system columns such as CreatedBy and Author. But, predefined columns may also include common columns such as Title.

Show Version Views

Indicate whether to display the view of list versions. Such as ListA_Versions.

Data Type

bool

Default Value

false

Remarks

Indicate whether to display the view of list versions. Such as ListA_Versions.

SSL Client Cert

The TLS/SSL client certificate store for SSL Client Authentication (2-way SSL).

Data Type

string

Default Value

""

Remarks

The name of the certificate store for the client certificate.

The SSLClientCertType field specifies the type of the certificate store specified by SSLClientCert. If the store is password protected, specify the password in SSLClientCertPassword.

SSLClientCert is used in conjunction with the SSLClientCertSubject field in order to specify client certificates. If SSLClientCert has a value, and SSLClientCertSubject is set, a search for a certificate is initiated. See SSLClientCertSubject for more information.

Designations of certificate stores are platform-dependent.

The following are designations of the most common User and Machine certificate stores in Windows:

MY	A certificate store holding personal certificates with their associated private keys.
CA	Certifying authority certificates.
ROOT	Root certificates.
SPC	Software publisher certificates.

In Java, the certificate store normally is a file containing certificates and optional private keys.

When the certificate store type is PFXFile, this property must be set to the name of the file. When the type is PFXBlob, the property must be set to the binary contents of a PFX file (for example, PKCS12 certificate store).

SSL Client Cert Password

The password for the TLS/SSL client certificate.

Data Type

string

Default Value

""

Remarks

If the certificate store is of a type that requires a password, this property is used to specify that password to open the certificate store.

SSL Client Cert Subject

The subject of the TLS/SSL client certificate.

Data Type

string

Default Value

"*"

Remarks

When loading a certificate the subject is used to locate the certificate in the store.

If an exact match is not found, the store is searched for subjects containing the value of the property. If a match is still not found, the property is set to an empty string, and no certificate is selected.

The special value "*" picks the first certificate in the certificate store.

The certificate subject is a comma separated list of distinguished name fields and values. For example, "CN=www.server.com, OU=test, C=US, E=support@company.com". The common fields and their meanings are shown below.

Field	Meaning
CN	Common Name. This is commonly a host name like www.server.com.
O	Organization
OU	Organizational Unit
L	Locality
S	State
C	Country
E	Email Address

If a field value contains a comma, it must be quoted.

SSL Client Cert Type

The type of key store containing the TLS/SSL client certificate.

Data Type

string

Default Value

""

Remarks

This property can take one of the following values:

USER - default	For Windows, this specifies that the certificate store is a certificate store owned by the current user. Note that this store type is not available in Java.

MACHINE	For Windows, this specifies that the certificate store is a machine store. Note that this store type is not available in Java.
PFXFILE	The certificate store is the name of a PFX (PKCS12) file containing certificates.
PFXBLOB	The certificate store is a string (base-64-encoded) representing a certificate store in PFX (PKCS12) format.
JKSFILE	The certificate store is the name of a Java key store (JKS) file containing certificates. Note that this store type is only available in Java.
JKSBLOB	The certificate store is a string (base-64-encoded) representing a certificate store in JKS format. Note that this store type is only available in Java.
PEMKEY_FILE	The certificate store is the name of a PEM-encoded file that contains a private key and an optional certificate.
PEMKEY_BLOB	The certificate store is a string (base64-encoded) that contains a private key and an optional certificate.
PUBLIC_KEY_FILE	The certificate store is the name of a file that contains a PEM- or DER-encoded public key certificate.
PUBLIC_KEY_BLOB	The certificate store is a string (base-64-encoded) that contains a PEM- or DER-encoded public key certificate.
SSHPUBLIC_KEY_FILE	The certificate store is the name of a file that contains an SSH-style public key.
SSHPUBLIC_KEY_BLOB	The certificate store is a string (base-64-encoded) that contains an SSH-style public key.
P7BFILE	The certificate store is the name of a PKCS7 file containing certificates.
PPKFILE	The certificate store is the name of a file that contains a PuTTY Private Key (PPK).
XMLFILE	The certificate store is the name of a file that contains a certificate in XML format.
XMLBLOB	The certificate store is a string that contains a certificate in XML format.

SSL Server Cert

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 will be rejected.

This property can take the forms:

Description	Example
A full PEM Certificate (example shortened for brevity)	-----BEGIN CERTIFICATE----- MIIChTCCAe4CAQAwDQYJKoZIhv. .....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)	ecadbdda5a1529c58a1e9e09828d70e4
The SHA1 Thumbprint (hex values can also be either space or colon separated)	34a929226ae0819f2ec14b4a3d904f801c bb150d

If not specified, any certificate trusted by the machine will be accepted. Use '*' to signify to accept all certificates (not recommended for security concerns).

SSO Domain

The domain of the user when using single sign-on (SSO).

Data Type

string

Default Value

""

Remarks

This property is only applicable when using single sign-on (UseSSO is set to true) and if the domain of the User (e.g. user@mydomain.com) is different than the domain configured within the SSO service (e.g. user@myssodomain.com).

This property may be required when using the AD FS, OneLogin, or OKTA SSO services.

SSO Login URL

The identity provider's login URL.

Data Type

string

Default Value

""

Remarks

The identity provider's login URL. This property can be found in the SharePoint account settings by navigating to **Administration Setup > Security Controls > Single Sign-On Settings** and then choosing the desired organization.

SSO Password

The password of the SSOProvider used to authenticate the user.

Data Type

string

Default Value

""

Remarks

The SSOUser and SSOPassword are together used to authenticate with the SSOProvider.

SSO Properties

Additional properties required to connect to the identity provider in a semicolon-separated list.

Data Type

string

Default Value

""

Remarks

Additional properties required to connect to the identity provider in a semicolon-separated list. SSOProperties is used in conjunction with the SSOLoginUrl.

SSO Provider

The name of the SSO provider you are trying to authenticate.

Data Type

string

Default Value

"OKTA"

Remarks

The name of the SSO provider you are trying to authenticate. The following options are available: OKTA,OneLogin,ADFS

SSO User

The SSOProvider user account used to authenticate.

Data Type

string

Default Value

""

Remarks

Together with SSOPassword, this field is used to authenticate against the specified SSOProvider.

STSURL

The URL of the security token service (STS) when using single sign-on (SSO).

Data Type

string

Default Value

""

Remarks

The URL of the security token service (STS) when using single sign-on (SSO). This rarely needs to be set explicitly.

Timeout

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

Data Type

string

Default Value

"60"

Remarks

If the Timeout property is set to 0, operations do not time out: They run until they complete successfully or encounter an error condition.

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

URL

The base URL for the site.

Data Type

string

Default Value

""

Remarks

The following are examples of valid URLs:

http://server/SharePoint/

http://server/Sites/mysite/

http://server:90/

The provider will use URL to derive URLs for other calls to the server.

Use Display Names

Boolean determining if the display names for the columns should be used instead of the API names.

Data Type

bool

Default Value

true

Remarks

Boolean determining if the display names for the columns should be used instead of the API names.

Use NTLMV1

Determines whether the driver will attempt to connect with NTLMv1 or NTLMv2 (default).

Data Type

bool

Default Value

false

Remarks

Determines whether the driver will attempt to connect with NTLMv1 or NTLMv2 (default).

User

The SharePoint user account used to authenticate.

Data Type

string

Default Value

""

Remarks

Together with Password, this field is used to authenticate against the SharePoint server.

For SharePoint On-Premise, User should include the domain and will look similar to the following: DOMAIN\Username.

For SharePoint Online, User will look similar to the following: username@domain.onmicrosoft.com.

Use Simple Names

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

Data Type

bool

Default Value

false

Remarks

Boolean determining if simple names should be used for tables and columns. SharePoint lists can have 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. If set to false, the tables and columns will appear as they do in SharePoint.

Use SSO

Whether or not to use single sign-on (SSO) to authenticate to SharePoint Online.

Data Type

bool

Default Value

false

Remarks

When set to true, single sign-on (SSO) will be used to authenticate to SharePoint Online using the account specified via User and Password. The Active Directory Federation Services (AD FS), OneLogin, and OKTA SSO identity providers are supported.

SSODomain may be required to be set if the domain configured on the SSO domain is different than the domain of the User.

SSO is only applicable when using SharePoint Online. SSO is not supported for On-Premise versions of SharePoint.

Logging

The adapter uses log4j to generate log files. The settings within the log4j configuration file will be used by the adapter to determine the type of messages to log. The following categories can be specified:

Error: Only error messages will be logged.

Info: Both Error and Info messages will be logged.

Debug: Error, Info, and Debug messages will be 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, i.e.

```
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 will never log 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 will default to the minimum value for that particular category. For example, if Verbosity is set to a value less than three and the Debug category is specified, the Verbosity will default to 3.

Here is a breakdown 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 SharePoint Adapter

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

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

The TDV Server will need to 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>\conf\server. Note reauthenticating to the TDV Studio will be required after restarting the server.

An example of the calls would be:

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

All logs for the adapter will be written to the "cs_cdata.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 accomplish this, navigate to Administrator -> Configuration to display the configuration window. Then expand Server -> Configuration -> Debugging and set the Debug Output Enabled option to True.

Using OAuth Authentication

OAuth requires the authenticating user to interact with SharePoint using the browser. The adapter facilitates this in various ways as described below.

Embedded Credentials

See Embedded Credentials to connect with the adapter's embedded credentials and skip creating a custom OAuth app.

Custom Credentials

Instead of connecting with the adapter's embedded credentials, you can register an app to obtain the OAuthClientId and OAuthClientSecret.

When to Create a Custom OAuth App

Creating a custom OAuth app is optional as the adapter is already registered with SharePoint and you can connect with its embedded credentials. You might want to create a custom OAuth app to change the information displayed when users log into the SharePoint OAuth endpoint to grant permissions to the adapter.

Creating a Custom OAuth App

See Creating a Custom OAuth App for a procedure.

Embedded Credentials

Authenticate using the Embedded OAuth Credentials

Desktop Authentication with the Embedded OAuth App

You can connect without setting any connection properties for your user credentials. Below are the minimum connection properties required to connect.

InitiateOAuth: Set this to GETANDREFRESH. You can use InitiateOAuth to avoid repeating the OAuth exchange and manually setting the OAuthAccessToken.

Url: Set this to the URL of your sharepoint site, such as https://mycompany.sharepoint.com.

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. Obtains a new access token when the old one expires.
3. Saves OAuth values in OAuthSettingsLocation to be persisted across connections.

Custom Credentials

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. Obtains a new access token when the old one expires.
3. Saves OAuth values in OAuthSettingsLocation to be persisted across connections.

Headless Machines

Using OAuth on a Headless Machine

To create SharePoint 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.

You can follow the headless OAuth authentication flow using the adapter's embedded OAuth credentials or using the OAuth credentials for your custom OAuth app.

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:

- **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.

Creating a Custom OAuth App

When to Create a Custom OAuth App

You might want to create a custom OAuth app to change the information displayed when users log into the SharePoint OAuth endpoint to grant permissions to the adapter.

Follow the steps below to create a custom OAuth app and obtain the connection properties in a specific OAuth authentication flow.

Steps to Create a Custom OAuth App

Follow the steps below to obtain the OAuth values for your app, the OAuthClientId and OAuthClientSecret.

1. Log in to https://portal.azure.com.
2. In the left-hand navigation pane, select Azure Active Directory then App Registrations and click the Add button.
3. Enter an app name and any URL you would like for the sign-on URL. The sign-on URL is not used by the adapter or in the authentication step, so it can be set to your home page or an arbitrary URL like http://localhost.
4. After creating the app, in the Properties section of your app settings, set the Multi-Tenanted option to Yes.
5. After you save the key, a value for the key is displayed once. Set OAuthClientSecret to the key value. Set OAuthClientId to the Application Id.
6. Select API Permissions and then click Add. If you plan for your app to connect without a user context, select the Application Permissions (OAuthGrantType = CLIENT). Otherwise, when selecting permissions, use the Delegated permissions.

7. In the API Permissions section, click on Add a permission and select Sharepoint. And choose the permissions you want your app to have. To view and edit lists you have to select at least the permission, AllSites.Manage.
8. Save your changes.
9. If you have selected to use permissions that require admin consent (such as the Application Permissions), you may grant them from the current tenant on the API Permissions page. Otherwise, follow the steps under Admin Consent

Admin Consent

Admin consent refers to when the Admin for an Azure Active Directory tenant grants permissions to an application which requires an admin to consent to the use case. The embedded app within the SharePoint Adapter, contains no permissions that require admin consent. Therefore, this information applies only to custom applications.

Admin Consent Permissions

When creating a new OAuth app in the Azure Portal, you must specify which permissions the app will require. Some permissions may be marked stating "Admin Consent Required". For example, all Groups permissions require Admin Consent. If your app requires admin consent, there are a couple of ways this can be done.

The easiest way to grant admin consent is to just have an admin log into portal.azure.com and navigate to the app you have created in App Registrations. Under API Permissions, there will be a button for Grant Consent. You can consent here for your app to have permissions on the tenant it was created under.

If your organization has multiple tenants or the app needs to be granted permissions for other tenants outside your organization, the GetAdminConsentURL may be used to generate the Admin Authorization url. Unlike the GetOAuthAuthorizationURL, there will be no important information returned from this endpoint. If the grants access, it will simply return a boolean indicating that permissions were granted.

Once an admin grants consent, authentication may be performed as normal.

Client Credentials

Client credentials refers to a flow in OAuth where there is no direct user authentication taking place. Instead, credentials are created for just the app itself. All tasks taken by the app are done without a default user context. This makes the authentication flow a bit different from standard.

Client OAuth Flow

All permissions related to the client oauth flow require admin consent. This means the app embedded with the SharePoint Adapter cannot be used in the client oauth flow. You must create your own OAuth app in order to use client credentials. See Creating a Custom OAuth App for more details.

In your App Registration in portal.azure.com, navigate to API Permissions and select the Microsoft Graph permissions. There are two distinct sets of permissions - Delegated and Application permissions. The permissions used during client credential authentication are under Application Permissions. Select the applicable permissions you require for your integration.

In addition to setting the standard OAuth client id and secret, to use client credentials you must also set the following connection properties:

- OAuthGrantType: Set this to CLIENT.
- AzureTenant: Set this to the tenant you wish to connect to.

Authentication with client credentials will take place automatically like any other connection, except there will be no window opened prompting the user. Because there is no user context, there is no need for a browser popup. Connections will take place and be handled internally.

Connecting to REST API

SharePoint REST API is supported both on Sharepoint OnPremise and on Sharepoint Online. To connect using the REST API set Schema to REST.

The property SharePointEdition may be used to define the edition of Sharepoint.

Sharepoint Online

SharePoint Online uses OAuth standard to authenticate. See the Using OAuth Authentication for more info.

Sharepoint OnPremise

See the Authenticating to SharePoint On Premises section in 'Establishing a Connection' for more info..

Using Kerberos

This section shows how to use the adapter to authenticate to SharePoint using Kerberos.

Authenticating with Kerberos

To authenticate to SharePoint using Kerberos, set the following properties:

- AuthScheme: Set this to NEGOTIATE
- KerberosKDC: Set this to the host name or IP Address of your Kerberos KDC machine.
- KerberosRealm: Set this to the realm of the SharePoint Kerberos principal. This will be the value after the '@' symbol (for instance, EXAMPLE.COM) of the principal value (for instance, MyService/MyHost@EXAMPLE.COM).
- KerberosSPN: Set this to the service and host of the SharePoint Kerberos Principal. This will be the value prior to the '@' symbol (for instance, MyService/MyHost) of the principal value (for instance, MyService/MyHost@EXAMPLE.COM).

Retrieve the Kerberos Ticket

You can use one of the following options to retrieve the required Kerberos ticket.

MIT Kerberos Credential Cache File

This option enables you to use the MIT Kerberos Ticket Manager or kinit command to get tickets. Note that you won't need to set the User or Password connection properties with this option.

1. Ensure that you have an environment variable created called KRB5CCNAME.
2. Set the KRB5CCNAME environment variable to a path pointing to your credential cache file (for instance, C:\krb_cache\krb5cc_0 or /tmp/krb5cc_0). This file will be created when generating your ticket with MIT Kerberos Ticket Manager.
3. To obtain a ticket, open the MIT Kerberos Ticket Manager application, click Get Ticket, enter your principal name and password, then click OK. If

successful, ticket information will appear in Kerberos Ticket Manager and will now be stored in the credential cache file.

4. Now that the credential cache file has been created, the adapter will use the cache file to obtain the kerberos ticket to connect to SharePoint.

As an alternative to setting the KRB5CCNAME environment variable, you can directly set the file path using the KerberosTicketCache property. When set, the adapter will use the specified cache file to obtain the kerberos ticket to connect to SharePoint.

Keytab File

If the KRB5CCNAME environment variable has not been set, you can retrieve a Kerberos ticket using a Keytab File. To do this, set the User property to the desired username and set the KerberosKeytabFile property to a file path pointing to the keytab file associated with the user.

User and Password

If both the KRB5CCNAME environment variable and the KerberosKeytabFile property have not been set, you can retrieve a ticket using a User and Password combination. To do this, set the User and Password properties to the user/password combo that you use to authenticate with SharePoint.

Cross-Realm Authentication

More complex Kerberos environments may require cross-realm authentication where multiple realms and KDC servers are used (e.g. where one realm/KDC is used for user authentication and another realm/KDC used for obtaining the service ticket).

In such an environment, the KerberosRealm and KerberosKDC properties can be set to the values required for user authentication. The KerberosServiceRealm and KerberosServiceKDC properties can be set to the values required to obtain the service ticket.

Advanced Settings

Fine Tuning the SharePoint Connection

To make it easier to access data in advanced integrations, use the following connection properties to control column name identifiers and other aspects of data access:

UseDisplayNames: Set this to true to return column names that match field names in the underlying API. By default, the adapter uses column names that match the field names defined in SharePoint.

UseSimpleNames: Set this to true to perform substitutions on special characters in column names that SharePoint allows but that many databases typically do not.

ShowPredefinedColumns: Set this to false to exclude fields derived from fields in the list; for example, Author and CreatedAt. This setting excludes the predefined fields from being returned in *SELECT **statements and schema discovery.

ShowHiddenColumns: When true, columns marked as hidden in SharePoint will be displayed by the adapter.

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 SSL Server Cert property for the available formats to do so.

Connecting Through a Firewall or Proxy

To connect through the Windows system proxy, set only the SharePoint authentication properties and the URL. To connect to other proxies, set ProxyAutoDetect to false and in addition set the following.

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

To connect to other proxies, set FirewallType, FirewallServer, and FirewallPort. To tunnel the connection, set FirewallType to TUNNEL. To authenticate to a SOCKS proxy, set FirewallType to SOCKS5. Additionally, specify FirewallUser and FirewallPassword.

Troubleshooting the Connection

To show adapter activity from query execution to HTTP calls, use Logfile and Verbosity. The examples of common connection errors below show how to use these properties to get more context. Contact the support team for help tracing the source of an error or circumventing a performance issue.

Authentication errors: Typically, recording a Logfile at Verbosity 4 is necessary to get full details on an authentication error.

Queries time out: A server that takes too long to respond will exceed the adapter's client-side timeout. Often, setting the Timeout property to a higher value will avoid a connection error. Another option is to disable the timeout by setting the property to 0. Setting Verbosity to 2 will show where the time is being spent.

The certificate presented by the server cannot be validated: This error indicates that the adapter cannot validate the server's certificate through the chain of trust. (If you are using a self-signed certificate, there is only one certificate in the chain).

To resolve this error, you must verify yourself that the certificate can be trusted and specify to the adapter that you trust the certificate. One way you can specify that you trust a certificate is to add the certificate to the trusted system store; another is to set SSLServerCert.

SQL Compliance

The SharePoint 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 SharePoint 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-Za-z0-9_:@].

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 SharePoint 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 ] } [ , ... ]
  ]
  [
    LIMIT <expression>
  ]
} | SCOPE_IDENTITY()

  <expression> ::=
    | <column_reference>
    | @ <parameter>
    | ?
    | COUNT( * | {  <expression> } )
    | { AVG | MAX | MIN | SUM | COUNT } ( <expression> )
    | <literal>
    | <sql_function>

  <search_condition> ::=
    {
      <expression> { = | != | <> | > | < | >= | <= | BEGINSWITH |
CONTAINS | IN | IS NULL | IS NOT NULL | AND | OR } [ <expression> ]
    } [ { AND | OR } ... ]
```

Examples

Return all columns:

```
SELECT * FROM Calendar
```

Rename a column:

```
SELECT "Location" AS MY_Location FROM Calendar
```

Search data:

```
SELECT * FROM Calendar WHERE Location <> 'Chapel Hill';
```

The SharePoint APIs support the following operators in the WHERE clause: =, !=, <>, >, <, >=, <=, BEGINSWITH, CONTAINS, IN, IS NULL, IS NOT NULL, AND, OR.

```
SELECT * FROM Calendar WHERE Location <> 'Chapel Hill';
```

Sort a result set in ascending order:

```
SELECT Id, Location FROM Calendar  ORDER BY Location 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):

```
SELECT Id, Location INTO "csv://Calendar.txt" FROM "Calendar" WHERE
Location = 'Chapel Hill'
```

You can specify other formats in the file URI. The possible delimiters are tab, semicolon, and comma with the default being comma. The following example exports tab-separated values:

```
SELECT Id, Location INTO "csv://Calendar.txt;delimiter=tab" FROM
"Calendar" WHERE Location = 'Chapel Hill'
```

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:

```
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 Calendar (Location) VALUES (?)";
PreparedStatement pstmt =
connection.prepareStatement(cmd,Statement.RETURN_GENERATED_KEYS);
pstmt.setString(1, "U.S.A.");
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.

```
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.

```
String cmd = "UPDATE Calendar SET Location='U.S.A.' 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 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.

```
<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.

```
Connection connection =
DriverManager.getConnection("jdbc:sharepoint:User=MyUserAccount;Pa
ssword=MyPassword;Auth
Scheme=NTLM;URL=http://sharepointserver/mysite;",);
String cmd = "DELETE FROM Calendar 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;

Data Model

SOAP Data Model

The SharePoint Adapter models SharePoint entities in relational Tables, Views, and Stored Procedures. The table definitions are dynamically obtained based on your SharePoint site. Any changes you make, such as adding a custom field or changing a field's data type, are automatically reflected when you connect.

Customizing the Data Model

The adapter sets defaults to facilitate the maximum number of integrations; however, the following connection properties allow a greater granularity of customization useful in advanced integrations:

- CalculatedDataType: The data type to be used for calculated fields.
- CreateIDColumns: Indicates whether or not to create supplemental ID columns for SharePoint columns that use values from information stored in other Lists.
- FolderOption: An option to determine how to display folders in results. Enter either FilesOnly, FilesAndFolders, Recursive, or RecursiveAll.
- PseudoColumns: Indicates whether or not to report pseudo columns as columns in the table metadata.

Tables

Tables describes the available tables.

The adapter can expose custom lists from SharePoint that are not mentioned in the Tables. The data model illustrates a sample of what your SharePoint site might look like. The actual data model will be obtained dynamically based on your user credentials and SharePoint site.

Views

Typically, entities that cannot be modified are represented as Views, or read-only tables. You can also access custom views of a list as relational views.

To get data from a custom view of a list, you can set the ViewID pseudo column in the WHERE clause.

```
SELECT * FROM ListName WHERE ViewID='ID of the view'
```

You can get the ID of the view from the Views list. You must specify the List pseudo column to get a list of views for that list. For instance:

```
SELECT * FROM Views WHERE List ='ListName'
```

Stored Procedures

Stored Procedures are function-like interfaces to the data source. They surface additional capabilities of the SharePoint API such as searching, updating, and modifying information.

Tables

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

Generally, querying SharePoint 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.

SharePoint Adapter Tables

Name	Description
Announcements	Create, update, delete, and query items in Announcement lists.
Attachments	Read or delete Attachments for the specified item on the specified list.
Calendar	Create, update, query, and delete items in SharePoint Calendar lists.
Contacts	Create, update, query, and delete items in SharePoint Contact lists.
Documents	Create, update, delete, and query Documents from SharePoint libraries.
Groups	Create, update, delete, and query Groups from SharePoint.
IssueTracking	Create, update, query, and delete items in SharePoint Issue Tracking lists.
Links	Create, update, query, and delete items in SharePoint Link lists.
Pictures	Create, update, delete, and query documents in a picture library.
Roles	Create, update, delete, and query Roles from SharePoint.
Tasks	Create, update, query, and delete items in a SharePoint Tasks list.
Users	Create, update, delete, and query Users from SharePoint.
	Create, update, delete, and query the available lists in SharePoint.

Announcements

Create, update, delete, and query items in Announcement lists.

Columns

Name	Type	ReadOnly	Description
ID [KEY]	*String*	True	The ID of the item. The format of the ID is: List | ItemID.
List	*String*	False	The name of the list.
Title	*String*	False	The title of the item.

Body	*String*	False	The body of the item.
Expires	*Date*	False	The date the item expires on.
Author	*String*	False	The user that created the item.
Editor	*String*	False	The last user that modified the item.
Modified	*Date*	False	The date modified.
Created	*Date*	False	The date created.

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
CAMLQuery	*String*	The query to be used while listing the list items.

Attachments

Read or delete Attachments for the specified item on the specified list.

Table Specific Information

Select

List and ItemId are required to return Attachments.

Insert

Call the AddAttachments stored procedure to add new attachments to a list item.

Columns

Name	Type	ReadOnly	Description
Url [KEY]	*String*	True	Description of the term set.

List	*String*	True	The list to retrieve attachments from.
ItemID	*String*	True	The ID of the item on the list to retrieve attachments from.
Name	*String*	True	The name of the attachment on the item.

Calendar

Create, update, query, and delete items in SharePoint Calendar lists.

Columns

Name	Type	ReadOnly	Description
ID [KEY]	*String*	True	The ID of the calendar entry. The format of the ID is: List | ItemID.
List	*String*	False	The name of the calendar list.
Title	*String*	False	The title of the entry.
ParticipantsPicker	*String*	False	The attendees of the calendar entry.
FreeBusy	*String*	False	A flag indicating the free or busy status during the event.
Facilities	*String*	False	The resources for this calendar entry.
Location	*String*	False	The location of the event.
EventDate	*Date*	False	The date the calendar entry starts.
EndDate	*Date*	False	The date the calendar entry ends.
Description	*String*	False	The description of the event.
Duration	*String*	False	The duration of the calendar entry.
EventType	*String*	False	The type of the event.
IsAllDayEvent	*String*	False	A flag indicating whether the entry is an all day event or not.

fRecurrence	*String*	False	A flag indicating whether this is a recurring event.
Author	*String*	False	The user that created the item.
Editor	*String*	False	The last user that modified the item.
Modified	*Date*	False	The date modified.
Created	*Date*	False	The date created.

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
CAMLQuery	*String*	The query to be used while listing the list items.

Contacts

Create, update, query, and delete items in SharePoint Contact lists.

Columns

Name	Type	ReadOnly	Description
ID [KEY]	*String*	True	The ID of the item. The value of this field is of the format: List|ItemId
List	*String*	False	The name of the contacts list.
FirstName	*String*	False	The first name of the contact.
Title	*String*	False	The last name of the contact.
FullName	*String*	False	The full name of the contact.
Email	*String*	False	The email of the entry.

Company	*String*	False	The company the contact works for.
JobTitle	*String*	False	The job title of the contact.
WorkPhone	*String*	False	The business phone of the contact.
HomePhone	*String*	False	The home phone of the contact.
CellPhone	*String*	False	The mobile phone of the contact.
WorkFax	*String*	False	The fax number of the contact.
WorkAddress	*String*	False	The address of the contact.
WorkCity	*String*	False	City for the address of the contact.
WorkState	*String*	False	State for the address of the contact.
WorkZip	*String*	False	Postal code for the address of the contact.
WorkCountry	*String*	False	Country for the address of the contact.
WebPage	*String*	False	Web page for the contact.
Comments	*String*	False	Notes about the contact.
Author	*String*	False	The user that created the item.
Editor	*String*	False	The last user that modified the item.
Modified	*Date*	False	The date modified.
Created	*Date*	False	The date created.

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

CAMLQuery	*String*	The query to be used while listing the list items.

Documents

Create, update, delete, and query Documents from SharePoint libraries.

Table Specific Information

Select

To pull items in a folder, you need to use the either the Path or FileDirRef field (depending on whether UseDisplayNames is True or False respectively) in the WHERE clause. For example:

```
SELECT * FROM MyDocumentLibrary WHERE
SubFolder='MyDocumentLibrary/MyFolder'
```

Insert

The Title column is required to insert to this table.

Columns

Name	Type	ReadOnly	Description
ID [KEY]	*String*	True	The ID of the document. The format of the ID is: List|ItemID.
List	*String*	False	The name of the document list.
LinkFileName	*String*	False	The title of the document.
CheckOutUser	*String*	False	The user that the document is checked out to.
Author	*String*	False	The user that created the item.
Editor	*String*	False	The last user that modified the item.
Modified	*Date*	False	The date modified.
Created	*Date*	False	The date created.

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
Subfolder	*String*	A subfolder to specify if you would like to list documents in subfolders as well. Use * to denote all subfolders. The default value is *.
CAMLQuery	*String*	The query to be used while listing the list items.

Groups

Create, update, delete, and query Groups from SharePoint.

Table Specific Information

Insert

The Name, DefaultLogin, and OwnerLogin columns are required to insert to this table.

To use the UserName pseudo column, you must set the value to the LoginName of the user. You can obtain the LoginName by querying the Users table.

Columns

Name	Type	ReadOnly	Description
Name# [KEY]	*String*	False	The name of the group.
Description#	*String*	False	A description of the group.
OwnerLogin #	*String*	False	The user name of the owner of the group. This value should be in the format DOMAIN\\username.
OwnerType#	*String*	False	The type of owner. User or group.

DefaultLogin #	*String*	False	The user name of the default user for the group. This value should be in the format DOMAIN\\username.

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
UserName	*String*	The logical name of the user to return groups for. Specify this value on the SELECT statement to return only groups the specified User is in.

IssueTracking

Create, update, query, and delete items in SharePoint Issue Tracking lists.

Columns

Name	Type	ReadOnly	Description
ID [KEY]	*String*	True	The ID of the list entry. The format of the ID is: List | ItemID.
List	*String*	False	The name of the issue tracking list.
Title	*String*	False	The title of the entry.
AssignedTo	*String*	False	The user the issue is assigned to.
Status	*String*	False	The status of the issue.
Priority	*String*	False	The priority assigned to this issue.
Category	*String*	False	The category that the issue belongs to.
DueDate	*Date*	False	The due date of the issue.

RelatedIssues	*String*	False	A list of related issues.
Comments	*String*	False	Comments associated with this issue.
Author	*String*	False	The user that created the item.
Editor	*String*	False	The last user that modified the item.
Modified	*Date*	False	The date modified.
Created	*Date*	False	The date created.

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
CAMLQuery	*String*	The query to be used while listing the list items.

Links

Create, update, query, and delete items in SharePoint Link lists.

Columns

Name	Type	ReadOnly	Description
ID [KEY]	*String*	True	The ID of the item. The format of the ID is: List|ItemID.
List	*String*	False	The name of the link list.
URL	*String*	False	The URL of the link entry.
Comments	*String*	False	The notes associated with the entry.
Author	*String*	False	The user that created the item.

Editor	*String*	False	The last user that modified the item.
Modified	*Date*	False	The date modified.
Created	*Date*	False	The date created.

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
CAMLQuery	*String*	The query to be used while listing the list items.

Pictures

Create, update, delete, and query documents in a picture library.

Columns

Name	Type	ReadOnly	Description
ID [KEY]	*String*	True	The ID of the document. The format of the ID is: List | ItemID.
List	*String*	False	The name of the list the document belongs to.
LinkFileName	*String*	False	The filename of the document on the server.
ImageCreateDate	*Date*	False	The date the picture was taken.
Description	*String*	False	The description associated with the document.
Keywords	*String*	False	The keywords associated with the document.
CheckedOutUser	*String*	False	The user the document is checked out to.

Author	*String*	False	The user that created the item.
Editor	*String*	False	The last user that modified the item.
Modified	*Date*	False	The date that the document was modified.
Created	*Date*	False	The date that the document was created.

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
CAMLQuery	*String*	The query to be used while listing the list items.

Roles

Create, update, delete, and query Roles from SharePoint.

Table Specific Information

Select

To use the UserName pseudo column, you must set the value to the LoginName of the user. You can obtain the LoginName by querying the Users table.

Insert

To insert a Role, at least the Name is required:

```
INSERT INTO Roles (Name) VALUES ('My Role')
```

Columns

Name	Type	ReadOnly	Description
Name# [KEY]	*String*	False	The name of the role.

Description#	*String*	False	A description of the role.
Permissions#	*Long*	False	A long representing the permissions for the role.
RoleType	*String*	True	The type of role.
IsHidden	*Boolean*	True	A boolean indicating if the role is hidden.

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
UserName	*String*	The login name of the user to return roles for. Specify this value on a SELECT statement to return only roles assigned to the specified user.
GroupName	*String*	The name of the group to return roles for. Specify this value on a SELECT statement to return only roles assigned to the specified group.

Tasks

Create, update, query, and delete items in a SharePoint Tasks list.

Columns

Name	Type	ReadOnly	Description
ID [KEY]	*String*	True	The ID of the item. The format of the ID is: List|ItemID.
List	*String*	False	The name of the issue tracking list.
Title	*String*	False	The title of the item.
Priority	*String*	False	The priority of the task.

Status	*String*	False	The status of the task.
PercentComplete	*String*	False	The completed progress of the task, expressed as a percentage.
AssignedTo	*String*	False	The SharePoint user the task is assigned to.
TaskGroup	*String*	False	The SharePoint group the task is assigned to.
Body	*String*	False	The description of the item.
StartDate	*Date*	False	The start date of the task.
DueDate	*Date*	False	The due date of the task.
WorkflowName	*String*	False	The name of the workflow associated with the task.
Author	*String*	False	The user that created the item.
Editor	*String*	False	The last user that modified the item.
Modified	*Date*	False	The date modified.
Created	*Date*	False	The date created.

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
CAMLQuery	*String*	The query to be used while listing the list items.

Users

Create, update, delete, and query Users from SharePoint.

Table Specific Information

Select

Retrieve all users created for the SharePoint Account:

```
SELECT * FROM Users
```

You can retrieve Users that belong to a specific Group. In this case specify the Group Name

```
Select * FROM Users WHERE [Group]="GroupName"
```

Or you can retrieve Users that have a specific Role. In this case specify the Role Name

```
Select * FROM Users WHERE [Role]="RoleName"
```

Insert

In order to add Users to SharePoint, the Group or Role the User is being added to must be specified. Additionally, specify the LoginName:

```
INSERT INTO Users (LoginName, Role) VALUES ('MYDOMAIN\MyUser', 'My Role')
```

Columns

Name	Type	ReadOnly	Description
LoginName# [KEY]	*String*	False	The login name of the user.
Name#	*String*	False	The name of the user.
Email#	*String*	False	The email address of the user.
IsInDomain Group	*Boolean*	True	A boolean indicating if the user is in the domain group.
IsSiteAdmin	*Boolean*	True	A boolean indicating if the user is a site admin.
Notes#	*String*	False	Optional notes concerning the user.
SecurityId	*String*	True	The security Id (SID) for the user.

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
Group	*String*	The group you are adding a user to or deleting the user from. This is an input-only value and either Group or Role must be specified for inserts, but may be optionally specified for deletions.
Role	*String*	The role you are adding a user to or deleting the user from. This is an input-only value and either Group or Role must be specified for inserts, but may be optionally specified for deletions.
NewName	*String*	The new name of the user. Specify this value when updating the user to change the name of the user.

Views

Create, update, delete, and query the available lists in SharePoint.

Table Specific Information

Views is a special table. It may be used to get, update, insert, and delete views from a specified List.

Select

In order to return results from Views, either the Id or List must be specified in the SELECT statement. For example:

```
SELECT * FROM Views WHERE List='MyListName'
```

Insert

The List, Name, Type, and Fields columns are required to insert to this table.

Columns

Name	Type	ReadOnly	Description
ID [KEY]	*String*	True	The Id of the view.
List	*String*	True	The list the view is associated with. A list must be specified when performing SELECT statements if the Id is not specified.
ViewID	*String*	True	The Id of the view. May only be unique for the specific list.
Name	*String*	False	The name of the view.
Type	*String*	False	The type of view. This must have a value on inserts and updates. The allowed values are *CALENDAR, GRID, HTML*. The default value is *HTML*.
Fields	*String*	False	A comma separated list of the fields associated with the view. This is space-sensitive.
IsDefault	*Boolean*	False	A boolean indicating if the view is the default view for the list.
Query	*String*	False	A query for the view.

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. To find out more about tables and stored procedures, please navigate to their corresponding entries in this help document.

SharePoint Adapter Views

Name	Description
FileVersions	Lists the versions of files available on SharePoint.
GetValidTerms	Gets a list of valid terms for the specified column on the specified table.
Lists	Lists the available lists in SharePoint.
Permissions	The permissions for a site or list.
Subsites	This lists the available subsites.

FileVersions

Lists the versions of files available on SharePoint.

View-Specific Information

Library and File must be specified to return results from this view.

Columns

Name	Type	Description
ID [KEY]	*String*	The ID of the version.
Comments	*String*	Comments about the particular version.
CreateBy	*String*	The username of the SharePoint user who modified this version of the file.
Date	*Datetime*	When the file was modified.
Size	*String*	The size of this version of the file.
Url	*String*	The URL to this version of the file.
Library	*String*	The library name on SharePoint you are listing versions from. A library must be specified to retrieve the versions for a file. The default value is Shared Documents.

File	*String*	The name of the file on SharePoint to list versions for. A file must be specified to retrieve the versions for a file.

GetValidTerms

Gets a list of valid terms for the specified column on the specified table.

Table Specific Information

GetValidTerms is a special view. It may be used to get valid terms for a Taxonomy or Managed Metadata column of a given list. To use the view, supply both the name of the table and the column for which you are looking to get valid terms. For example:

```
SELECT * FROM GetValidTerms WHERE List='MyListName' AND
ColumnName='MyManagedMetadataColumn'
```

Columns

Name	Type	Description
ID [KEY]	*String*	The identifier of the term.
TermLabelValue	*String*	The label of the term.
Description	*String*	Description of the term set.
NameInRequestedLang	*String*	The name of the term set in the language requested by the client.
IsOpen	*Boolean*	Boolean indicating if the term set is open.
Deprecated	*Boolean*	Boolean indicating if the term is deprecated.
InternalId	*String*	Internal identifier for the term.
TermSetContact	*String*	Term set contact.
ContainerDesc	*String*	Container node for the description.
SingleTermLabelDesc	*String*	This fully describes a single term label.

IsDefaultLabel	*Boolean*	True if the term label is the default term label.
BelongsTo	*String*	This item describes a term set to which a term belongs.
IsTaggingAvailable	*Boolean*	If the term set is available for tagging, this value is true.
TermPath	*String*	Term path of the term with term labels.
TermpathoftermwithIds	*String*	Term path of term with identifiers.
ChildTerms	*String*	A string value that indicates a custom sort order for the child terms of the term identified by PertainingToTerm.
HasChildTerms	*Boolean*	True if the term has child terms.
PertainingToTerm	*Boolean*	Identifier of the term that this term set information is pertaining to.

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
List	*String*	The name of the list to get valid terms for.
ColumnName	*String*	The name of the column to get valid terms for.
LocaleId	*String*	The locale Id for the term. Defaults to 1033.

Lists

Lists the available lists in SharePoint.

Table Specific Information

Lists can be used to list the tables in SharePoint. This will only return actual lists in SharePoint and not any special tables associated with the adapter.

The following columns can be used in the WHERE clause: Title and BaseTemplate.

Columns

Name	Type	Description
ID [KEY]	*String*	The Id of the list.
Title	*String*	The title of the list. This column may be used in the WHERE clause and may be used with a wild card (*) character.
Description	*String*	A description for the list.
BaseTemplate	*String*	Indicates the type of template used to create the list. This column may be used in the WHERE clause.
Version	*Double*	The version of the list.
Url	*String*	The default URL of the list.
EmailAlias	*String*	The email alias of the list.
ImageUrl	*String*	The image URL of the list.
ItemCount	*Integer*	The number of items in the list.
Item_Deleted	*Datetime*	The last time an item was deleted from this list.
Item_Modified	*Datetime*	The last time an item was modified from this list.
SendToUrl	*String*	The send-to URL of the list.
Created	*Datetime*	The time when the list was created.
AllowDeletion	*String*	Whether items can be deleted.
AllowMultiResponses	*Boolean*	Boolean indicating if multiple responses are enabled for the survey.

Direction	*String*	A string that contains LTR if the reading order is left-to-right, RTL if it is right-to-left, or None.
EnableAssignedToEmail	*Boolean*	Boolean indicating if assigned-to emails are enabled. Only applies to issues lists.
EnableAttachments	*Boolean*	Boolean indicating if attachments may be added to items in the list. Does not apply to document libraries.
EnableModeration	*Boolean*	Boolean indicating if content approval is enabled for the list.
EnableVersioning	*Boolean*	Boolean indicating if versioning is enabled for the list.
Hidden	*Boolean*	Boolean indicating if the list is hidden so that it does not appear on the Documents and Lists page, Quick Launch bar, Modify Site Content page, or Add Column page as an option for lookup fields.
MultipleDataList	*Boolean*	Boolean indicating if a meeting-workspace site contains data for multiple meeting instances within the site.
Ordered	*Boolean*	Boolean indicating if items in the list can be sorted on the Edit View page.
Showuser	*Boolean*	Boolean indicating if the names of users are shown in the results of the survey.

Permissions

The permissions for a site or list.

Columns

Name	Type	Description
MemberID [KEY]	*String*	The ID of the permission.

Mask	*Long*	A 32-bit integer in 0x00000000 format that represents a Microsoft.SharePoint.SPRights value and defines the permission. Use the pipe symbol ('|') in C# or Or in Microsoft Visual Basic to delimit values when creating a custom permission mask that combines permissions.
MemberIsUser	*Bool*	Indicate whether it is the permission for user.
MemberGlobal	*Bool*	Indicate whether it is the permission for group.
RoleName	*String*	A string that contains the name of the site group, the name of the cross-site group, or the user name (DOMAIN\User_Alias) of the user to whom the permission applies.

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	*String*	A string that contains the name of the list or site.
ObjectType	*String*	A string that specifies either List or Web.
ItemID	*String*	ID of the item.

Subsites

This lists the available subsites.

Columns

Name	Type	Description
Title	*String*	The name of the subsite.
Url	*String*	The url of the subsite.

Stored Procedures

Stored procedures are available to complement the data available from the SOAP Data Model. It may be necessary to update data available from a view using a stored procedure because the data does not provide for direct, table-like, two-way updates. In these situations, the retrieval of the data is done using the appropriate view or table, while the update is done by calling a stored procedure. Stored procedures take a list of parameters and return back a dataset that contains the collection of tuples that constitute the response.

SharePoint Adapter Stored Procedures

Name	Description
AddAttachment	Add an Attachment to a SharePoint List item.
AddList	Creates a list on a SharePoint site.
AddListColumn	Adds a new column to the specified list.
AddUserToGroup	Add the user to specified group.
AddUserToRole	Add the user to specified role.
CheckInDocument	Checks in a document to SharePoint and releases the lock on the document.
CheckOutDocument	Checks out a document from SharePoint.
CopyDocument	Copies a document from the SharePoint library.
CreateFolder	Adds a folder to a document library on a SharePoint site.
CreateSchema	Creates the schema file for the specified SharePoint list. The schema file may be customized manually to exclude unwanted columns or include additional information about columns.
DeleteAttachment	Delete an attachment from a SharePoint list item.
DeleteDocument	Delete a document on the SharePoint library.
DeleteList	Permanently deletes a list from a SharePoint site.
DeleteListColumn	Deletes a column from the specified list.
DeleteUserFromGroup	Delete the user from specified group.
DeleteUserFromRole	Delete the user from specified role.
DiscardCheckOutDocument	Discards a check out on a document in SharePoint. This does not check a new file into SharePoint. It only releases the lock on the document.
DownloadAttachment	Download a document from the SharePoint list.
DownloadDocument	Download a document from the SharePoint library.
UpdateList	Updates a list on a SharePoint site.

UpdateListColumn	Updates a column to the specified list.
UploadDocument	Upload a document to the SharePoint library.

AddAttachment

Add an Attachment to a SharePoint List item.

Input

Name	Type	Description
File	*String*	The path of the local file to be added.
List	*String*	The name of the List on the SharePoint server.
ItemID	*String*	The ID of the item on the List to add attachments for.

Result Set Columns

Name	Type	Description
URL	*String*	The URL of the newly created item.
Result	*String*	Boolean value indicating whether the stored procedure was successful.

AddList

Creates a list on a SharePoint site.

Input

Name	Type	Description
Name	*String*	The name of the list on the SharePoint server.

Template	*String*	The name of the template to use for the list creation. The allowed values are *GenericList, DocumentLibrary, Survey, Links, Announcements, Contacts, Events, Tasks, DiscussionBoard, PictureLibrary, DataSources, WebTemplateCatalog, UserInformation, WebPartCatalog, ListTemplateCatalog, XMLForm, MasterPageCatalog, NoCodeWorkflows, WorkflowProcess, WebPageLibrary, CustomGrid, DataConnectionLibrary, WorkflowHistory, GanttTasks, Meetings, Agenda, MeetingUser, Decision, MeetingObjective, TextBox, ThingsToBring, HomePageLibrary, Posts, Comments, Categories, IssueTracking, AdminTasks*. The default value is *GenericList*.
Description	*String*	The description of the list to add.

Result Set Columns

Name	Type	Description
Result	*String*	Boolean value indicating whether the operation was successful.

AddListColumn

Adds a new column to the specified list.

Input

Name	Type	Description
List	*String*	The name of the list on the SharePoint server.
ColumnName	*String*	The name of the column to add.
DisplayName	*String*	The display name of the column to add.
DefaultValue	*String*	The default value of the column to add.

ColumnType	*String*	The data type of the column to add.
MaxLength	*String*	The values' maximum length of the column to add.
PrimaryKey	*String*	Boolean value indicating whether or not the column should be primary key.
ReadOnly	*String*	Boolean value indicating whether or not the column is read only.
Required	*String*	Boolean value indicating whether or not the column is required.

Result Set Columns

Name	Type	Description
Result	*String*	Boolean value indicating whether the operation was successful.

AddUserToGroup

Add the user to specified group.

Input

Name	Type	Description
LoginName	*String*	The login name of the user.
Group	*String*	The group you are adding a user to and selecting or deleting the user from. This is an input-only value and either Group or Role must be specified for inserts and selects, but may be optionally specified for deletions.

Result Set Columns

Name	Type	Description
Result	*Boolean*	Boolean value indicating whether the stored procedure was successful.

AddUserToRole

Add the user to specified role.

Input

Name	Type	Description
LoginName	*String*	The login name of the user.
Role	*String*	The role you are adding a user to and selecting or deleting the user from. This is an input-only value and either Group or Role must be specified for inserts and selects, but may be optionally specified for deletions.

Result Set Columns

Name	Type	Description
Result	*Boolean*	Boolean value indicating whether the stored procedure was successful.

CheckInDocument

Checks in a document to SharePoint and releases the lock on the document.

Input

Name	Type	Description
File	*String*	The path of the file you are using to overwrite the document on SharePoint with. For example: C:\myfolder\myfile.txt.
Library	*String*	The name of the library on the SharePoint server. For example: Shared Documents.
Comment	*String*	A comment to leave when checking the file in.
RemoteFile	*String*	The path of the file on the server. This can be the full URL or simply the file name.

Result Set Columns

Name	Type	Description
Result	*String*	Boolean value indicating whether the operation was successful.

CheckOutDocument

Checks out a document from SharePoint.

Input

Name	Type	Description
Library	*String*	The name of the library on the SharePoint server.
RemoteFile	*String*	The path of the file on the server. This can be the full URL or simply the file name.

Result Set Columns

Name	Type	Description
Result	*String*	Boolean value indicating whether the operation was successful.

CopyDocument

Copies a document from the SharePoint library.

Input

Name	Type	Description
DocumentName	*String*	The name of the document in the document library to be copied.
DocumentLibrary	*String*	The name of the document library the document is currently stored on.

NewDocumentLibrary	*String*	The name of the document library the document is being copied to.
NewDocumentName	*String*	The new name of the document once it has been copied. If left blank, this will be the same as the DocumentName.
MetadataName#	*String*	The name of a metadata field to be set for the document.
ReturnID	*String*	The return ID of the document.

Result Set Columns

Name	Type	Description
Result	*String*	Boolean value indicating whether the operation was successful.
_dlc_DocId	*String*	The document ID.
_dlc_DocIdUrl	*String*	The URL of the document ID.
Vti_author	*String*	The creator of the document.
Vti_etag	*String*	The e-tag of the document.
ID	*String*	The ID of the document.
FileRef	*String*	The file reference of the document.

CreateFolder

Adds a folder to a document library on a SharePoint site.

Input

Name	Type	Description
Library	*String*	The name of the library on the SharePoint server.
Name	*String*	Name of the folder to which the document is to be added.

Result Set Columns

Name	Type	Description
Result	*String*	Boolean value indicating whether the stored procedure was successful.

CreateSchema

Creates the schema file for the specified SharePoint list. The schema file may be customized manually to exclude unwanted columns or include additional information about columns.

Input

Name	Type	Description
SchemaDirectory	*String*	The directory where the schema file will be saved to. To begin querying these new files, set the Location connection property to the folder containing these new schema files.
Table	*String*	The name of the table for which to create a schema.
TableDescription	*String*	An optional description of the table.

Result Set Columns

Name	Type	Description
Result	*String*	Returns Success or Failure.
SchemaFile	*String*	The generated schema file.
Columns	*String*	The number of columns found.

DeleteAttachment

Delete an attachment from a SharePoint list item.

Input

Name	Type	Description
URL	*String*	Full URL to attachment to be deleted.

Result Set Columns

Name	Type	Description
Result	*String*	Boolean value indicating whether the stored procedure was successful.

DeleteDocument

Delete a document on the SharePoint library.

Input

Name	Type	Description
Library	*String*	The name of the library on the SharePoint server.
Path	*String*	The path of the file (or folder) to remove from the document library.

Result Set Columns

Name	Type	Description
Result	*String*	Boolean value indicating whether the stored procedure was successful.

DeleteList

Permanently deletes a list from a SharePoint site.

Input

Name	Type	Description

List	*String*	The name of the list on the SharePoint server.

Result Set Columns

Name	Type	Description
Result	*String*	Boolean value indicating whether the operation was successful.

DeleteListColumn

Deletes a column from the specified list.

Input

Name	Type	Description
List	*String*	The name of the list on the SharePoint server.
ColumnName	*String*	The name of the column to delete.

Result Set Columns

Name	Type	Description
Result	*String*	Boolean value indicating whether the operation was successful.

DeleteUserFromGroup

Delete the user from specified group.

Input

Name	Type	Description
LoginName	*String*	The login name of the user.

Group	*String*	The group you are adding a user to and selecting or deleting the user from. This is an input-only value and either Group or Role must be specified for inserts and selects, but may be optionally specified for deletions.

Result Set Columns

Name	Type	Description
Result	*Boolean*	Boolean value indicating whether the stored procedure was successful.

DeleteUserFromRole

Delete the user from specified role.

Input

Name	Type	Description
LoginName	*String*	The login name of the user.
Role	*String*	The role you are adding a user to and selecting or deleting the user from. This is an input-only value and either Group or Role must be specified for inserts and selects, but may be optionally specified for deletions.

Result Set Columns

Name	Type	Description
Result	*Boolean*	Boolean value indicating whether the stored procedure was successful.

DiscardCheckOutDocument

Discards a check out on a document in SharePoint. This does not check a new file into SharePoint. It only releases the lock on the document.

Input

Name	Type	Description
Library	*String*	The name of the library on the SharePoint server.
RemoteFile	*String*	The name of the file being checked out.

Result Set Columns

Name	Type	Description
Result	*String*	Boolean value indicating whether the operation was successful.

DownloadAttachment

Download a document from the SharePoint list.

Input

Name	Type	Description
File	*String*	The path of the file to be saved.
RemoteFile	*String*	The path of the file on the server. This can be the full URL or simply the file name. If you use the name of the file, the latest version will be downloaded.

Result Set Columns

Name	Type	Description
Result	*String*	Boolean value indicating whether the operation was successful.

DownloadDocument

Download a document from the SharePoint library.

Input

Name	Type	Description
File	*String*	The path of the file to be saved.
Library	*String*	The name of the library on the SharePoint server.
RemoteFile	*String*	The path of the file on the server. This can be the full URL or simply the file name. If you use the name of the file, the latest version will be downloaded.

Result Set Columns

Name	Type	Description
Result	*String*	Boolean value indicating whether the operation was successful.

UpdateList

Updates a list on a SharePoint site.

Input

Name	Type	Description
List	*String*	The name of the list on the SharePoint server.
AllowMultiResponses	*String*	Set to True to allow multiple responses to the survey.
Description	*String*	A string that contains the description for the list.
Direction	*String*	A string that contains LTR if the reading order is left-to-right, RTL if it is right-to-left, or None.
EnableAssignedToEmail	*String*	Set to True to enable assigned-to e-mail for the issues list.
EnableAttachments	*String*	Set to True to enable attachments to items in the list. Does not apply to document libraries.

EnableModeration	*String*	Set to True to enable Content Approval for the list.
EnableVersioning	*String*	Set to True to enable versioning for the list.
Hidden	*String*	Set to True to hide the list so that it does not appear on the Documents and Lists page, Quick Launch bar, Modify Site Content page, or Add Column page as an option for lookup fields.
MultipleDataList	*String*	Set to True to specify that the list in a Meeting Workspace site contains data for multiple meeting instances within the site.
Ordered	*String*	Set to True to specify that the option to allow users to reorder items in the list is available on the Edit View page for the list.
ShowUser	*String*	Set to True to specify that names of users are shown in the results of the survey.
Title	*String*	A string that contains the title of the list.

Result Set Columns

Name	Type	Description
Result	*String*	Boolean value indicating whether the operation was successful.

UpdateListColumn

Updates a column to the specified list.

Input

Name	Type	Description
List	*String*	The name of the list on the SharePoint server.
ColumnName	*String*	The name of the column to update.

DisplayNam e	*String*	The updated value of the display name.
DefaultValue	*String*	The updated default value of the specified column.
ColumnType	*String*	The updated data type of the specified column.
MaxLength	*String*	The updated maximum length of the specified column.
PrimaryKey	*String*	Use this to make or not the exisitng column, primary key.
ReadOnly	*String*	Use this to make or not the exisitng column, readonly.
Required	*String*	Use this to make or not the exisitng column, required.

Result Set Columns

Name	Type	Description
Result	*String*	Boolean value indicating whether the operation was successful.

UploadDocument

Upload a document to the SharePoint library.

Input

Name	Type	Description
File	*String*	The path of the file to be added.
FileContent	*String*	Base64 encoded content of the file to be added. If specified, the value of 'File' input will be ignored.
Library	*String*	The name of the library on the SharePoint server.
Name	*String*	The name assigned to the new file.
ReturnID	*String*	Boolean value indicating whether to return the ID and other metadata fields of the newly created record. An extra API request is needed to get this value. The default value is *false*.

MetadataName#	*String*	The name of a metadata field to be set for the document.
MetadataValue#	*String*	The value of a metadata field to be set for the document.

Result Set Columns

Name	Type	Description
Result	*String*	Boolean value indicating whether the stored procedure was successful.
ID	*String*	The internal ID of the document.
_dlc_DocId	*String*	The document ID.
_dlc_DocIdUrl	*String*	The URL of the document ID.
Vti_author	*String*	The creator of the document.
Vti_etag	*String*	The e-tag of the document.
FileRef	*String*	The file reference of the document.

REST Data Model

The SharePoint Adapter models SharePoint entities in relational Tables, Views, and Stored Procedures. The table definitions are dynamically obtained based on your SharePoint site. Any changes you make, such as adding a custom field or changing a field's data type, are automatically reflected when you connect.

Tables

Tables describes the available tables.

Lists in your sharepoint site are exposed as relational tables dynamically. Which means any change you make in your lists, i.e adding new list or adding new fields, will be reflected on the driver.

Views

Views are tables that cannot be modified. Typically, read-only data are shown as views.

Stored Procedures

Stored Procedures are function-like interfaces to the data source. They surface additional capabilities of the SharePoint API such as searching, updating, and modifying information.

Using OData standard

Since the REST API is OData based, server side filters, are done using OData standard. So the driver takes the most of the server filtering, by reading the metadata file and determing which filters can be done on the server.

NOTE: When executing "SELECT *" queries, the SharePoint REST API response, does not return all the available fields. So to avoid too many null values, the provider will select all the columns explicitly using the **$select** filter. However, the provider will do this only if the **$select** filter's length is not bigger than 1500, to avoid an error from SharePoint REST API regarding the URL length. This is a limitation of the SharePoint REST API, so in these cases, the only way to see the actual value of some columns, is to explicitly select them in your query.

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. To find out more about tables and stored procedures, please navigate to their corresponding entries in this help document.

SharePoint Adapter Views

Name	Description
Files	Query the available files on your sharepoint site.
Groups	Query the available groups on your sharepoint site.
Lists	Query the available lists on your sharepoint site.
Roles	Query the roles your users can have.
Users	Query the available users on your sharepoint site.

Files

Query the available files on your sharepoint site.

Columns

Name	Type	Description
Id [KEY]	*String*	
CreatedBy_Id	*String*	
CreatedBy_Name	*String*	
CreatedBy_Puid	*String*	
ETag	*String*	
LastModifiedBy_Id	*String*	
LastModifiedBy_Name	*String*	
LastModifiedBy_Puid	*String*	
Name	*String*	
Size	*Int*	
TimeCreated	*Datetime*	
TimeLastModified	*Datetime*	
Url	*String*	

Groups

Query the available groups on your sharepoint site.

Columns

Name	Type	Description
AllowMembersEditMembership	*Bool*	
AllowRequestToJoinLeave	*Bool*	
AutoAcceptRequestToJoinLeave	*Bool*	
CanCurrentUserEditMembership	*Bool*	
CanCurrentUserManageGroup	*Bool*	
CanCurrentUserViewMembership	*Bool*	
Description	*String*	
OnlyAllowMembersViewMembership	*Bool*	
OwnerTitle	*String*	
RequestToJoinLeaveEmailSetting	*String*	
LinkedOwner	*String*	
LinkedUsers	*String*	

Lists

Query the available lists on your sharepoint site.

Columns

Name	Type	Description
HasUniqueRoleAssignments [KEY]	*Bool*	
LinkedFirstUniqueAncestorSecurableObject	*String*	
LinkedRoleAssignments	*String*	
Id [KEY]	*String*	
AllowContentTypes	*Bool*	

AllowDeletion	*Bool*
BaseTemplate	*Int*
BaseType	*Int*
BrowserFileHandling	*Int*
ContentTypesEnabled	*Bool*
CrawlNonDefaultViews	*Bool*
Created	*Datetime*
CurrentChangeToken_StringValue	*String*
CustomActionElements_Items	*String*
DataSource_Properties	*String*
DefaultContentApprovalWorkflowId	*String*
DefaultDisplayFormUrl	*String*
DefaultEditFormUrl	*String*
DefaultItemOpenUseListSetting	*Bool*
DefaultNewFormUrl	*String*
DefaultViewPath_DecodedUrl	*String*
DefaultViewUrl	*String*
Description	*String*
Direction	*String*
DisableGridEditing	*Bool*
DocumentTemplateUrl	*String*
DraftVersionVisibility	*Int*
EffectiveBasePermissions_High	*Long*
EffectiveBasePermissions_Low	*Long*

EffectiveBasePermissionsForUI_High	*Long*
EffectiveBasePermissionsForUI_Low	*Long*
EnableAssignToEmail	*Bool*
EnableAttachments	*Bool*
EnableFolderCreation	*Bool*
EnableMinorVersions	*Bool*
EnableModeration	*Bool*
EnableRequestSignOff	*Bool*
EnableVersioning	*Bool*
EntityTypeName	*String*
ExcludeFromOfflineClient	*Bool*
ExemptFromBlockDownloadOfNonViewableFiles	*Bool*
FileSavePostProcessingEnabled	*Bool*
ForceCheckout	*Bool*
HasExternalDataSource	*Bool*
Hidden	*Bool*
ImagePath_DecodedUrl	*String*
ImageUrl	*String*
IrmEnabled	*Bool*
IrmExpire	*Bool*
IrmReject	*Bool*
IsApplicationList	*Bool*
IsCatalog	*Bool*
IsEnterpriseGalleryLibrary	*Bool*

IsPrivate	*Bool*
IsSiteAssetsLibrary	*Bool*
IsSystemList	*Bool*
ItemCount	*Int*
LastItemDeletedDate	*Datetime*
LastItemModifiedDate	*Datetime*
LastItemUserModifiedDate	*Datetime*
ListExperienceOptions	*Int*
ListItemEntityTypeFullName	*String*
MajorVersionLimit	*Int*
MajorWithMinorVersionsLimit	*Int*
MultipleDataList	*Bool*
NoCrawl	*Bool*
OnQuickLaunch	*Bool*
PageRenderType	*Int*
ParentWebPath_DecodedUrl	*String*
ParentWebUrl	*String*
ParserDisabled	*Bool*
ReadSecurity	*Int*
SchemaXml	*String*
ServerTemplateCanCreateFolders	*Bool*
TemplateFeatureId	*String*
Title	*String*
ValidationFormula	*String*

ValidationMessage	*String*
WriteSecurity	*Int*
LinkedContentTypes	*String*
LinkedCreatablesInfo	*String*
LinkedDefaultView	*String*
LinkedDescriptionResource	*String*
LinkedEventReceivers	*String*
LinkedFields	*String*
LinkedForms	*String*
LinkedInformationRightsManagementSettings	*String*
LinkedItems	*String*
LinkedParentWeb	*String*
LinkedRootFolder	*String*
LinkedSubscriptions	*String*
LinkedTitleResource	*String*
LinkedUserCustomActions	*String*
LinkedViews	*String*
LinkedWorkflowAssociations	*String*

Roles

Query the roles your users can have.

Columns

Name	Type	Description
Id [KEY]	*Int*	
BasePermissions_High	*Long*	
BasePermissions_Low	*Long*	
Description	*String*	
Hidden	*Bool*	
Name	*String*	
Order	*Int*	
RoleTypeKind	*Int*	

Users

Query the available users on your sharepoint site.

Columns

Name	Type	Description
AadObjectId_NameId	*String*	
AadObjectId_NameIdIssuer	*String*	
Email	*String*	
Expiration	*String*	
IsEmailAuthenticationGuestUser	*Bool*	
IsShareByEmailGuestUser	*Bool*	
IsSiteAdmin	*Bool*	
UserId_NameId	*String*	
UserId_NameIdIssuer	*String*	
UserPrincipalName	*String*	
LinkedAlerts	*String*	
LinkedGroups	*String*	

Stored Procedures

Stored procedures are available to complement the data available from the SOAP Data Model. It may be necessary to update data available from a view using a stored procedure because the data does not provide for direct, table-like, two-way updates. In these situations, the retrieval of the data is done using the appropriate view or table, while the update is done by calling a stored procedure. Stored procedures take a list of parameters and return back a dataset that contains the collection of tuples that constitute the response.

SharePoint Adapter Stored Procedures

Name	Description
DownloadAttachment	Download a document from the SharePoint list.
DownloadDocument	Download a document from the SharePoint library.
GetAdminConsentURL	Gets the admin consent URL that must be opened separately by an admin of a given domain to grant access to your application. Only needed when using custom OAuth credentials.
GetCurrentUser	Retrieves information about the current logged in user.
GetOAuthAccessToken	Gets the OAuth access token from SharePoint.
GetOAuthAuthorizationURL	Gets the SharePoint authorization URL. Access the URL returned in the output in a Web browser. This requests the access token that can be used as part of the connection string to SharePoint.
RefreshOAuthAccessToken	Refreshes the OAuth access token used for authentication with SharePoint.

DownloadAttachment

Download a document from the SharePoint list.

Stored Procedure Specific Information

Insert

RemoteFile can be both relative to the server, or it can be the full URL of the file. Below are some examples:

```
EXECUTE DownloadAttachment File =
'C:/Users/User/Desktop/DownloadedFile.txt', RemoteFile =
'https://mysite.sharepoint.com/Lists/MyCustomList/Attachments/1/Fi
leToDownload.txt';

EXECUTE DownloadAttachment File =
'C:/Users/User/Desktop/DownloadedFile.txt', RemoteFile =
'/Lists/MyCustomList/Attachments/1/FileToDownload.txt';
```

Input

Name	Type	Description
File	*String*	The path of the file to be saved. You should include the new filename. For example, 'C:/Users/User/Desktop/DownloadedFile.txt'.
RemoteFile	*String*	The path of the file on the server. This can be the full URL or the path relative to the server.

Result Set Columns

Name	Type	Description
Result	*String*	Boolean value indicating whether the operation was successful.

DownloadDocument

Download a document from the SharePoint library.

Stored Procedure Specific Information

Insert

RemoteFile can be both relative to the library, or it can be the full URL of the file. Below are some examples:

```
EXECUTE DownloadAttachment File =
'C:/Users/User/Desktop/DownloadedFile.txt', Library = 'Shared
Documents', RemoteFile = 'https://mysite.sharepoint.com/Shared
Documents/newFolder/FileToDownload.txt';

EXECUTE DownloadAttachment File =
'C:/Users/User/Desktop/DownloadedFile.txt', Library = 'Shared
Documents', RemoteFile = '/newFolder/FileToDownload.txt';
```

Input

Name	Type	Description

File	*String*	The path of the file to be saved. You should include the new filename. For example, 'C:/Users/User/Desktop/DownloadedFile.txt'.
Library	*String*	The name of the library on the SharePoint server. For example, 'Shared Documents'.
RemoteFile	*String*	This can be either the relative path to the library or the full URL of the file.

Result Set Columns

Name	Type	Description
Result	*String*	Boolean value indicating whether the operation was successful.

GetAdminConsentURL

Gets the admin consent URL that must be opened separately by an admin of a given domain to grant access to your application. Only needed when using custom OAuth credentials.

Input

Name	Type	Description
CallbackUrl	*String*	The URL the user will be redirected to after authorizing your application. This value must match the Reply URL in the Azure AD app settings.
State	*String*	The same value for state that you sent when you requested the authorization code.
Scope	*String*	The scope or permissions you are requesting from the Admin The default value is *AllSites.Manage*.

Result Set Columns

Name	Type	Description

URL	*String*	The authorization URL, entered into a Web browser to obtain the verifier token and authorize your app.

GetCurrentUser

Retrieves information about the current logged in user.

Result Set Columns

Name	Type	Description
Id	*Int*	The id of the user.
Title	*String*	The title of the user.
Email	*String*	The email of the user.
IsSiteAdmin	*Boolean*	Whether the user is a site admin or not.

GetOAuthAccessToken

Gets the OAuth access token from SharePoint.

Input

Name	Type	Description
AuthMode	*String*	The type of authentication mode to use. The allowed values are APP, WEB.
Verifier	*String*	The verifier token returned by SharePoint after using the URL obtained with GetOAuthAuthorizationURL. Required for only the Web AuthMode.
CallbackUrl	*String*	The URL the user will be redirected to after authorizing your application.
Scope	*String*	The scope or permissions you are requesting from the user. The default value is *AllSites.Manage*.
State	*String*	Any value that you wish to be sent with the callback.

Result Set Columns

Name	Type	Description
OAuthAccessToken	*String*	The authentication token returned from SharePoint.
OAuthRefreshToken	*String*	A token that may be used to obtain a new access token.
ExpiresIn	*String*	The remaining lifetime for the access token in seconds.

GetOAuthAuthorizationURL

Gets the SharePoint authorization URL. Access the URL returned in the output in a Web browser. This requests the access token that can be used as part of the connection string to SharePoint.

Input

Name	Type	Description
CallbackUrl	*String*	The URL that Sharepoint will return to after the user has authorized your app.
Scope	*String*	The scope or permissions you are requesting from the user. The default value is *AllSites.Manage*.
State	*String*	Any value that you wish to be sent with the callback.

Result Set Columns

Name	Type	Description
URL	*String*	The URL to be entered into a Web browser to obtain the verifier token and authorize the data provider with.

RefreshOAuthAccessToken

Refreshes the OAuth access token used for authentication with SharePoint.

Input

Name	Type	Description
OAuthRefreshToken	*String*	The old token to be refreshed.

Result Set Columns

Name	Type	Description
OAuthAccessToken	*String*	The authentication token returned from SharePoint.
ExpiresIn	*String*	The remaining lifetime on the access token.

Data Type Mapping

Data Type Mappings

The adapter maps types from the data source to the corresponding data type available in the schema. The table below documents these mappings.

SharePoint	CData Schema
Choice (menu)	string
Currency	float
Date and Time	datetime
Hyperlink or Picture	string
Lookup	string
Multiple lines of text	string
Number	float
Person or Group	string
Single line of text	string

Task Outcome	string
Yes/No	bool