

SQL



This page explains the **SQL Adapter** in Bridge context. If you were looking for the same information regarding the [PAS Designer](#), refer to [SQL Adapter](#) in the Designer guide.

Tagged Values

<<SQL Alias>>

Tagged Value	Description	Allowed Values
General		
dbConnectionString	The format of the database connection string depends on the type of the database. For more details see Database Server-Specific Notes for SQL Adapters .	
dbType	Type of the database.	Oracle, SQLServer, InterBase, SQLBase, ODBC, DB2, Informix, Sybase, MySQL, PostgreSQL, SQLite, DbTypeVariable
dbTypeVariable	<p>If the tagged value dbType is set to DbTypeValue, the dbTypeVariable tagged value is used to define the type of the database. The dbType then can be defined by a setting variable.</p> <p>This is to handle the case, that you not want to hard code the dbType, but to configure it at runtime via the E2E Bridge.</p> <p>See Using Global Setting Variables for more information on how to define a global setting variable in the E2E Bridge.</p>	<p>Any global setting variable from the E2E Bridge.</p> <p>Example: <code>{{my_setting_variable}}</code></p>
user	DB user. Optional the password can be given after a '/'. However, this is recommended for development purposes only.	Example: <code>{{DB_USER}} / {{DB_PASSWORD}}</code>
options	This tagged value can hold a comma separated list of <name>=<value> pairs. These list elements are interpreted as native options. The possible name-value pairs depend on the database type. A comprehensive list can be found at https://www.sqlapi.com/ApiDoc/servers/	Example: <code>SSPROP_INIT_ENCRYPT=VARIANT_TRUE</code>
transactionIsolationLevel	<p>Bridge 7 Specify here the required transaction isolation level of the SQL connection according to SQL-92 standard. Refer to Wikipedia for a detailed description of the available isolation levels.</p> <p>Please note that not all databases support all levels. In this case a database-specific mapping will occur.</p> <p>For persistent state databases no other than <UNSPECIFIED> and DB MS default are allowed.</p>	<p>Default Use the default isolation level of the connected database system.</p> <p>Read Dirty Lowest isolation level. Dirty reads allowed, SQL adapter may fetch not-yet-committed changes of other transactions.</p>

On this Page:

- [Tagged Values](#)
 - [<<SQL Alias>>](#)
 - [<<SQL Adapter>>](#)
 - [<<SQLConfigurationAdapter>>](#)
- [SQL Adapter Parameters](#)
- [SQL Configuration Adapter Parameters](#)
- [SQL Adapter Parameter Types](#)
 - [SQLConfiguration](#)
 - [SQLAuthentication](#)

Related Pages:

- [Database Server-Specific Notes for SQL Adapters](#)
- [Database-Specific Mappings](#)
- [Charset Definitions](#)
- [Time Zones](#)
- [Using Global Setting Variables](#)
- [SQL Options Documentation](#)
- [Troubleshooting the SQL Adapter](#)
- [Database Isolation Levels \(Wikipedia\)](#)

		R e a d c o m m i t t e d	Lock-based concurrency control.
		R e p e a t a b l e r e a d	Lock-based concurrency control.
		S e r i a l i z a b l e	Highest isolation level. Lock-based concurrency control.
Localization			
charset	Any database uses a charset to encode Strings. If the database uses UNICODE charsets (UTF-8, UTF-16, UTF-32), encoding is handled automatically. If the database is not UNICODE compliant, the Bridge assumes 7-bit ASCII by default. However, in many cases it necessary to define the charset explicitly. This is done by the tagged value charset as shown below. The charset needs to be the same as defined at the database settings. All possible charset definitions are listed in section Charset Definitions .	Example: UTF-8 See Charset Definitions for a list of possible values.	
timezone	You can enter a valid time zone or the value local , which uses the time zone of the xUML service. See Time Zones for a list of possible values. If timezone does not contain any content (is NULL), UTC is used.	D e f a u l t i s N U L L	Example: "Australia/Melbourne", "CET", "Etc/GMT+10"
unicodeMode	Added in Builder 6.0.15.5 Runtime 2015.15 Specify the encoding for database access. <div>We recommend to use the Platform default unless you suspect an encoding incompatibility (see Troubleshooting the SQL Adapter). This option represents the former behavior and is fully backwards-compatible - means, it can be used with older xUML Runtimes. The two other (force mode) options will be ignored by older Runtimes without warning.</div>	P l a t f o r m d e f a u l t (d e f a u l t)	Use the platform default mode. This is <ul style="list-style-type: none">Unicode: for Windows systemsnon-Unicode: for all others This option is backwards compatible to older Runtimes.
		U n i c o d e	Force Unicode mode.
		n o n - U n i c o d e	Force non-Unicode mode.
Connection Pooling			
connectionPooling	Added in Builder 5.1.8.58 Runtime 5.1.82.0 This tagged value controls the connection pooling. If true, each connection is put into a pool after use. If an SQL adapter requires a connection, it is taken from the pool. If no connection is available, a new connection is being created and put into the pool after	tr ue	Database connections are pooled.

	use. The time the connection is kept in the pool depends on the other pooling parameters.	false	Database connections are not pooled.
maxConnectionAge	After a given connection age (in minutes) the connection will be closed and removed from the pool.	Connection age in minutes, default is 15 minutes , -1 means forever.	
maxConnectionIdleTime	Connections not used for the time specified (in minutes) will be closed and removed from the pool. This is useful for connections going through firewalls because such connections might be cut off after some time.	Values in minutes, default is 60 .	
maxConnectionReuse	This tagged value controls how often a connection can be re-used. After the connection has been re-used for maxConnectionReuse , it will be closed and not put back into the pool. This feature has been introduced because some databases had problems if the connection was re-used too often. Value -1 means the connection will be re-used forever. In this case you should define reasonable values for maxConnectionAge or maxConnectionIdleTime (see above).	0	pooling is implicitly switched off.
		-1	connections are pooled forever
		available	number of connections to be pooled, default is 1000.
<div>Note that the pooling is implicitly switched off, if maxConnectionReuse is set to 0.</div>			
Qualifier			
schema	String that prefixes tables and stored procedures. For example, if schema is set to S1, all tables accessing the current DB are prefixed by "S1". <div>This works only if the tables are marked using the <code>TABLE::</code> keyword, e.g. <code>TABLE::EMPLOYEE</code> in SQL statements. If you do not prefix the table name by <code>TABLE::</code>, the tablename is used as it is.</div>		
tableQualifier	String that prefixes tables. For example, if tableQualifier is set to TQ1, all tables accessing the current DB are prefixed by "TQ1", e.g. <code>TQ1EMPLOYEE</code> . If schema and table qualifier are given, all tables will become: <code><schema>.<tableQualifier>.<tableName></code> . <div>This works only if the tables are marked using the <code>TABLE::</code> keyword, e.g. <code>TABLE::EMPLOYEE</code> in SQL statements. If you do not prefix the table name by <code>TABLE::</code>, the tablename is used as it is.</div>		

<<SQL Adapter>>

Tagged Value	Description	Allowed Values	
alias	Specify the SQL alias resp. the database the adapter should connect to.	any valid SQL alias	
action	Holds the action to perform on the database.	execute (default)	Execute an SQL statement.
		getHandle	Get a connection handle for subsequent fetchNext actions.
		fetchNext	Fetch next record.
		closeHandle	Close the connection handle. If all records have been fetched, the handle is closed automatically.
dbType	Overwrite the database type defined in the SQL alias.		
prefetchedRecords	Number of pre-fetched records. This makes sense for SQL queries, especially when using getHandle / fetchNext actions for bulk fetch use cases.		
sql	Holds the SQL statement to be performed on the database.	Any valid SQL statement as a string.	

<<SQLConfigurationAdapter>>

Tagged Value	Description	Allowed Values	
alias	Specify the SQL alias resp. the database to configure	any valid SQL alias	
action	Specify the configuration action.	setAuthentication (default)	Set the authentication configuration.

SQL Adapter Parameters

Name	Type	Direction	Description
sql	String	in	Use this parameter to provide a dynamic SQL statement.
inputBindings	Map	in	Use this parameter to provide parameter/value pairs for parameterized statements.
affectedRows	Integer	out	This parameter returns the number of rows affected by the SQL statement.
	any class	out	Result set of the database query. The SQL Adapter tries to match the table column names with the attribute names of the output class. For information on type mapping refer to Database-Specific Mappings .

SQL Configuration Adapter Parameters

Name	Type	Direction	Description
configuration	SQLConfiguration	in	Use this parameter to provide SQL connection configurations. At the moment these are authentication configurations only.

SQL Adapter Parameter Types

SQLConfiguration

Attribute	Type	Description	Values/Example
authentication	SQLAuthentication	An object containing the user and the password.	

SQLAuthentication

Attribute	Type	Description	Values/Example
password	String	Password.	
username	String	Username.	