Installing and Configuring Database Access

This section describes all steps required to run a service that connects to databases:

- Installing SQL client tools
- · Defining the SQL adapter preferences
- Defining database parameters

When installing the BRIDGE, all additional libraries (SQL Libraries, SAP Libraries, and Java Libraries) are being installed as well.

After a successful installation, the BRIDGE is capable to connect to database backends.

First, you need to install the client tools or the ODBC drivers and configure each database access if needed.

Databases like DB2 or Oracle are usually running on a dedicated database server. The BRIDGE opens a connection to the database server via the database client tools or ODBC.

Prerequisites

The BRIDGE has been successfully installed. When running the BRIDGE in server mode (as opposed to workstation mode), the BRIDGE node instance has to be a member of a BRIDGE domain. For more information on installation modes and BRIDGE domains, refer to Installation Modes and BRIDGE Domain.

Supported Databases

Database	Versions						
DB2	DB2 CLI version 6.x or higher						
Informix	Informix CLI version 2.x or higher						
InterBase	InterBase client API version 5.x or higher, all Firebird versions						
Microsoft SQL Server	ODBC: SQLServer ODBC version 2000.x or SQL Server Native Client (ODBC API) version 2005 and higher						
	OLE DB: SQL Server OLE DB version 2.5 or higher						
MySQL	MySQL C API version 3.23.x or higher						
Oracle	Oracle OCI version 8 or higher						
PostgreSQL	libpq version 7.1.x or higher						
SQLBase	CAPI version 6.x or higher						
SQLite	libsqlite version 3.x or higher						
Sybase	Open Client version 10.0 or higher						

On this Page:

- Prerequisites

 Supported
 Databases
- Installing Database Access
 Defining the ODBC Database Parameters

Related Pages:

- Installing and Configuring Database Access for DB2
- Installing and Configuring Database Access for Informix
- Installing and Configuring Database Access for Microsoft SQL Server
- Installing and Configuring Database Access for MySQL
- Installing and Configuring Database Access for Oracle
- Troubleshooting Database
 Access
- Installation Modes and BRIDGE Domain

Related Documentation:

- Database-Specific
- Mappings
- xUML Service Settings

Installing Database Access

- To install database access to a DB2 database refer to Installing and Configuring Database Access for DB2.
- To install database access to an Oracle database refer to Installing and Configuring Database Access for Oracle.
- To install database access to a MySQL database refer to Installing and Configuring Database Access for MySQL.

If you want to access the installed database via the xUML RUNTIME that is embedded in **Scheer PAS** *B UILDER*, please note that you have to make the database client available to the BUILDER:

- Install a database client to the system you are running the BUILDER on.
- Make sure that the database client location makes part of the PATH variable. After having changed PATH, you need to restart MagicDraw to make it read the new content of P ATH.

Defining the ODBC Database Parameters

All well-known databases like Oracle, DB2, Inter Base, Firebird, dBase, Access, FoxPro, Paradox, SQL Server, and so on can be connected via ODBC. Install the required ODBC drivers and define the parameters for each connection.

Before starting a deployed xUML service that connects to a database backend, you may want to redefine required database parameters in the BRIDGE. You find more information about database-specific information (database names and mappings) in section Database-Specific Mappings.

Select the xUML service in the navigation (in this example CustomerQuery and stop the service.

UML Servi	ce Documentation History Logg	ng Dump S	lettings	Version	Persistent State Status		
Status					Preferences		
Running (PD: 12910) Start Stop Delete Kill Expert				iport	∴ Tracing and Mockaps Enabled		
					Dump Context on Error		
					Bridge Server Log Level	Info	
Comilar							
Service	e Instances				Transaction Log Level	None	
Service Activation	Label	Stereotype	Port	WSDL	Transaction Log Level Transaction Log Rotation	Note	
Activation	Label sLMLControllerService.sLMLControllerService	Stereotype E2ERESTService	Port 29021	WSD4	Transaction Log Level Transaction Log Rotation Interval	None DAILY	

Switch to tab Settings. From the dropdown list, select the option SQL Adapter Connection in the Addons category.

XUML Service I Docur	nentation History Logo	ina Dumo	Settings	Version	Persistent State	Statu	
xUML Service S	Settings						
					_		
	Settings / Deployment Macros	✓ View Eq			52)		
Show 10 v entries	Global Settings		Filter:				
	Settings / Deployment Macros						
кеу	Add-Ons			Original Value	e în Model		
getCompositeName:	Dump Configuration			CustomerOu	PEV		
geroompositeiriane.	Persistent State			oustonnerqu	ory.		
	REST HTTP Server	linglaugh in hol	d Turno (to a	at a list of varia	blas		
	REST Service Base	isplayed in boi	u. Type (to gi		Dieb.		
Showing 1 to 1 of 1 entries	SAP Global				Previous 1	Next	
	SOAP HTTP Server					_	
	SOAP HTTP Service Request					Apply	
	SOAP HTTP UI Server						

The default parameters are defined in the component diagram of the xUML service (see CustomerQuery example below). The connection is defined by a dependency between the composite named **CustomerQ uery** and the SQL alias **customers**. The database user and password are defined in the tagged value **us er** on the SQL alias.

«E2EComposite» CustomerQuery (category = "Services", controlPort = 29021, kilUVMThreadsOnShutdown, version = "7.0.0", wsdIPerService)	
«E2ESOAPService» ?₀ SqlOdbcService (port = 19021) «E2ESOAPPortType» :: CustomerQueryPort	connectionPooling, dbConnectionString = "NorthwindCustomers", dbType = ODBC, user = "user/password"}

CustomerQuery

UML Service	Documentation	History	Logging	Dump	Settings	Version	Persistent State	Stat
xUML Serv	vice Settings							
	SQL Adapte	r Connection	~	View E	xport as CSV (charset=CP12	52)	
how 10 $ \sim $ entri	es					Filt	er:	
Key		Current	Value			Original Valu	e in Model	
customers: Conn false):	ection Pooling (true	true				true		
customers: DB N	ame:	Northwi	indCustomers			NorthwindC	ustomers	
customers: DB T	/pe:	ODBC				ODBC		
customers: Max (minutes):	15	15				15		
customers: Max (minutes):	Connection Idle Time	60				60		
customers: Max	Connection Reuse:	1000				1000		
		Changed	l values are dis	splayed in bo	old. Type { to g	et a list of varia	ables.	
howing 1 to 6 of 6	5 entries						Previous 1	Next
								Apply

Enter the following database parameters and click Apply:

Key	Value
custo mers: DBTy pe	Name of the SQL Service component e.g. ODBC
custo mers: DBNa me	Name of the database, e.g. NorthwindCustomers
custo mers: Conn ectio n Pooli ng	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 use. The time the connection is kept in the pool depends on the other pooling parameters.
custo mers: Max Conn ectio n Reuse	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).
	Note that the pooling is implicitly switched off, if maxConnectionReuse is set to 0.
custo mers: Max Conn ectio n Age	After a given connection age (in minutes) the connection will be closed and removed from the pool.
custo mers: Max Conn ectio n Idle Time	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.

1

(i) For each SQL adapter alias found in the activity diagrams of a UML model, you will find a pair of the two parameters **DBType** and **DBName** on the SQL adapter connection settings. According to the example with the both parameters in the table above, you would find an alias called customers in the UML model of the deployed xUML service CustomerQuery.

Now, select SQL Adapter Authorization from the list.

CustomerQuery								
xUML Service	Documentation	History	Logging	Dump	Settings	Version	Persistent State	Status
xUML Service Settings								
SQL Adapter Authorization View Export as CSV (charset + CP1252)								
Show 10 ${\scriptstyle \lor}$ entrie	rs					Filt	er:	
Key		Current \	/alue			Original Value	e in Model	
customers: DBPa	ssword:	•••••	••					
customers: DBUse	er:	user				user		
		Changed	values are dis	played in bo	ld. Type { to ge	t a list of varia	bles.	
Showing 1 to 2 of 2	entries						Previous 1	Next

Enter the following database parameters and click **Apply**:

Кеу	Value
customers: DBUser	Database user
customers: DBPassword	Database password

0

Users who do not have the permission to view the settings cannot see the **Settings** tab at all. This prevents that unauthorized users access sensitive information like passwords, etc.

For more details on changing the settings refer to xUML Service Settings.