

Using Global Setting Variables

To manage overall settings that are used by multiple services, the Bridge provides the possibility to define setting variables. This is helpful e.g. when several services are using the same settings.

If, for example, several services are using the same backend and the backend access parameters are managed via the service settings, you can define the backend specific settings once in a set of setting variables. Then, on service level, you use the setting variables. Every time the backend parameters change, you only will only have to change the setting variables and the changed settings will automatically be applied to all concerned services.

On this Page:

- [Defining Setting Variables](#)
- [Using Setting Variables](#)
- [Using Setting Variables in a Service Model](#)

Defining Setting Variables

On Bridge level, you can define setting variables. Setting variables can then be used on xUML service level.

To define setting variables, select the node instance, on which the Bridge is running. Then, click the sub-navigation item **xUML Service** and select the tab **Setting Variables**.

Figure: Tab Setting Variables on the Bridge

xUML Services

Global Setting Variables

Name	Value
------	-------

[Add New Variable](#) [Apply](#)

By using the button **Add New Variable**, you can define one or more new setting variables.

Figure: Defining a New Settings Variable

xUML Services

Global Setting Variables

Name	Value
database	NorthwindCustomers
password	admin
user	sa

[Add New Variable](#) [Apply](#)

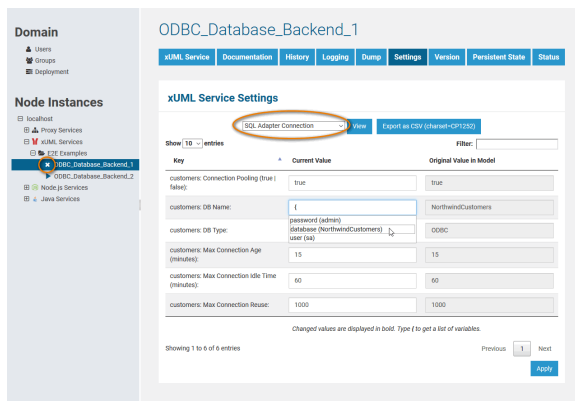
The values in bold are indicating, that these changes have not been saved yet. Click **Apply** to persist all your changes.

All running services that are using the setting variable will still use the old setting value until they are restarted.

Using Setting Variables

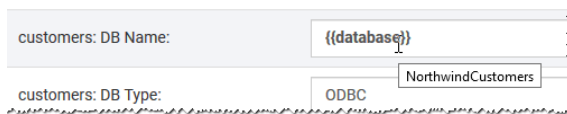
Switch to the service you want to use a setting variable with and stop it (**ODBC_DatabaseBackend_1** in the example below). Settings can only be changed on stopped services. Then, switch to the **Settings** tab and select the type of settings you want to change (**SQL Adapter Connection** in the example below).

Figure: Applying a Setting Variable



Start typing **{** in the value column of the setting you want to change. A drop down box appears that contains all defined setting variables. Select the one suitable for this setting and the value will be replaced by the name of the variable (**{{database}}**) in the example above). Click **Apply** to save your changes. The original value of the setting coming from the service model will still be displayed in a separate column on the right. The bold value of the setting indicates, that this setting has been changed from its original value.

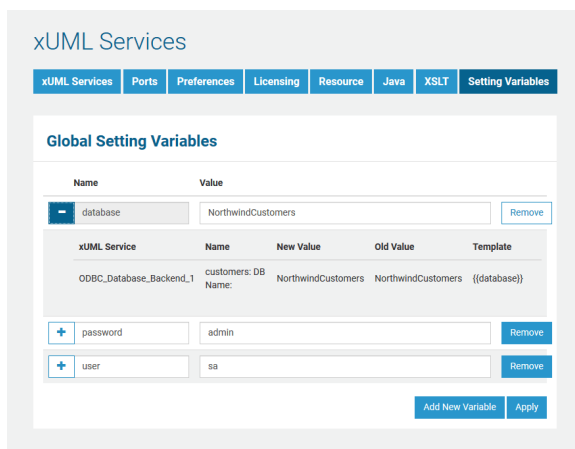
When you hover with your mouse over a variable, a tool tip appears showing the actual value of the variable from the global definition in the **Setting Variables** tab.



If the value of a setting variable has changed, the tooltip will show the new (changed) value. Nevertheless, the service will use the old setting value as long as the service has not been restarted.

Switch back to tab **Setting Variables** of the Bridge. A plus sign next to variable **database** indicates, that this variable is used by one or more services. Click the plus sign to expand a list of services using this variable.

Figure: List of Services Using a Setting Variable



You cannot remove setting variables that are in use.

Using Setting Variables in a Service Model

Instead of providing concrete values, you can use setting variables in your service model.

Figure: Setting Variables in the Component Diagram of an xUML Service Model



```
{charset = "utf-8",
connectionPooling,
dbConnectionString = "{{database}}"
dbType = ODBC,
user = "{{user}}{{password}}"}

```

Just like using setting variables in the Bridge, enclose the name of the setting variable with double curly brackets.

Analogous, you can also use setting variables in Action Script like in the example below:

```
local gratification = setting("Gratification",
"{{Global_gratification_setting}}");

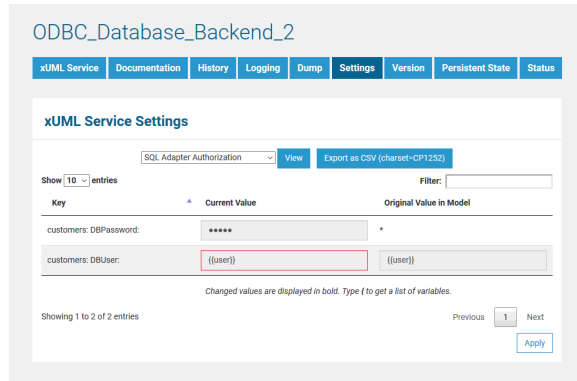
```

Setting variables used in a service model are checked when you deploy the service. If you use setting variables that are not defined, you will get a warning: **The following setting variables are undefined: {{password}}, {{user}}**. The concerned service will nevertheless start up and run, if no other errors prevent this.

If you just ignore this warning, the name of the setting variable will be used as a setting value.

Undefined setting variables are display in red on the **Settings** tab of the Bridge.

Figure: Missing Setting Variables in an xUML Service Setting



Undefined setting variables coming from an xUML model will get automatically created as a global setting variable with an empty value in the Bridge.

Figure: Automatically Created Setting Variables

Global Setting Variables

Name	Value
+ database	NorthwindCustomers Remove
+ password	Remove

xUML Service	Name	New Value	Old Value	Template
ODBC_Database_Backend_2	customers: DBPassword:	{{password}}	{{password}}	{{password}}

+ user	Remove
--------	------------------------

xUML Service	Name	New Value	Old Value	Template
ODBC_Database_Backend_2	customers: DBUser:	{{user}}	{{user}}	{{user}}

[Add New Variable](#) [Apply](#)

In the example above, the setting variables **password** and **user** are used in the listed xUML service, but have no value assigned to. To indicate this problem, the services list of these variables is expanded and the variable value is displayed in red.