

# Conditional Processing

Only in DbVisualizer Pro



This document and the Database Profile Framework in general is appropriate only when using the licensed DbVisualizer Pro edition.

- [Introduction](#)
- [Conditional processing when database connection is established](#)
- [Conditional processing during command execution](#)

## Introduction

Conditional processing simply means that a profile can adjust its content based on certain conditions. A few examples:

- Which version of the database is being accessed
- The format of the database URL
- The client environment i.e Java version, vendor, etc.
- User properties
- Database connection properties

Conditional processing is especially useful when adapting the profile for different versions of the database (and/or JDBC driver). Another use is to replace generic error messages with more user friendly messages.

If you have some programming skills conditions are expressed using **if**, **else if** and **else** statements.

There are two phases when conditions are processed:

1. **Conditional processing when database connection is established**  
**If**, **Elseif** and **Else** elements can be specified almost everywhere in the profile
2. **Conditional processing during command execution**

The **OnError** element is used to define a message that will appear in DbVisualizer if a command fails. Conditions are used to control what message should appear

DbVisualizer uses the **type** attribute to determine which **If** elements should be executed in which of the two phases. If this attribute is set to the value **runtime**, it will be processed in the second phase. If it is not specified, it will be processed in the first phase.

## Conditional processing when database connection is established

These are the call signatures for the utilities used when processing conditions:

```
boolean #util.isNull(String string)
boolean #util.isNullOrWhitespace(String string)
boolean #util.isDatabaseVersionLT(Integer major)
boolean #util.isDatabaseVersionLT(Integer major, Integer minor)
boolean #util.isDatabaseVersionLTE(Integer major)
boolean #util.isDatabaseVersionLTE(Integer major, Integer minor)
boolean #util.isDatabaseVersionEQ(Integer major)
boolean #util.isDatabaseVersionEQ(Integer major, Integer minor)
boolean #util.isDatabaseVersionGTE(Integer major)
boolean #util.isDatabaseVersionGTE(Integer major, Integer minor)
boolean #util.isDatabaseVersionGT(Integer major)
boolean #util.isDatabaseVersionGT(Integer major, Integer minor)
boolean #util.isDatabaseType(String type)
boolean #util.isNotDatabaseType(String type)
```

The following example shows the use of conditions that are processed during connect of the database connection.

```

<Command id="sybase-ase.getLogins">
  <If test="#util.isDatabaseVersionLTE(5)">
    <SQL>
      <![CDATA[
select name from master.dbo.syslogins
      ]]>
    </SQL>
  </If>
  <ElseIf test="#util.isDatabaseVersionEQ(9)">
    <SQL>
      <![CDATA[
select name, suid from master.dbo.syslogins
      ]]>
    </SQL>
  </ElseIf>
  <Else>
    <SQL>
      <![CDATA[
select name, suid, dbname from master.dbo.syslogins
      ]]>
    </SQL>
  </Else>
</Command>

```

The above means that if the major version of the database being accessed is less than or equal to 5, the first SQL is used. If the major version is equal to 9, the second SQL is used, and the last SQL is used for all other versions. The test attribute may contain conditions that are ANDed or ORed. Conditions can contain multiple evaluations, combined using parenthesis. The If, ElseIf and Else elements may be placed anywhere in the XML file.

Here is another example that controls whether certain nodes will appear in the database objects tree or not.

```

<!-- Getting Table Engines was added in MySQL 4.1 -->
<If test="#util.isDatabaseVersionGTE(4, 1)">
  <GroupNode type="TableEngines" label="Table Engines" isLeaf="true"/>

  <!-- "Errors" was added in MySQL 5 -->
  <If test="#util.isDatabaseVersionGTE(5)">
    <GroupNode type="Errors" label="Errors" isLeaf="true"/>
  </If>
</If>
<Commands>
  <OnError>
    <!-- The ORA-942 error means "the table or view doesn't exist" -->
    <!-- It is caught here since these errors typically indicates -->
    <!-- that the user don't have privileges to access the SYS and/or -->
    <!-- V$ tables. -->
    <If test="#result.getErrorCode() eq 942" context="runtime">
      <Message>
        <![CDATA[
You don't have the required privileges to view this object.
        ]]>
      </Message>
    </If>
    <ElseIf test="#result.getErrorCode() eq 17008" context="runtime">
      <Message>
        <![CDATA[
Your connection with the database server has been interrupted!
Please <a href="connect" action="connect">reconnect</a> to re-establish the connection.
        ]]>
      </Message>
    </ElseIf>
  </OnError>
  ...
</Commands>

```

As you can see, this example contains **nested** uses of If.

## Conditional processing during command execution

Using conditional processing to evaluate any errors from a Command may be useful to rephrase error messages to be more user friendly.

```
<Commands>
  <OnError>
    <!-- The ORA-942 error means "the table or view doesn't exist" -->
    <!-- It is caught here since these errors typically indicates -->
    <!-- that the user don't have privileges to access the SYS and/or -->
    <!-- V$ tables. -->
    <If test="#result.getErrorCode() eq 942" context="runtime">
      <Message>
        <![CDATA[
You don't have the required privileges to view this object.
        ]]>
      </Message>
    </If>
    <ElseIf test="#result.getErrorCode() eq 17008" context="runtime">
      <Message>
        <![CDATA[
Your connection with the database server has been interrupted!
Please <a href="connect" action="connect">reconnect</a> to re-establish the connection.
        ]]>
      </Message>
    </ElseIf>
  </OnError>
  ...
</Commands>
```

The **OnError** element can be used in **Commands** and **Command** elements. If used in **Commands** element, its conditions are processed for all its commands. If it's part of a specific Command, it is processed only for that command.