

# Attributes and named parameters management

Copyright © 2007 Dassault Systemes

### *Abstract*

The attributes are simple properties valued or not, that are intended to be used by events on system components when working with Dysfunctional Analysis & Simulation (SD9). The named parameters are law parameters. They are intended to be used when the user define the law of a component event when working with Dysfunctional Analysis & Simulation (SD9).

# Table of Contents

---

<b>Attributes management .....</b>	<b>4</b>
Main principles .....	4
Displaying the attribute manager window .....	4
Creating an attribute .....	5
Creating a version of an attribute .....	6
Editing an attribute .....	7
Renaming an attribute .....	7
Removing an attribute .....	8
Removing a version of an attribute .....	8
Generating an XML DocBook file report on attributes .....	8
<b>Named parameters management .....</b>	<b>10</b>
Main principles .....	10
Displaying the named parameter manager window .....	10
Creating a named parameter .....	11
Creating a version of a named parameter .....	12
Editing a named parameter .....	13
Renaming a named parameter .....	13
Removing a named parameter .....	14
Removing a version of a named parameter .....	14
Generating an XML DocBook file report on named parameters .....	14

# Attributes management

## Main principles

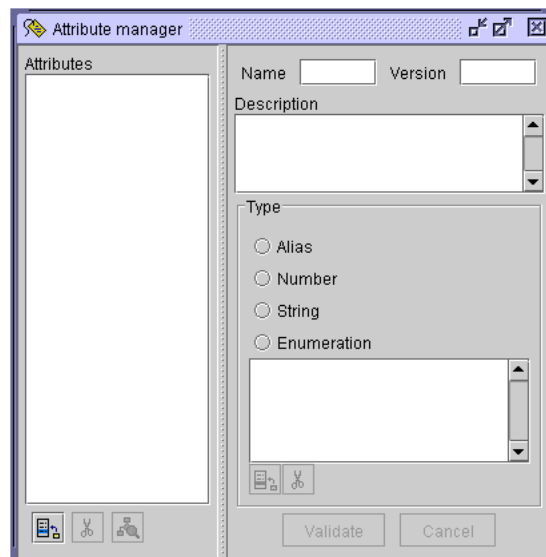
The attribute manager panel allows the user to create attributes that are used to qualify events defined on system components.

## Displaying the attribute manager window

In order to show the attribute manager window, the user clicks on the icon button which follows:



Then the following window appears:



The window is composed of a list on the left part containing the existing attribute names. The right part of the window contains the description of an attribute selected in the left list. If there is no selection in the list, the fields are empty. When an attribute is selected, the fields are filled in with its properties.

The attribute properties are the following ones:

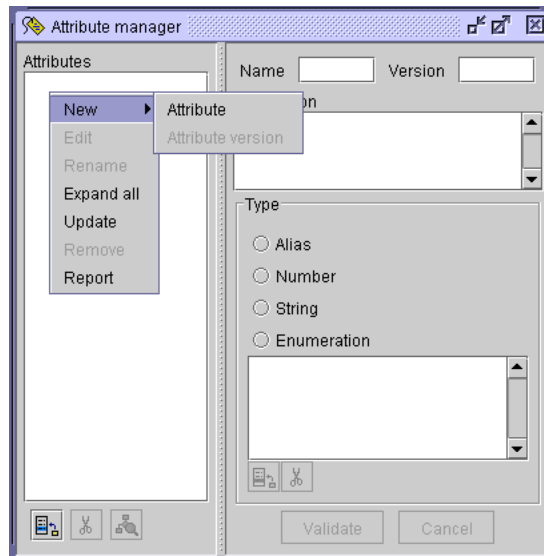
- **Name** of the attribute
- **Version** of the attribute
- **Description** of the attribute
- **Type** of the attribute

The attribute type can be of the following values:

- Alias: the attribute is just a name without value
- Number: the attribute must be associated to a numeric value
- String: the attribute must be associated to a string of characters value
- Enumeration: the attribute must be associated to a value contained in a defined list

In addition to these properties, the attributes of **enumeration** type contains a list of possible values visible in the text area located in the bottom right part of the window.

When the user right clicks on the attribute list, a popup menu appears as follows:



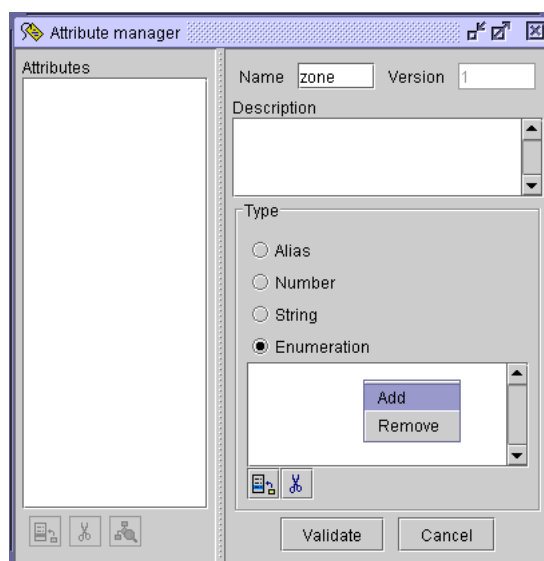
This menu contains all the commands that the user can apply on attributes.

## Creating an attribute

In order to create an attribute, the user proceed as follows.

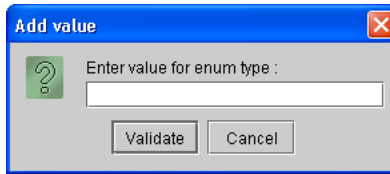
First, he selects the **New -> Attribute** command in the attribute list popup menu. The fields on the right part of the window become editable, except the version. Then the user fill in the name of the attribute, its description and selects the radio button corresponding to its type.

If the attribute is an enumeration, the user can fill in the list of values. To do it, he right clicks in the list located under the **Enumeration** radio button. A popup menu appears as on the figure bellow:

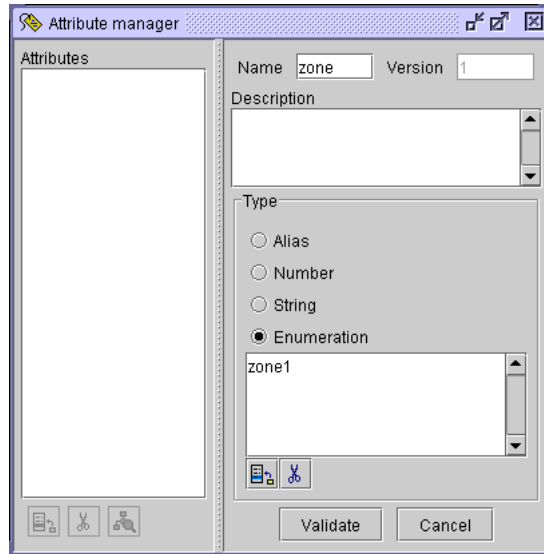


He can then add a value using the **Add** command or remove one using the **Remove** command.

When he wants to add a value, he just has to fill the value in the field of the following dialog box:

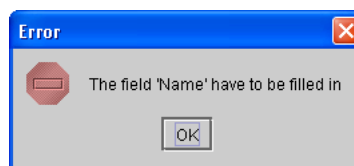


Assuming he add the value 'ZONE1' to the list, the attribute manager window then will appear as follows:

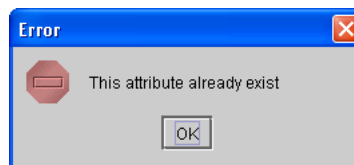


When all the fields are filled in the user clicks on the **Validate** button to confirm the actions or on the **Cancel** button to cancel the actions.

The field **Name** must be filled in, unless the following error message box appears when the user clicks on the **Validate** button:



If the name entered by the user is already used for another attribute, then the following error box is showed when the user clicks on the **Validate** button:

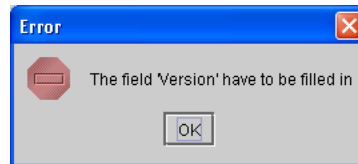


## Creating a version of an attribute

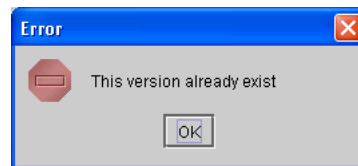
In order to create a version of an attribute, the user proceed as follows.

First he selects an attribute in the left list. Then he selects the **New -> Attribute version** command in the attribute list popup menu. The fields on the right part of the window become editable, except the name. Then the user fill in the version of the attribute, its description and selects the radio button corresponding to its type.

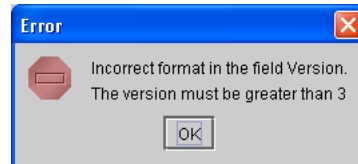
The field **Version** must be filled in, unless the following error message box appears when the user clicks on the **Validate** button:



If the version already exist, the following error message box appears when the user clicks on the **Validate** button:



The version must be greater than the last version, unless the following error message box appears when the user clicks on the **Validate** button:

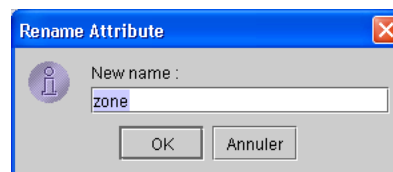


## Editing an attribute

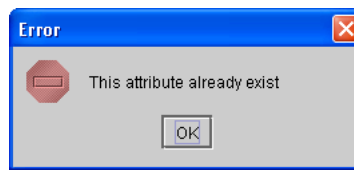
In order to edit an attribute, the user proceed as follows. First he selects a version of an attribute in the left list. Then he selects the **Edit** command in the attribute list popup menu, or he clicks on the icon "Edit". The fields in the right part of the window become editable, except name and version. As for the creation process, the user can fill the fields in and when he has finished, can confirm its modifications or cancel them.

## Renaming an attribute

In order to rename an attribute, the user proceed as follows. First he selects an attribute in the left list. Then he selects the **Rename** command in the attribute list popup menu. The following dialog box appears:

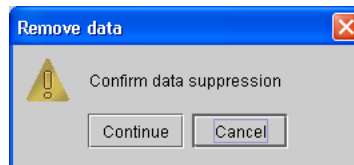


The user enter a new name. If the new name already exist, the following error message box appears when the user clicks on the **OK** button:



## ***Removing an attribute***

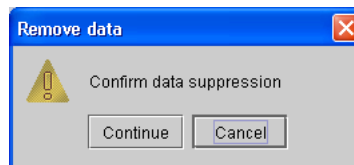
In order to remove an attribute and all its versions, first, he selects an attribute in the left list. Then the user selects the **Remove** command in the attribute list popup menu, or he clicks on the icon "Remove". The following dialog box appears:



If the user confirm the deletion, the attribute and all its version are removed from the list.

## ***Removing a version of an attribute***

In order to remove a version of an attribute, first, he selects a version of an attribute in the left list. Then the user selects the **Remove** command in the attribute list popup menu, or he clicks on the icon "Remove". The following dialog box appears:



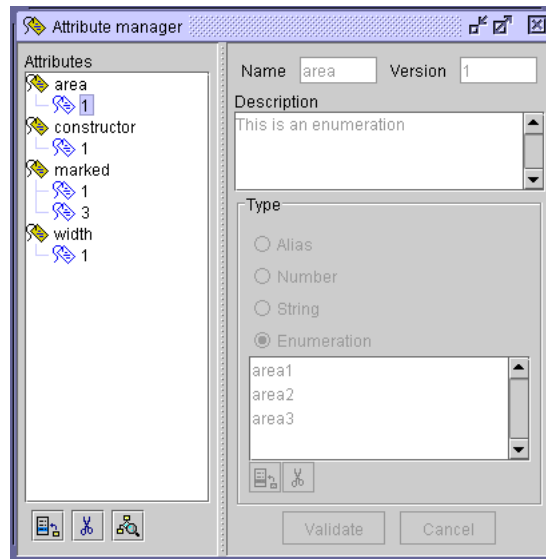
If the user confirm the deletion, the version of the selected attribute is removed from the list.

## ***Generating an XML DocBook file report on attributes***

In order to generate an XML DocBook file report on the attribute list, the user selects the **Report** command in the attribute list popup menu.

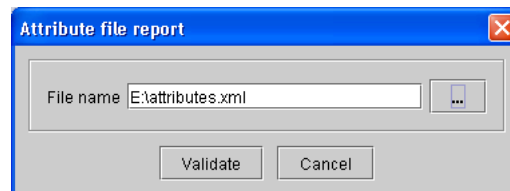


For instance, we assume that the content of the attribute list is as follows:



The attribute list contains four attributes: an alias named 'marked' with 2 versions, a number named 'width', a string named 'constructor' and an enumeration named 'area' with 'area1', 'area2' and 'area3' as possible values.

When the user selects the **Report** command, the following dialog window appears:



The user enters the name of the file in the **File name** field.

At the end, he clicks on the **Validate** button to confirm or on the **Cancel** button to cancel its action.

The XML DocBook file is generated.

# Named parameters management

## Main principles

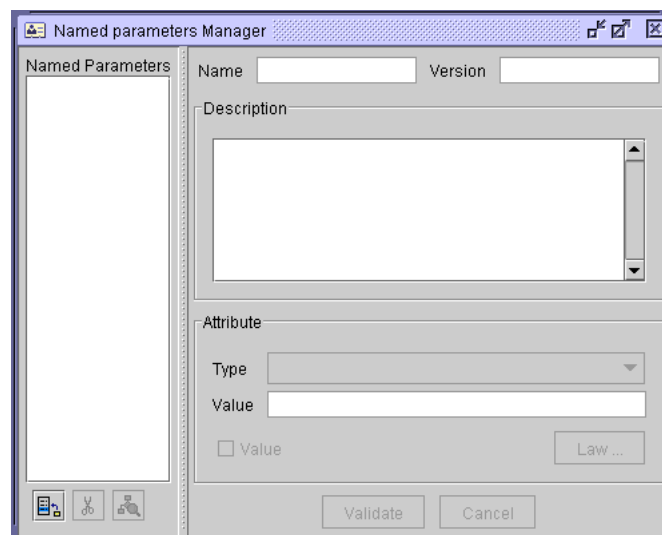
The named parameter manager panel allows the user to create named parameters reusable as many as he wants, on the laws defined on component events. If the value of a named parameter change, all of the laws that use it are automatically impacted.

## Displaying the named parameter manager window

In order to show the named parameter manager window, the user clicks on the icon button which follows:



Then the following window appears:



The window is composed of a list on the left part containing the existing named parameter names. The right part of the window contains the description of a named parameter selected in the left list. If there is no selection in the list, the fields are empty. When a named parameter is selected, the fields are filled in with its properties.

The named parameter properties are the following ones:

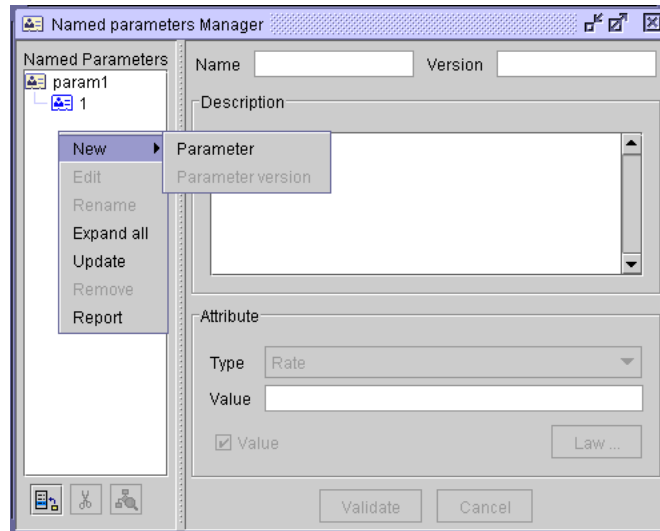
- **Name** of the named parameter
- **Version** of the named parameter
- **Description** of the named parameter
- **Type** of the named parameter
- **Value** of the named parameter

To the named parameter types corresponds a domain of definition for the values. The named parameter type can be of the following values:

- Probability: the value must be between 0 and 1 included
- Rate: the value must be a real greater than or equal to 0
- Duration: the value must be a real strictly greater than 0

- Factor: the value must be a real greater than or equal to 0
- Boolean: the value must be equal to 0 or to 1

When the user right clicks on the named parameter list, a popup menu appears as follows:



This menu contains all the commands that the user can apply on named parameters.

## Creating a named parameter

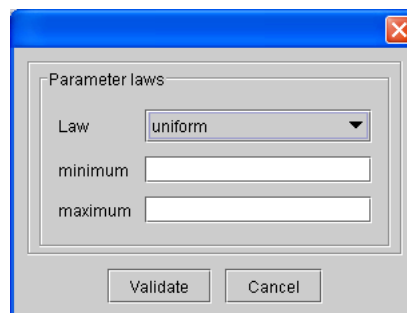
In order to create a named parameter, the user proceed as follows.

First, he selects the **New -> Parameter** command in the named parameter list popup menu. The fields on the right part of the window become editable, except the version. Then the user fill in the name of the named parameter, its description and selects its type among the possible values of the corresponding combo box.

After that, he has to fill the value. The value can take two different forms. The first form is a simple numerical value (example: 0.001, 2.0E-02, etc.). The second form is a uniform or lognormal parameter law (example: uniform 1.000E-03 3.000E-03, lognormal 1.000E-03 3.000E-03, etc.)

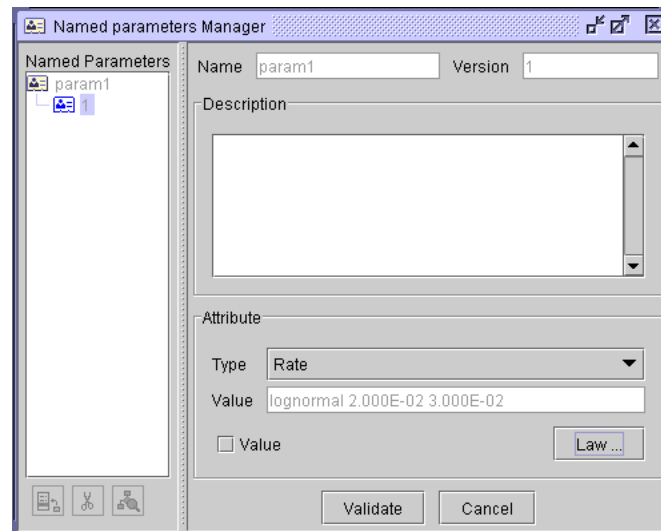
There is two possibilities to specify the parameter value. The first possibility is to enter the value directly into the field. Alternatively, the user can access to a parameter law editor which assist him by clicking on the **Law...** button.

When the user clicks on the **Law...** button, the following dialog window appears:



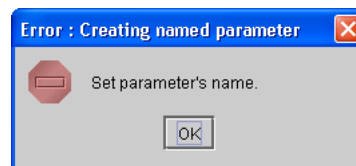
The user selects the law, and its parameters and then clicks on the **Validate** button to confirm the actions or on the **Cancel** button to cancel them.

Assuming that he add the parameter named 'param1' with the value 'lognormal 0.02 0.03' to the list, the named parameter manager window then will appear as follows:

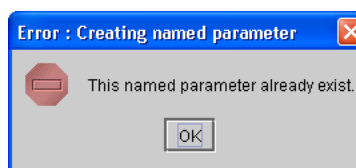


When all the fields are filled in the user clicks on the **Validate** button to confirm the actions or on the **Cancel** button to cancel the actions.

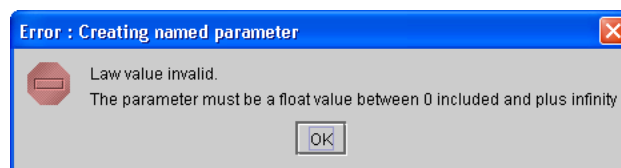
The field **Name** must be filled in, unless the following error message box is showed when the user clicks on the **Validate** button:



If the name entered by the user is already used for another parameter, then the following error box is showed when the user clicks on the **Validate** button:



If the value is not correct, then the following error box is showed when the user clicks on the **Validate** button:



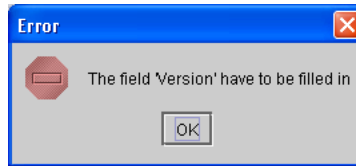
## Creating a version of a named parameter

In order to create a version of a named parameter, the user proceed as follows.

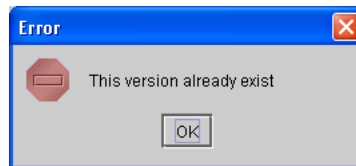
First he selects a parameter in the left list. Then he selects the **New -> Parameter version** command in the named parameter list popup menu. The fields on the right part of the window become editable, except the name. Then he

user fill in the version of the named parameter, its description and selects its type among the possible values of the corresponding combo box.

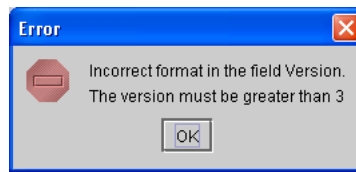
The field **Version** must be filled in, unless the following error message box appears when the user clicks on the **Validate** button:



If the version already exist, the following error message box appears when the user clicks on the **Validate** button:



The version must be greater than the last version, unless the following error message box appears when the user clicks on the **Validate** button:

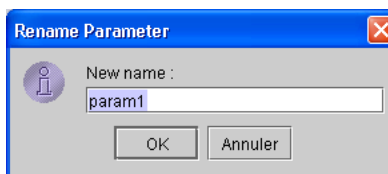


## Editing a named parameter

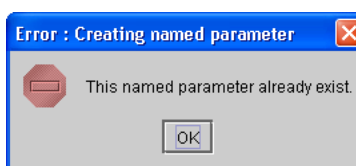
In order to edit a named parameter, the user proceed as follows. First he selects a version of a named parameter in the left list. Then he selects the **Edit** command in the named parameter list popup menu, or he clicks on the icon "Edit". The fields in the right part of the window become editable, except name and version. As for the creation process, the user can fill the fields in and when he has finished, can confirm its modifications or cancel them.

## Renaming a named parameter

In order to rename a named parameter, the user proceed as follows. First he selects a named parameter in the left list. Then he selects the **Rename** command in the named parameter list popup menu. The following dialog box appears:

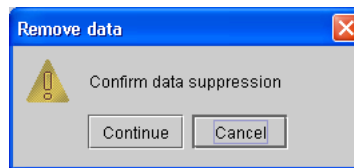


The user enter a new name. If the new name already exist, the following error message box appears when the user clicks on the **OK** button:



## Removing a named parameter

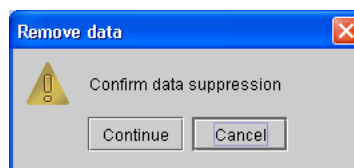
In order to remove a named parameter and all its versions, first, he selects a named parameter in the left list. Then the user selects the **Remove** command in the named parameter list popup menu, or he clicks on the icon "Remove". The following dialog box appears:



If the user confirm the deletion, the named parameter and all its versions are removed from the list.

## Removing a version of a named parameter

In order to remove a version of a named parameter, first, he selects a version of a named parameter in the left list. Then the user selects the **Remove** command in the named parameter list popup menu, or he clicks on the icon "Remove". The following dialog box appears:

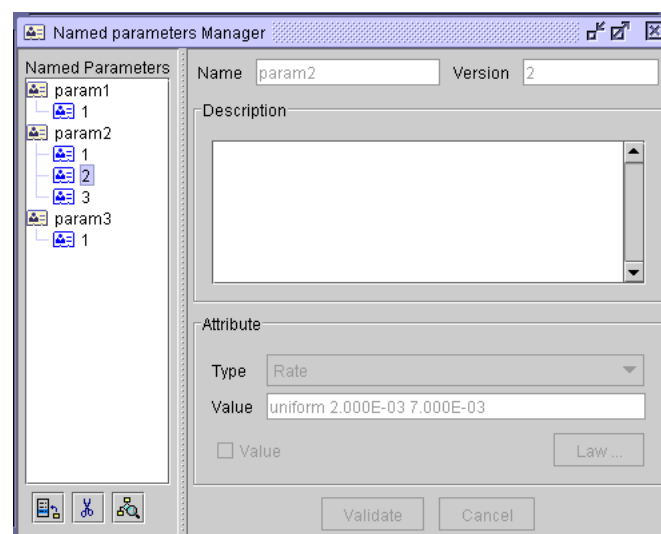


If the user confirm the deletion, the version of the selected named parameter is removed from the list.

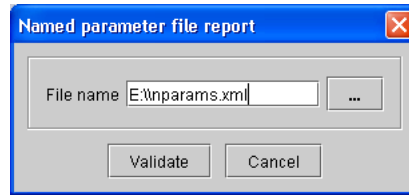
## Generating an XML DocBook file report on named parameters

In order to generate an XML DocBook file report on the named parameter list, the user selects the **Report** command in the named parameter list popup menu.

For instance, we assume that the content of the named parameter list is as follows:



The named parameter list contains the three named parameters. When the user selects the **Report** command, the following dialog window appears:



The user enters the name of the file in the **File name** field.

At the end, he clicks on the **Validate** button to confirm or on the **Cancel** button to cancel its action.

The XML DocBook file is generated.