



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

Genesys Administrator Extension Developer's Guide

Dialog Input Elements

Dialog Input Elements

The `<dialog>` tags define the user-interface elements that are displayed in the wizard to collect user input. A `<dialog>` section can have one or more `<input>` elements to collect user input.

Contents

- 1 Dialog Input Elements
 - 1.1 `<input>`
 - 1.2 `<description>`
 - 1.3 `<objectselect>`
 - 1.4 `<filter>`
 - 1.5 `<selection>`
 - 1.6 `<option>`
 - 1.7 `<validation>`
 - 1.8 `<rule>`

<input>

The <input> tag defines user-interface elements in the wizard to collect user input. One <dialog> tag can have one or more <input> elements to collect user input. It can define different types of input (for example: string, integer, date, boolean, selectOne or selectMultiple configuration objects), and validation rules for these inputs.

Child Elements

- <description>
- <objectselect>
- <selection>
- <validation>

Attributes

Name	Type	Description	Default	Values
default	string	Defines the default value to display in the user interface as the default input value.	—	—
hidden	Boolean	Defines whether to hide the input parameter in the user interface (true == hide, false == show). You can use this attribute in conjunction with a JavaScript expression evaluation to dynamically hide or show an input parameter in a subsequent <dialog>.	false	—
name	string (required)	A unique name for this input parameter. A JavaScript variable with this name is instantiated where the entered value is stored.	—	—
readonly	Boolean	Defines whether the input parameter is read-only or changeable in the user interface (true == readonly, false == changeable). You can use this attribute in conjunction with a JavaScript expression	false	—

Name	Type	Description	Default	Values
		evaluation to dynamically hide or show an input parameter in a subsequent <dialog>.		
required	Boolean	Defines whether an input value is a required parameter. If it is a required parameter, it must be filled out prior to advancing to the next step in the wizard.	false	—
title	string	Defines the title that is displayed in the user interface for this input element.	—	—
type	string	Defines the expected input type and constrains what the user can input.	string	string, integer, boolean, password, selectOne, selectMultiple

<description>

This tag describes what the SPD will do. For example:

```
<description>This is the solution file to deploy a GVP solution</description>
```

The **<input>** tag can be used to provide descriptive information that helps the user to determine what to fill in for the input value.

This element has no attributes or child elements.

<objectselect>

For **<input>** elements with type="selectOne" or type="selectMultiple", this tag can be used to indicate that a configuration object is to be selected. The child **<filter>** element(s) indicate to the user interface which configuration object is to be selected for this **<input>** element.

This element has no attributes.

Child Elements

- **<filter>**

<filter>

Defines a user-interface element to filter the objects displayed in the object picker. Multiple **<filter>** elements can be defined.

This element has no child elements.

Attributes

Name	Type	Description	Default	Values
name	string (required)	Defines the parameter of an to use as a filter. The following parameters are supported: "type", "subtype" and "name".	—	—
value	string (required)	Defines the value for a filter. For example: <pre><filter name="type" value="CfgApplication" /> <filter name="subtype" value="CfgGVPMCP" /> <filter name="name" value="MCP" /></pre>	—	—

<selection>

For **<input>** elements with type="boolean", type="selectOne", or type="selectMultiple" types, the **<selection>** tag can be used to specify a constrained list of input values. The child **<option>** elements can define these allowed input values, or a JavaScript array can be used to define these allowed input values.

Child Elements

- **<option>**

Attributes

Name	Type	Description	Default	Values
array	string	Specifies a JavaScript variable that contains a simple array of values	—	—

Name	Type	Description	Default	Values
		to be used as the allowed input values in the user interface.		
errmsg	string	Specifies an error message to display in the user interface if no value is selected.	—	—
index	string	Specifies the name of a JavaScript variable that contains the index value when the array renders the user interface. This value can be used in the <code><option></code> tag to provide a more intuitive user interface.	—	—

`<option>`

The `<option>` element can be used as a template when an **array** is specified in the `<selection>` element. For example:

```
<selection array="foo" index="i" errmsg="Please select a value from Foo.">
  <option name="${foo[i].firstname+' '+foo[i].lastname}" value="${i}"/>
</selection>
```

Attributes

Name	Type	Description	Default	Values
default	Boolean	Defines whether this option is selected by default in the user interface.	false	—
name	string (required)	Defines the option name to be displayed in the user interface.	—	—
value	string (required)	Defines the value to be assigned to the input parameter JavaScript variable if this option is selected.	—	—

<validation>

For `<input>` elements with `type="string"` or `type="integer"` types, this element can be used to specify validation constraints on the input value.

Child Elements

- `<rule>`

Attributes

Name	Type	Description	Default	Values
default	string	Defines whether by default the input value is accepted or rejected. If <code>default="accept"</code> is specified, then all <code><rule></code> elements must be set to <code>type="reject"</code> . If <code>default="reject"</code> is specified, then all <code><rule></code> elements must be set to <code>type="accept"</code> .	accept	accept, reject
errmsg	string	Specifies an error message to be displayed in the user interface if the value is rejected. This attribute is required if <code>default="reject"</code> is specified.	—	—

<rule>

Defines a validation rule.

This element has no child elements.

Attributes

Name	Type	Description	Default	Values
errmsg	string	Specifies an error message to be displayed in the user interface if the value is rejected. This attribute is required if <code>type="reject"</code> is specified.	—	—

Name	Type	Description	Default	Values
expr	string	Defines a JavaScript expression to be evaluated. It must evaluate to a true (non-falsy) value for the input to be accepted or rejected, depending on the rule type.	—	—
range	string	If specified, then wildcard is ignored. This attribute defines a range of values for the input value to be accepted or rejected, depending on the rule type. For example: range="1, 3, 4-8, >20, <100".	—	—
regex	string	If specified, then range and wildcard are ignored. This attribute defines a regular expression that must be matched for the input value to be accepted or rejected, depending on the rule type.	—	—
type	string	Defines the type of validation rule.	accept	accept, reject
wildcard	string	If specified, then this attribute defines a wildcard string that must be matched for the input value to be accepted or rejected, depending on the rule type. Wildcard characters include "*" and "?". For example: wildcard="foo*", wildcard="b?r", and wildcard="*sg03*".	—	—