

hmc.xml Reference

The `hmc.xml` file specifies the hybris Management Console configuration for About Extensions.

-
- Tags of the hmc.xml File
 - Basic Structure
 - Positioning of Elements
 - <configuration> Tag
 - <action> Tag
 - <defaultattributechip> Tag
 - <attributetype> Tag
 - <display> and <editor> Tags
 - <parameter> Tag
 - <defaultmappings> Tag
 - <defaultmapping> Tag
 - <explovertree> Tag
 - <group> Tag
 - <typeref> Tag
 - <staticcontent> Tag
 - <externalnode> Tag
 - <type> Tag
 - <labels> Tag
 - <label> Tag
 - <defaultreference> Tag
 - <itemlayout> Tag
 - <attribute> AttributeType </attribute> Tag
 - <organizer> Tag
 - <search> Tag
 - <condition> Tag
 - <result> Tag
 - <listview> Tag
 - <itemlayout> Tag
 - <item> Tag
 - <attribute> AttributeDisplayType </attribute> and <attribute> AttributeEditorType </attribute> Tags
 - <display> ClassConfigType </display> and <editor> ClassConfigType </editor> Tags
 - <editview> Tag
 - <treeview> Tag
 - <explorable> Tag
 - <attribute> AttributeExplorableType </attribute> Tag
 - <editor> Tag
 - <exclude> Tag
 - <essentials> Tag
 - <customtab> Tag
 - <tab> Tag
 - <section> Tag
 - <defaultsection> Tag
 - <exclusion> AtomicEditorType </exclusion> Tag
 - hMC Section-Related Tags
 - <columnlayout> Tag
 - <row> Tag
 - <listlayout> Tag
 - <table> Tag
 - <tr> Tag
 - <td> Tag
 - List of Editor Types That Can Be Used in <attribute> Tags
-

About this Document

This document describes the elements of the `hmc.xml` file.

Audience: Technical consultants, developers

Preconditions: Basic understanding of the hybris Commerce Suite's [Build Framework](#)

Related extension: [hmc Extension](#)

Validity: 4.0-RC and higher

Based on hybris version: 3.1-u6

hmc Extension Resources

[Expand all](#) [Collapse all](#)

See Also

- [Configuring the hmc Interface](#)
- [Type System Documentation](#)

Tags of the hmc.xml File

Basic Structure

The basic structure of an `hmc.xml` file is as follows:

```
<configuration xmlns="hybris.de/schemas/hmc/generic"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="hybris.de/schemas/hmc/generic hmc.xsd">

  <action classname="" type="toolbar" />

  <defaultattributechips>
    ...
  </defaultattributechips>

  <defaultmappings>
    ...
  </defaultmappings>

  <explovertree>
    <externalnode classname="" />
    <group name="">
      ...
    </group>
    <typeref type="" />
    <staticcontent />
  </explovertree>

  <type name="">
    <organizer>
      <search>
        <condition attribute="">
          ...
        </condition>
      </search>
    </organizer>
  </type>
</configuration>
```

```
<result>
  <action classname="" type="toolbar">
    ...
  </action>
</result>

<editor>
  <essentials>
    ...
  </essentials>

  <customtab name="" classname="">
    ...
  </customtab>

  <tab name="">
    <defaultsection name="">
      ...
    </defaultsection>

    <section name="">
      ...
    </section>
  </tab>
  <tabref idref="" />
</editor>
</organizer>
</type>
```

```
</configuration>
```

The **hmc.xml** consists of three major parts which correspond to the major parts of the hybris Management Console (hMC):

hmc.xml part	Tag(s)	hMC correspondent	Description
global hMC Actions	<action>	hMC Toolbar Actions	<action> tags may be used multiple times.
Explorer Tree	<exploretree>	Explorer Tree	The <exploretree> tag may be used only once.
Organizer	<type>	Organizer for a hybris Commerce Suite type	The <type> tag may be used multiple times for different types.

As the **hmc.xml** file is validated against the **hmc.xsd** file, the **hmc.xml** file's content must conform to the order given above. A type definition order that does not conform to the **hmc.xsd** will cause the hybris Commerce Suite to fail the extension build with an error message like the following one:

```
[yxmlschemavalidator]
[yxmlschemavalidator] ERROR(S):
[C:\hybris\platform\ext\hmc\web\webroot\WEB-INF\classes\de\hybris\platform\hmc\hmc.xml
]
[yxmlschemavalidator]
[yxmlschemavalidator] line:6225,column:51 : cvc-complex-type.2.4.d: Invalid content
was found starting with element
    'search'. No child element is expected at this point.
[yxmlschemavalidator]
[yxmlschemavalidator] ERROR(S):
[C:\hybris\platform\ext\hmc\web\webroot\WEB-INF\classes\de\hybris\platform\hmc\hmc.xml
]

BUILD FAILED
C:\hybris\platform\build.xml:31: The following error occurred while executing this
line:
C:\hybris\platform\resources\ant\antmacros.xml:368: The following error occurred while
executing this line:
C:\hybris\platform\resources\ant\antmacros.xml:381: The following error occurred while
executing this line:
C:\hybris\platform\resources\ant\util.xml:37: The following error occurred while
executing this line:
C:\hybris\platform\resources\ant\antmacros.xml:383: The following error occurred while
executing this line:
C:\hybris\platform\resources\ant\antmacros.xml:467: The following error occurred while
executing this line:
C:\hybris\platform\resources\ant\util.xml:251: The following error occurred while
executing this line:
C:\hybris\platform\ext\hmc\buildcallbacks.xml:20: The following error occurred while
executing this line:
C:\hybris\platform\resources\ant\antmacros.xml:767: The following error occurred while
executing this line:
C:\hybris\platform\resources\ant\antmacros.xml:791: The following error occurred while
executing this line:
C:\hybris\platform\resources\ant\util.xml:234: null

Total time: 15 seconds
C:\hybris\platform>
```

Positioning of Elements

If you are using the **append** or **replace** inheritance modes, some of the hMC tags allow you to define the position where they should be placed. The respective attribute is called **position** and is optional. The elements are:

- **attribute**
- **externalnode**
- **group**
- **section**
- **staticcontent**
- **tab**
- **text**
- **typeref**

According to the value of **position**, the hMC tries to insert (or replace, respectively) the representation of the element in the specified position. So, if you create a tab and set its position value to 4, the hMC will try to put it as the fourth tab that is available. The following code snippet gives you an example on how to use the position attribute:

```
<tab name="tab.Properties1" mode="append" position="4">
  <!-- the mode attribute in the line above says what inheritance mode is used -->
  <section name="section.strings">
    <listlayout>
      <attribute name="EnginePower" />
    </listlayout>
  </section>
</tab>
```

In that case, the hMC will try to put the **Properties1** tab in the fourth position from the left.

The **position** attribute is only effective during the exact moment of hMC configuration. If another hMC element with the same value for **position** is read after the current one, the current one will be moved to the next position. In the code snippet, **Properties1** would then end up on position 5.

<configuration> Tag

The root tag that includes all others. Its subtags are:

- <action> (ActionType)
- <defaultattribtechips> (DefaultAttributeChipsType)
- <defaultmappings> (ClassMappingsType)
- <explovertree> (ExplorerTreeType)
- <type> (TypeType)

<action> Tag

Configuration of global actions (create, save, and toolbar actions).

Please refer to [hMC Actions](#) to learn more about the various actions and where you need to place them.

Attribute	Description / Comment
classname="string" (mandatory)	Class name (fully qualified) of the action class.
name="string"	Name of the action. Will be used as a key to look up a localized string.
type="string" [possible values: toolbar, item, create, save, remove, searchresult] (mandatory)	The type of the action.
icon="string"	The file name of an icon to use for this action (if applicable).
disabledicon="string"	The file name of a disabled icon to use for this action (if applicable).

toolbaricon="string"	The file name of a toolbar icon to use for this action (if not specified, the action will not be show with the toolbar).
showtoolbarlabel="boolean"	If true, a text will be shown next to the icon in the toolbar. Default is false.
hidebutton="boolean"	Hide the Button in the Editor. Default is false.
autosave="boolean"	For item actions, setting autosave to true will enable the action even when the item is not yet created. When pressing the action button, the item will automatically be saved first. Default is false .
align="string" [possible values: left, right]	The alignment of the action if it is a toolbar action (left, right) - default is left.

Subtags:

- <parameter> (AttributeTypeType): can be used to provide custom parameters to your own action classes

<defaultattributechip> Tag

Configuration of the default attribute editor and display chips.

- <attributetype> (AttributeTypeType)

<attributetype> Tag

Contains descriptions of default editor and/or display chips for a given attribute type.

Attribute	Description / Comment
classname="string" (mandatory)	Class name (fully qualified) of the attribute type for which the default chips are configured.

Subtags:

- <display> (ClassConfigType)
- <editor> (ClassConfigType)

<display> and <editor> Tags

Configuration of the attribute display resp. editor chip which is used for the attributetype (specified in the classname attribute).

Attribute	Description / Comment
classname="string" (mandatory)	Class name (fully qualified) of the chip class this element describes.

Subtags

- <parameter> (ParameterType)

<parameter> Tag

Parameter which will be provided to the class.

Attribute	Description / Comment
key="string" (mandatory)	The name of the parameter's key.
value="string" (mandatory)	The parameter's (string) value.

<defaultmappings> Tag

Configuration to replace core classes with custom classes.

Subtags

- <defaultmapping> ClassMappingType</defaultmapping>

<defaultmapping> Tag

Maps a hybris Management Console core class to a custom class which is to be used instead.

Attribute	Description / Comment
source="string" (mandatory)	Source class name (fully qualified), i.e. the class which should be replaced. Works with the following classes: <ul style="list-style-type: none">• AttributeChip• AttributeEditorChip (and, by consequence, all attribute editors)• ToolbarChip• MainChip• OrganizerListChip• OrganizerSearchChip• OrganizerItemChip• OrganizerItemCreatorChip• GenericItemCreatorChip• DefaultContentChip
target="string"; (mandatory)	Target class name (fully qualified), i.e. the class which is to be used instead of the source class.

<explovertree> Tag

Description of the hMC's Explorer Tree (what nodes are shown etc.).

Attribute	Description / Comment
width="integer"	Width of the explorer tree in pixels.
mode="modeType"	Mode can be one of the following: append , replace , or delete . Default is append .

Subtags

- <group> (GroupType)
- <typeref> (TypeReferenceType)
- <staticcontent> (StaticContentNodeType)
- <externalnode> (ExternalNodeType)

<group> Tag

Adds a group node (i.e. a container or folder like 'Catalog' or 'System') to the explorer tree which in turn can contain other nodes.

Attribute	Description / Comment
name="string" (mandatory)	The name of the group node. This value will be used for localization reference. For example, the <group name="cmsgroup" description="group.cms.description"> definition will cause the hMC to use the localization string cmsgroup to display the entry itself and the localization string group.cms.description as the description of the entry. Please also refer to the documentation on localization for details.
mode="modeType"	Mode can be one of append , replace or delete . Default is append .
expand="string"	If set to true the node will already be expanded after a login into the hybris Management Console.
position="integer"	The position of this group within the explorer tree.

Subtags

- <group> (GroupType)
- <typeref> (TypeReferenceType)
- <staticcontent> (StaticContentNodeType)
- <externalnode> (ExternalNodeType)

<typeref> Tag

Adds a node which references a type's Organizer to the explorer tree. In other words, this points to the hMC configuration for a type Organizer.

Attribute	Description / Comment
type="string" (mandatory)	The code of the type. The appropriate type description is used to determine how the organizer is built.
position="string"	The position of this node in the group.
mode="modeType "	Mode can be one of append , replace , or delete . Default is append .

<staticcontent> Tag

Can be used to add a .JSP file as a node (while not using a chip class or representing a type).

Attribute	Description / Comment
name="string" (mandatory)	The name of the node. This name will be used as a key to look up a localized string.
uri="string" (mandatory)	The URI of the jsp to be used.
icon="string"	The url of the icon to be used for the node.
position="string"	The position of this node in the group.

<externalnode> Tag

Adds a given chip class as a node which does not represent a type .

Attribute	Description / Comment
classname="string" (mandatory)	The (fully qualified) name of the chip class to be used.
position="string"	The position of this node in the group.

<type> Tag

Description of a single type (appearance in the editor, in lists, etc.).

Attribute	Description / Comment
name="string" (mandatory)	The code of the type which is described here.
icon="string"	Pathname of an icon which should be used for this type.
useselector="boolean"	If useselector is set to true, attribute editors of this type will always show a select box to choose an item of this type (see for example the 'unit' attribute in a 'product' item).
mode="modeType"	Mode can be one of append , replace , or delete . Default is replace .

Subtags

- <organizer> (OrganizerType)
- <defaultreference> (DefaultReferenceType)
- <labels> (LabelCollectionType)

<labels> Tag

Contains labels.

Subtags

- <label> LabelType </label>

<label> Tag

A label.

Attribute	Description / Comment
name="string" (mandatory)	The name of the label "create" -> the type will be shown in the global "create" menu of the hmc. "wizard" -> the type will be indirectly created using a wizard, which in turn can be provided using <parameter key="wizardtype" value="[type]"/>.

Subtags

- <parameter> ParameterType </parameter>

<defaultreference> Tag

Describes how items of this type look (by default) when they are referenced in a different location.

Attribute	Description / Comment
searchattribute="string"	Defines which attribute is used for reference searches (e.g. in the autocomplete reference editor).
showempty="boolean"	If true , empty attributes will be shown in the reference (e.g. like xyz - empty - 123) instead of simply omitting the appropriate attribute.
mode="modeType"	Mode can be one of append , replace , or delete . Default is replace .

Subtags

- <itemlayout> (AttributeCollectionType)

<itemlayout> Tag

Lists a number of attributes which are shown when items of this type are referenced in other editors.

Attribute	Description / Comment
mode="modeType "	Mode can be one of append , replace , or delete . Default is append .

Subtags

- <attribute> (AttributeType)

<attribute> AttributeType </attribute> Tag

Describes one attribute in this list.

Attribute	Description / Comment
name="string"	The qualifier of the attribute.
mode="modeType "	Mode can be one of append , replace , or delete . Default is append .
position="integer"	The position of the attribute within an <itemlayout>.

<organizer> Tag

Contains descriptions on how the different organizer components represent this type.

Attribute	Description / Comment
mode="modeType "	Mode can be one of append , replace , or delete . Default is append .

Subtags

- <search> (SearchType)
- <result> (ResultType)

- <editor> (EditorType)

<search> Tag

Contains configurations of the organizer search section.

Attribute	Description / Comment
mode="modeType "	Mode can be one of append , replace or delete . Default is append .
autopperform="boolean"	Specifies whether the search is triggered automatically on opening the organizer in the hMC. If set to true , the search is triggered automatically; if set to false (default), then the user needs to click on the Search button to trigger the search.

Subtags

- <condition> (ConditionType)

<condition> Tag

Describes a default condition which will be added to the search section.

Attribute	Description / Comment
attribute="string" (mandatory)	Name of the attribute for which this condition applies.
operator="string" [possible values are: equal, empty, is not null, is null, like, starts with, ends with, contains, >, <, yes, no]	Operator to use in this condition.
defaultvalue="string"	Default value for this condition's operator.
mode="modeType "	Mode can be one of append , replace , or delete . Default is append .
position="integer"	The position of this condition within the search.

Subtags

- <editor> (ClassConfigType)

<result> Tag

Contains configurations of the organizer result section.

Attribute	Description / Comment
defaultview="string" [possible values: list, edit, tree]	Sets the default view mode (list, edit, tree) for the organizer result section. If no default view is explicitly set, the list view will be used.
mode="modeType "	Mode can be one of append , replace , or delete . Default is append .

Subtags

- <listview> (ListViewType)
- <editview> (EditViewType)
- <treeview> (TreeViewType)
- <action> (ActionType)

<listview> Tag

Configuration of the organizer result section when in list view mode.

Attribute	Description / Comment
mode="modeType "	Mode can be one of append , replace , or delete . Default is append .

scrollbar="boolean"	Can be true or false (false to hide the scrollbar). Default is true .
---------------------	--

Subtags

- `<itemlayout>` (ItemDisplayLayoutType)

Although `<listview>`, `<editview>`, and `<treeview>` all have an `<itemlayout>` subtag, the subtag is not the same across all three of them. `<listview>` and `<treeview>` include an `itemdisplaylayout` type, whereas `<editview>` includes an `itemeditorlayout` type. We will still put all the subtags into one single explanation and mention any differences explicitly.

<itemlayout> Tag

Defines which attributes of the type are shown in the result list (one attribute per column).

Attribute	Description / Comment
mode="modeType "	Mode can be one of append , replace , or delete . Default is append .

Subtags

- `<attribute>` (AttributeDisplayType if nested in a `<display>` tag)
- `<attribute>` (AttributeEditorType if nested in an `<editor>` tag)
- `<item>` (ItemType)

<item> Tag

With this tag, you can show a reference of the respective item within a reference or search result list.

The `<item>` tags' attributes

Attribute	Description / Comment
width="string"	The width of this element in pixels.
title="string"	Alternative title of list columns (instead of the attribute name, for example). Will be used as a key to lookup a localized string.

<attribute> AttributeDisplayType </attribute> and <attribute> AttributeEditorType </attribute> Tags

Configure how to display resp. editor attribute.

The `<attribute> AttributeDisplayType </attribute>` and `<attribute> AttributeEditorType </attribute>` tags' attributes

Attribute	Description / Comment
name="string" (mandatory)	The name of this tab. The name will be used as a key to look up a localized string.
mode="modeType "	Mode can be one of append , replace , or delete . Default is append .
position="integer"	The position of this condition within the search.
width="string"	The width of this element in pixels.
jsp="string"	The url of a jsp file which is to be used for this attribute (only relevant for display chips!).
title="string"	Alternative title of list columns (instead of the attribute name, for example). Will be used as a key to lookup a localized string.

Subtags

- `<display>` (ClassConfigType if nested in a `<listview>` or `<treeview>` tag)
- `<editor>` (ClassConfigType when nested in an `<editview>` tag)

<display> ClassConfigType </display> and <editor> ClassConfigType </editor> Tags

Set the display resp. editor class to be used for this attribute.

The `<display>` `ClassConfigType` `</display>` and `<editor>` `ClassConfigType` `</editor>` tags' attributes

Attribute	Description / Comment
classname="string" (mandatory)	The fully qualified classname of the respective editor or display chip to use. For editor chips the class needs to extend <code>AbstractAttributeEditorChip</code> . Display chips need to extend <code>AbstractAttributeDisplayChip</code> .

Subtags

- `<parameter>` (`ParameterType`)

`<editview>` Tag

Configuration of the organizer result section when in `EditView` mode.

Attribute	Description / Comment
mode="modeType "	Mode can be one of append , replace , or delete . Default is append .
scrollbar="boolean"	'false' to disable the scrollbar in the search result list. Default is 'true'.

Subtags

- `<itemlayout>` (`ItemEditorLayoutType`)

Although `<listview>`, `<editview>`, and `<treeview>` all have an `<itemlayout>` subtag, the subtag is not the same across all three of them. `<listview>` and `<treeview>` include an `itemdisplaylayout` type, whereas `<editview>` includes an `itemeditorlayout` type. We will still put all the subtags into one single explanation and mention any differences explicitly.

`<treeview>` Tag

Configuration of the organizer result section when in `TreeView` mode.

Attribute	Description / Comment
mode="modeType "	Mode can be one of append , replace , or delete . Default is append .
scrollbar="boolean"	'false' to disable the scrollbar in the search result list. Default is 'true'.

Subtags

- `<itemlayout>` (`ItemDisplayLayoutType`)

Although `<listview>`, `<editview>`, and `<treeview>` all have an `<itemlayout>` subtag, the subtag is not the same across all three of them. `<listview>` and `<treeview>` include an `itemdisplaylayout` type, whereas `<editview>` includes an `itemeditorlayout` type. We will still put all the subtags into one single explanation and mention any differences explicitly.

- `<explorable>` (`ExplorableType`)

`<explorable>` Tag

Define which attributes are explorable, i.e. which attributes are shown and explorable below the item nodes.

Attribute	Description / Comment
mode="modeType "	Mode can be one of append , replace , or delete . Default is append .

Subtags

- `<attribute>` (`AttributeExplorableType`)

`<attribute>` `AttributeExplorableType` `</attribute>` Tag

Attribute which should be explorable, i.e. which will be shown below an item node in the tree view result list.

Attribute	Description / Comment
name="string" (mandatory)	The attribute's qualifier
mode="modeType"	Mode can be one of append , replace , or delete . Default is append .
suppressroot="boolean" [0..1]	If set to true , the appropriate attribute itself will not appear as a node, but all its explorable subnodes will be shown directly (i.e. one node level will be skipped).

<editor> Tag

Contains configurations of the organizer editor section.

Attribute	Description / Comment
mode="modeType "	Mode can be one of append , replace , or delete . Default is append .

Subtags

- <exclude> (ExcludeType)
- <essentials> (EssentialsType)
- <tab> (TabType)
- <customtab> (CustomTabType)
- <tabref> (TabRefType)

<exclude> Tag

Contains the attributes which should be excluded (omitted) in the editor of this type. Be sure you do not explicitly add these attributes in any of the other sections!

If there are attributes which have not been explicitly configured to appear on any tab/section inside the editor and you do not 'exclude' them, then they will appear inside the so-called 'defaultsection' (which is by default located on the 'administration' tab).

Attribute	Description / Comment
mode="modeType "	Mode can be one of append , replace , or delete . Default is append .

Subtags

- <exclusion> (AtomicEditorType)

<essentials> Tag

Contains the attributes which are "essential" for this type. These attributes will be shown on top of every tab in the editor.

Attribute	Description / Comment
mode="modeType "	Mode can be one of append , replace , or delete . Default is append .

Subtags

- <table> (TableType)
- <listlayout> (ListLayoutType, see also the <listlayout> tag)
- <columnlayout> (ColumnLayoutType)

<customtab> Tag

Unlike the <tab> tag which is generically filled with sections, attribute editors etc, the <customtab> tag allows you to write your own tab from scratch.

In our core hmc implementation, the "Access rights" tab (in a ComposedType editor) is an example of a custom tab. Its functionality is quite unique compared to any "normal" tab so that it had to be completely customized.

You will have to provide your own chip class with its own jsp and thus you are able to do almost anything you like on this tab. You only have to implement the "ClipChip" interface, but this is more like a marker interface since you can implement the methods as 'empty' if you like.

You also have the option to extend our "normal" tab (GenericTabEditorChip) to inherit its functionality (so you can still use <section>s, for example, and then add some custom things).

Attribute	Mandatory	Description / Comment
name="string"	yes	The name of this custom tab. The name will be used as a key to look up a localized string.
classname="string"	yes	The fully qualified classname of the attribute editor to use (needs to implement ClipChip!).
mode="string"	no	Mode can be one of append , replace , or delete . Default is append .
position="integer"	no	The position of this tab within the tabs.

Subtags

- <section> SectionType </section> - Is only effective if your customtab extends GenericTabEditorChip. And then it has the same meaning as in the <tab> element.
- <defaultsection> DefaultSectionType</defaultsection> - Is only effective if your customtab extends GenericTabEditorChip. And then it has the same meaning as in the <tab> element.

<tab> Tag

Describes one tab of the editor, which in turn contains sections and attributes.

Attribute	Description / Comment
name="string" (mandatory)	The name of this tab. The name will be used as a key to look up a localized string.
mode="modeType "	Mode can be one of append , replace , or delete . Default is append .
position="integer"	The position of this tab within the tabs.

Subtags

- <section> SectionType </section>
- <defaultsection> DefaultSectionType</defaultsection>

<section> Tag

Adds a section to the tab.

Attribute	Description / Comment
name="string" (mandatory)	The name of this tab. The name will be used as a key to look up a localized string.
mode="modeType "	Mode can be one of append , replace , or delete . Default is append .
position="integer"	The position of this section within the tab.

Subtags

- <table> (TableType)
- <listlayout> (ListLayoutType)
- <columnlayout> (ColumnLayoutType)

<defaultsection> Tag

Adds a default section to the tab, which causes every attribute which is not explicitly specified elsewhere in this editor (also referred to as **unbound**) to be displayed in this section (see the <exclude> tag on information how to exclude an attribute from an item editor entirely).

Attribute	Description / Comment
name="string" (mandatory)	The name of this tab. The name will be used as a key to look up a localized string.
mode="modeType "	Mode can be one of append , replace , or delete . Default is append .
preferred="boolean"	If true this section should be preferred over possible other default sections within this editor.

Subtags

- `<exclusion>` (`AtomicEditorType`)

`<exclusion> AtomicEditorType </exclusion>` Tag

Attribute which should be excluded from the enclosing default section or from the item editor (if it is placed inside the `<exclude>` tag). Thus you are able to exclude single attributes completely from an editor if you do not configure it anywhere else.

Attribute	Description / Comment
<code>attribute="string"</code>	Name (qualifier) of the type's attribute.

hMC Section-Related Tags

`<columnlayout>` Tag

Configure the layout of the type's attributes using a layout with two columns.

Attribute	Description / Comment
<code>width="string"</code>	The width of this element in pixels.
<code>height="string"</code>	The height of this element in pixels.
<code>leftwidth="positiveInteger"</code>	Explicitly set the width of the left column.
<code>rightwidth="positiveInteger"</code>	Explicitly set the width of the right column.

Subtags

- `<row>` (`RowType`)

`<row>` Tag

Describes one row of a column layout (containing up to 2 elements for a 2-column layout).

The contained elements (e.g. attribute editors) are rendered from left to right. If you only provide one element in a row then it will be rendered as if you use the HTML attribute `'colspan=2'`.

If you want to explicitly have one element to fill up only the left or the right column (without any spanning) then please use the `<void>` element for the respective other column.

Attribute	Description / Comment
<code>width="string"</code>	The width of this element in pixels.
<code>height="string"</code>	The height of this element in pixels.
<code>mode="modeType "</code>	Mode can be one of append , replace or delete . Default is append .
<code>position="integer"</code>	The position of this row within the columlayout.

Subtags (most are equivalent to, for example, the `<td>` subtags)

- `<attribute>` (`AttributeChipType`)
- `<customattributechip>` (`CustomAttributeChipType`)
- `<customchip>` (`CustomChipType`)
- `<inlineshortcut>` (`InlineShortcutType`)
- `<shortcut>` (`ShortcutType`)
- `<action>` (`ActionType`) - (item action)
- `<text>` (`TextChipType`)
- `<void>` (`VoidType`) - can be used to fill up one column to prevent spanning of a single element inside a row

`<listlayout>` Tag

Configure the layout of the type's attributes using a list layout (all containing elements, e.g. attribute editors, are rendered top to bottom).

Attribute	Description / Comment
width="string"	The width of this element in pixels.
height="string"	The height of this element in pixels.
mode="modeType "	Mode can be one of append , replace , or delete . Default is append .

Subtags

- <attribute> (AttributeChipType)
- <customattributechip> (CustomAttributeChipType)
- <customchip> (CustomChipType)
- <table> (TableType)
- <listlayout> (ListLayoutType)
- <inlineshortcut> (InlineShortcutType)
- <shortcut> (ShortcutType)
- <action> (ActionType)
- <text> (TextChipType)

<table> Tag

Configure the layout of the type's attributes using a table layout. This layout mimics the html table element, so you can use a few well known features of that element.

Attribute	Description / Comment
width="string"	The width of this element in pixels.
height="string"	The height of this element in pixels.

Subtags

- <tr> (TRType)

<tr> Tag

Describes one row of an editor table (works just like the HTML tr element).

Attribute	Description / Comment
width="string"	The width of this element in pixels.
height="string"	The height of this element in pixels.

Subtags

- <td> (TDType)

<td> Tag

Describes the content of one table cell of an editor table (works just like the HTML td element).

Attribute	Description / Comment
width="string"	The width of this element in pixels.
height="string"	The height of this element in pixels.
colspan="positiveInteger"	Just like in an html table this configures the column spanning of a td element.
rowspan="positiveInteger"	Just like in an html table this configures the row spanning of a td element.

Subtags

- <attribute> (AttributeChipType)
- <customattributechip> (CustomAttributeChipType)
- <customchip> (CustomChipType)

- <table> (TableType)
- <listlayout> (ListLayoutType)
- <inlineshortcut> (InlineShortcutType)
- <shortcut> (ShortcutType)
- <action> (ActionType)
- <text> (TextChipType)

List of Editor Types That Can Be Used in <attribute> Tags

The following list comprises all editor chips that are available to the platform by default.

Attribute	Description / Comment
name="string"	Name (qualifier) of the type's attribute.
mode="modeType "	Mode can be one of append , replace or delete . Default is append .
width="positiveInteger"	Width of the attribute editor.
labelwidth="nonNegativeInteger"	Width of the attribute label.
maxlength="positiveInteger"	With maxlength you can restrict the entry length in some text editors (e.g. in stringeditor and textareaeditor).
hidelabel="boolean"	If true the label of the attribute will not be shown.
descriptionattribute="string"	Name of description attribute.
editable="boolean"	By setting this to 'false' the respective attribute will not be editable (regardless of the current user's access rights). Default is 'true'.
position="integer"	The position of this attribute within the editor.

Subtags

- <defaulteditor> (AtomicEditorType)
- <customeditor> (CustomEditorType)
- <stringeditor> (StringEditorType)
- <stringselecteditor> (AtomicEditorType)
- <textareaeditor> (TextAreaEditorType)
- <wysiwygeditor> (WysiwygEditorType)
- <integereditor> (LongEditorType)
- <shorteditor> (LongEditorType)
- <longeditor> (LongEditorType)
- <doubleeditor> (AtomicEditorType)
- <floateditor> (AtomicEditorType)
- <booleaneditor> (AtomicEditorType)
- <advancedbooleaneditor> (AdvancedBooleanEditorType)
- <dateeditor> (AtomicEditorType)
- <datetimeeditor> (DateTimeEditorType)
- <longasdateeditor> (AtomicEditorType)
- <longasdatetimeeditor> (AtomicEditorType)
- <javaclasseditor> (AtomicEditorType)
- <enumerationvalueselecteditor> (EnumerationValueSelectEditorType)
- <referenceeditor> (ReferenceEditorType)
- <partofreferenceeditor> (PartOfReferenceEditorType)
- <referencecollectioneditor> (ReferenceCollectionEditorType)
- <collectioneditor> (CollectionEditorType)
- <subtypeselecteditor> (SubTypeSelectEditorType)
- <specialeditor> (SpecialEditorType)
- <mediaeditor> (MediaEditorType)
- <mapeditor> (MapEditorType)
- <specialcreator> (SpecialCreatorType)