

Concept - Better inheritance with extends

Abstract

Introduce new property value for **extends=remove** to completely turn off inheritance or parts of it.

Current situation

Currently extending the configuration in Magnolia can be done the following way:

- extends=/path/to/extended/config/node

One can override parts of an inherited configuration node by using

- extends=override

See also:

[Extends of Configurations](#)

<http://documentation.magnolia-cms.com/templates/stk/template-definitions.html#Extendsproperty>

<http://documentation.magnolia-cms.com/reference/configuration.html#Extendingconfiguration>

"Misuse" of extends=override

In Magnolia 4.5 extends=override was often used to "deactivate" inheritance on subnodes.

In Magnolia 5, if used like this, the Node2BeanTransformer will provide the default implementation from the type-mapping, which may cause errors in the new UI.

Example

Example of stkPromo component in standard templating kit:

Node name	Value	Type	Status	Modification date
standard-templating-kit	-	-	●	-
config	-	-	●	-
renderers	-	-	●	-
trees	-	-	●	-
virtualURIMapping	-	-	●	-
templates	-	-	●	-
dialogs	-	-	●	-
pages	-	-	●	-
generic	-	-	●	-
functional	-	-	●	-
components	-	-	●	-
pur	-	-	●	-
promos	-	-	●	-
stkPromo	-	-	●	Nov 17, 2008 12:20 PM
form	-	-	●	-
tabs	-	-	●	-
tabTeaser	-	-	●	Oct 16, 2008 7:51 PM
fields	-	-	●	-
inheritable	-	-	●	Nov 7, 2008 3:49 PM
extends	/modules/standard-templating-kit/dialogs/generic/controls/inheritable	String	-	-
hideTeaserImage	-	-	●	-
extends	override	String	-	-
extends	/modules/standard-templating-kit/dialogs/generic/master/baseTeaserInte	String	-	-
content	-	-	●	-
stages	-	-	●	-
features	-	-	●	-
teasers	-	-	●	-
links	-	-	●	-
extras	-	-	●	-
version	2.5.0-SNAPSHOT	String	-	-
public-user-registration	-	-	●	-

In this example stkPromo will inherit from /modules/standard-templating-kit/dialogs/generic/master/baseTeaserInternal while adding a new field "inheritable" to the tab "tabTeaser" and overriding the field "hideTeaserImage". Overriding the field "hideTeaserImage" will result in the Node2BeanTransformer using the default ConfiguredFieldDefinition (defined in magnolia-ui-form: /META-INF/magnolia/ui-form.xml). Thus causing

RegistrationException: Could not find fieldType for definition info.magnolia.ui.form.field.definition.
ConfiguredFieldDefinition

as there is no FieldBuilder available for that definition. See:

Error rendering macro 'jira'

Unable to locate Jira server for this macro. It may be due to Application Link configuration.

Something similar happens, when extending actions and overriding parts of them:

 Unable to locate Jira server for this macro. It may be due to Application Link configuration.

Proposal

We should introduce another value for the extends property

- extends=remove

to fully remove subnodes from the resulting node (after inheritance).

"Normal" use of extends should be handled as is, but simply overriding configuration nodes should not result in errors that make the UI unusable. Therefore some of the default type-mapping would have to be adjusted, e.g. in the above stkPromo example, extends=override should result in a functioning dialog: the type-mapping might be adjusted to TextFieldDefinition or ConfiguredFieldDefinition should be supplied with working defaults.

We should fix existing bootstrap files with

- a groovy script, which updates nodes with only one property (extends=override)

and

- create a migration task, which changes the respective property (extends=override), if only that one exists.