

Consolidate package names for various i18n-related features

Relates to [MAGNOLIA-5284](#) - Define new package name(s) CLOSED

We have 3 distinct i18n "areas" in Magnolia:

- * System: used for translations of Magnolia itself
- * Content: localization of content (a site in multiple languages)
- * Authoring: provides authors with means to translate their sites (relies on i18n-content)

Additionally, templates currently use the "system" components for some additional translations that are currently not handled by authors (such as "Read more" links)

Currently, the following packages exist:

- * `org.magnolia.cms.i18n` (main)
- * `org.magnolia.ui.api.i18n` (ui)
- * `org.magnolia.ui.framework.i18n` (ui)
- * `org.magnolia.ui.vaadin.gwt.client.editor.i18n` (ui)
- * `org.magnolia.cms.gui.i18n` (legacy-admininterface)

The `org.magnolia.cms.i18n` package contains components for system and content. It also contains, i.e. `Content`-API based wrappers. Additionally, `org.magnolia.jcr.wrapper.i18n.NodeWrapper` is, well, in the wrong package; it should be packaged along the i18n-content features.

After an initial short discussion with [~pbaerfuss], we thought about having these 3 packages:

- * `org.magnolia.i18n.system`
- * `org.magnolia.i18n.content`
- * `org.magnolia.i18n.authoring`

... but as shown above, we probably want to keep the differentiations between api, framework, Would `org.magnolia.i18n.authoring.api` and `org.magnolia.i18n.authoring.framework` work? If we want `org.magnolia.ui` as a parent to both, I'd favor dropping the `org` and using package names in a "single word", such as `org.magnolia.i18ncontent`.

TODO : use template below...

- > The goal of this template is to remind you of many aspects that you could consider, to help you create a good concept.
- > Tip: Use whichever sections are appropriate for the concept, delete the rest, rename and add at will, these are simply a suggestion.
- > Tip: Delete all these comments.

Purpose

The Problem

- > AKA (Also known as): Starting Position
- > Why is this concept being considered? What is the issue that someone is having that needs to be addressed?

Goals

- > This feature will be a success if it achieves these criteria.
- > Tests could be defined based on these goals.

Use Cases

- > AKA: User Stories
- > List concrete ways in which this feature will be used.

Proposal

Concept

- > AKA: Design
- > What is the overall approach to solve the problem. First summarize the high level overview in a few lines.

Reasoning

- > AKA: Rationale
- > Pros and Cons
- > Consequences of this approach.

Implementation

- > Describe concrete details of how the feature will be implemented. Could include class diagrams or interfaces.
- > Requirements
- > Thoughts
- > Open Questions

Etcetera

- > Possible Improvements
- > Discarded Proposals
- > Discussion
- > References