

I18n keys - with strikethroughs

- + New suggested key
- 🚫 Key suggested for deprecation
- ✓ old key which is consistent and doesn't require changes
- ? To be discussed

Apps

- + "apps.<app>.label",
- 🚫 "<app>.app.label"

App launcher group

+ "appLauncherLayout.groups.<group>.label", //you can immediately identify in an i18n file that it's a group name (compared to the old keys below), moreover *appLauncherLayout* is the name of the app launcher configuration node

- 🚫 "app-launcher.<group>.label",
- 🚫 "app-launcher.<group>"

Actionbar

Sections

//new suggested keys respect the path in configuration:

- + "apps.<app>.subApps.<subapp>.actionbar.sections.<section>.label",
- + "subApps.<subapp>.actionbar.sections.<section>.label",
- + "actionbar.sections.<section>.label",

vs

- 🚫 "<app>.<subapp>.actionbar.sections.<section>.label",
- 🚫 "<app>.<subapp>.actionbar.sections.<section>.label",
- 🚫 "<app>.<subapp>.actionbar.sections.<section>.label",
- 🚫 "<app>.<subapp>.actionbar.sections.<section>.label",

Actions

New suggested keys:

- + "apps.<app>.subApps.browser.actions.<action>.label",
- + "subApps.browser.actions.<action>.label",
- ✓ "actions.<action>.label",

Currently supported keys:

- ? "<app>.browser.actions.<action>.label", //short key, doesn't respect configuration path but is shorter

Columns

New suggested keys:

- ? "apps.<app>.subApps.<subapp>.workbench.contentViews.<view-name>.columns.<column-name>.label",
- ? "subApps.<subapp>.workbench.contentViews.<view-name>.columns.<column-name>.label",
- ? "workbench.contentViews.<view-name>.columns.<column-name>.label",

✔ "columns.<column-name>.label", //we could probably support this one, if no objections

vs currently supported keys:

? "<app>.<subapp>.views.<view-type>.<column-name>.label",

? "<app>:<subapp>.views.<view-type>.<column-name>.label",

? "<app>.<subapp>.views.<view-type>.<column-name>.label",

? "<app>:<subapp>.views.<view-type>.<column-name>.label",

? "<app>:<subapp>.views.<view-type>.<column-name>.label",

? "<app>:<subapp>.views.<view-type><column-name>.label",

Pretty hard to remember, isn't it? 🤔 Also notice <view-type>. This is tree/list and it doesn't matter how is the view called in config although all the other parts are always the node names in JCR (yaml).

Forms

Dialog

New suggested keys:

+ "<module>.dialogs.<dialogPath>.label",

+ "dialogs.<dialogPath>.label", //this is "must have". We currently cannot use module-independent translations

? "dialogs.testDialog.label",

//currently supported keys:

👉 "<module>.<dialogPath>.label",

Tabs

New suggested keys:

+ "<module>.dialogs.<dialogPath>.form.tabs.<tab>.label",

+ "dialogs.<dialogPath>.form.tabs.<tab>.label",

+ "form.tabs.<tab>.label",

//currently supported keys:

? "<module>.<dialogPath>.<tab>.label", Do we still want to have this one? It's shorter but not a big difference and it doesn't respect configuration path.

👉 "<tab>.label", //That's pretty dangerous since it's too short and without a type prefix. How you can possibly know where it belongs by looking in the translation file? Is it for template/dialog/app?

Fields

New suggested keys:

+ "<module>.dialogs.<dialogPath>.form.tabs.<tab>.fields.<field>.label",

? "dialogs.<dialogPath>.form.tabs.<tab>.fields.<field>.label",

? "form.tabs.<tab>.fields.<field>.label",

✔ "fields.<field>.label" //we might want to have generic translation for all fields with this name (fallback)


//currently supported keys:

? "<module>.<dialogPath>.<tab>.<field>.label", //vs "<module>.dialogs.<dialogPath>.form.tabs.<tab>.fields.<field>.label",

? "<module>.<dialogPath>:<tab>.<field>.label",


? "<module>:<dialog>.<tab>.<field>.label",

? "<module>:<dialog>:<tab>.<field>.label", //vs "form.tabs.<tab>.fields.<field>.label",


 "<module>:<dialog>:<tab>.<field>.label",

Validators

New suggested keys:

 <parentFieldId18nKey>validators.<validator>.errorMessage //respects configuration path

//currently supported keys:

 <parentFieldId18nKey>validation.errorMessage //no information about validator, just field definition