Internationalization

Internationalization Guide

About

This will allow any piece of static text in the application to be modified dynamically.

Usage

n the UI code, there are three options for converting the text provided into translated text.

- Component: `<translate-i18n key="action.foo">Foo</translate-i18n>`
- Directive: `<label for="foo" translate="label.foo">Foo</label>`
- Pipe: `{{ action.foo | translate }}`

In addition, the `application.yml` file is an example of a file where dynamic values from the server, can be populated with an i18n identifier for translation. The json files used to generate the metadata wizards also make use of these identifiers and can be translated. They all use the same `messages.properties` files.

Here is an example of the identifers and text found in the `messages.properties` files.

```
action.cancel=Cancel
action.save=Save
action.delete-type=Delete { type }
```

The `action.delete-type=Delete { type }` example shows how dynamic text can be inserted into the translation as well. These values are determined by the application and are populated at runtime based on user interaction. Many of these values can be translated separately.

Conventions

Choosing an identifier

To allow for easier identification of relevant strings the identifiers are provided in the following format

```
type.kebab-case-text
```

For example

```
<button>example text</button>
<!-- would become -->
<button translate="action.example-text">example text</button>
```

Where `type` is one of the provided types listed in the types section

`type` and `text` are separated by two dashes.

`text` is the text content, dash separated words, all lower case, with punctuation removed.

Types

The following types are provided:

- `action` buttons or links that cause a state change within the app
- `label` label for an input or a section
- `message` messages that warn a user of exceptions in interactions, i.e. validation messages
- `value` text for displaying values such as `true` or `false`
- `branding.*` this special type is used to denote the customizable values located in the `ui > brand.ts` file

Localize Text Only

```
<!-- BAD -->
<!-- this will pull both the text AND the icon into the translation -->
<button i18n="action.some-text">
  some text
  <i class="fa fa-icon"></i>
</span>

<!-- GOOD -->
<!-- this will only localize the text -->
<!-- NOTE: ng-container does not create any new dom on the page -->
<button>
  <translate key="action.some-text">some text</ng-container>
  <i class="fa fa-icon"></i>
</button>
```

Updating Text

When updating text, update the identifier to match the new text field.

```
<!-- BAD -->
<!-- identifier does not match text after update -->
<label translate="label.name">Address</label>

<!-- GOOD -->
<!-- identifier and text match -->
<label translate="label.address">Address</label>
```