This document shows the various ways how to set a property of a document to translatable allowing to supply translations for this property as well as how to declare a translation domain.

${table_of_content}

Set a Property to Translatable

Go to [your_erp5_instance]/portal_property_sheets/ and select a portal type. Pick any of the properties.

Check the **Translatable** property. This will create three new accesors methods: getTranslatedProperty, getPropertyTranslationDomain and setPropertyTranslationDomain, which you can call like:

```
context.getTranslated[propertyName]()
```

which will return a value based on the current localizer language.

Alternatively you can set the properties in the Zope Interface aswell or via a Python script (discouraged), such as:

```
_properties = (
  {
    'id'          : 'title',
    'description' : '',
    'type'        : 'string',
    'default'     : '',
    'mode'        : 'w',
    'translatable': 1,
  },
```

Set a Property to Translation Domain

There are different ways to define the translation domain for a property.

Using the translation tab (or zope interface)

As above go to your portal type property sheet, select the translation tab and set the translation domain for the property.

By default *erp5_ui* (user interface), *erp5_content* (content) and *default* should be available as options.

Set via content_translation translation domain (requires erp5_content_translation BT5)

If you set *content_translation* to translation_domain, then the *content_translation* method is enabled on the property. It allows to store translation data in the document itself instead of the Localizer's message catalog. This means, every single document can have different translations per language.

In addition, it also automatically creates special translation accessors for languages enabled in Localizer. If for example, Portuguese/Brazil (pt-BR) and French (fr) are enabled on Localizer and *content_translation* is set to **translation_domain** on the Person portal_type, the following methods are created:

```
Person.getPtBrTranslatedTitle
Person.setPtBrTranslatedTitle
Person.getFrTranslatedTitle
Person.setFrTranslatedTitle
```

If you want to enable a special UI for these content translation accessors including search features, install erp5_content_translation business template. This will enable a Translation tab on the portal type documents where you can enter the translation text per language.

Scripting into the property sheet (discouraged)

Just add the attribute translation_domain with the domain value for the property. For example:

```
_properties = (
    {
        'id'         : 'title',
        'description' : '',
        'type'       : 'string',
        'default'    : '',
        'mode'       : 'w',
        'translatable': 1,
        'translation_domain': 'erp5_content',
    },
```

Overridding with a python script (discouraged)

Create a script named PortalType_getPropertyTranslationDomain, and return the domain as a string.

Manually setting (discouraged)

You can use the accessor setPropertyTranslationDomain in a script to define it on the object

How is the translation domain looked up?

When a translation is required for a property, ERP5 tries to get the translation domain. Here is the order how this look up is perfomed:

- Try to get the property from the object directly
- If a script like PortalType_getPropertyTranslationDomain is defined, use it
- Use the domain defined in the translation tab if there is one
- Use the domain defined in the property sheet if there is one
- If no domain is defined, no translation is performed

${related_subject_list}