

# Table of Contents

Sometimes it's important to translate records of a table/databook. If you have such a requirement, you can use different solutions:

- Use a view and translate column values via database
- Translate values on the client (without database)

If you choose the first option, everything will be fine because it'll be possible to use filtering and searching via database. With the second option, only client-side filtering and searching will work. *Why?*

If you translate, for example, English content to German on the client side, the value in the database is still English. If the user tries to search a German value, the database won't find it because all values are in English.

The second option is easier to use, and the first one needs some database/view logic. The second option will load/fetch all records because of sorting and filtering. If you have only few records, or searching and filtering is not important, the second option will be perfect.

### How Client Translation Works

StringDataType has the following methods:

```
public setTranslator(ITranslator pTranslator)
public ITranslator getTranslator()
```

If you set the translator for a column with a StringDataType, every value will be translated automatically. The data type has the method:

```
public Object prepareValue(Object pObject)
```

which is responsible for the translation.

### What is an ITranslator?

We have default implementations for the interface. Simply use a TranslationMap, an UIComponent, or an instance of IControl. Usually it's not needed to implement your own translator.

### Example

The [UIEnumCellEditor](#) automatically translates the display values. Simply check the source code or use the following snippet:

```
StringDataType sdtName = new StringDataType();
sdtName.setTranslator(getApplication().getLauncher().getTranslation());

MemDataBook mdbData = new MemDataBook();
mdbData.getRowDefinition().addColumnDefinition(new
ColumnDefinition("SYSTEMNAME"));
```

```
mdbData.getRowDefinition().addColumnDefinition(new ColumnDefinition("NAME",
stdName));
mdbData.getRowDefinition().addColumnDefinition(new
ColumnDefinition("NAME_INTERN"));
mdbData.setMemFilter(true);
mdbData.setMemSort(true);

//also possible
StringDataType sdtSysName = (StringDataType)mdbData.getRowDefinition()
.getColumnDefinition("SYSTEMNAME").getDataType();
sdtSysName.setTranslator(getApplication().getLauncher().getTranslation());
```

From:

<https://doc.sibvisions.com/> - **Documentation**

Permanent link:

<https://doc.sibvisions.com/jvx/client/model/databook/recordtranslation>



Last update: **2020/07/08 17:40**