openCRX Feature Acceleration: DataBindings

Version 1.11.0

www.opencrx.org

13-Apr-2007 @ 01:30:30 PM

Table of Contents

List of Figures

List of Listings

1 About the Feature

Data bindings for **PropertySet** allow to add **user-defined fields** to **CrxObject** (and hence to virtually any openCRX object) as follows:

1. add the field as feature definition to the ui customizing
2. customize the new feature in ui customizing

Example:

```
<featureDefinition>
  <org.openmdx.ui1.FeatureDefinition qualifiedName="org:opencrx:kernel:account1:Contact:Custom!custom1">
    <_object>
      <type>org:w3c:string</type>
      <multiplicity>0..1</multiplicity>
      <changeable>true</changeable>
    </_object>
    <_content/>
  </org.openmdx.ui1.FeatureDefinition>
</featureDefinition>
<elementDefinition>
  <org.openmdx.ui1.ElementDefinition name="org:opencrx:kernel:account1:Contact:Custom!custom1">
    <_object>
      <active>true</active>
      <toolTip>
        <_item>Custom field 1</_item>
      </toolTip>
      <label>
        <_item>Custom 1</_item>
      </label>
      <dataBindingName>org.opencrx.kernel.gui.generic.servlet.StringPropertyDataBinding</dataBindingName>
    </_object>
    <_content/>
  </org.openmdx.ui1.ElementDefinition>
</elementDefinition>
```

Advantages of this new feature:

- **There is no limitation regarding the number of user-defined attributes** whereas user-definable attributes (userXXX) modeled into the class **CrxObject** are limited to 4 single-valued attributes and 1 multi-valued attribute of each type.

- **Attributes can be added to (removed from) classes without the need to change the model** and hence it is sufficient to be familiar with the customization of openCRX in order to "extend" a class (i.e. it is not necessary to be familiar with UML modeling, MOF restrictions, etc.).

- **Attributes can be added to (removed from) classes without the need to change the DB schema**; all the required tables already exist in the standard distribution of openCRX.

- **Attributes can be added to (removed from) classes without the need to recompile/rebuild openCRX**; it is sufficient to deploy the new/changed ui customization files.

- **Attributes can be customized exactly like regular openCRX attributes** (including placement at will in respective Inspectors).

2 Implementation

The following data binding classes for **Property** will be implemented:

- BooleanPropertyDataBinding
- DecimalPropertyDataBinding
- IntegerPropertyDataBinding
- DatePropertyDataBinding
- DateTimePropertyDataBinding
- StringPropertyDataBinding

In combination with (the already existing feature) domain reference it is also possible to define code value attributes.

In order to make the mapping ui element ←→ object feature customizable, an extension of openMDX/portal is also required (currently, the mapping is 1:1 and feature values are accessed in a hard-coded way using refGetValue(), refSetValue(), ...):

Data bindings will allow to

- intercept the feature access with busines logic ;
- access features of referenced or composite objects
- ...

Data bindings are an extension point of **PortalExtension_1_0**. The DefaultDataBinding accesses features using the RefObject_1_0 operations.