openbiz-cubi

Rapid PHP Application Development Platform

Project Home   Downloads   **Wiki**   Issues   Source   Export to GitHub

Search | Current pages ▼ | for | | Search

OpenbizFrameworkMetadata
describe Openbiz metadata concept.
Phase-Implementation

Openbiz Metadata

Steps to Build Openbiz Applications

Openbiz is a metadata centric framework, so the application development process is some different with the traditional one.
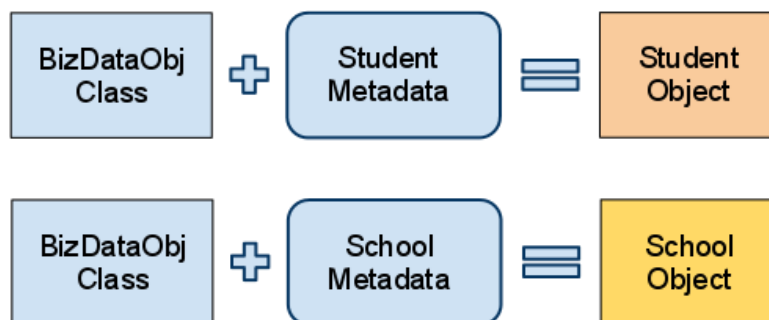
- Step 1: gather requirements
- Step 2: design data models, e.g. database schema
- Step 3: write business objects with DO and their metadata
- Step 4: write user interface with Form and View
- Step 5: write custom DO, Form and Service classes if necessary
- Step 6: refine the metadata and custom code
- Step 7: test and debug

Step 3 and 4 are all about editing XML metadata. You can edit the metadata with your favorite xml editor or text editor. Openbiz Cubi has tools (scripts) to help generating and editing metadata.

Manage Metadata

Introduction of Metadata

The core concept of the Openbiz is its metadata-driven mechanism. What is metadata? From the dictionary, metadata is a component of data which describes the data. It is "data about data". Metadata files in Openbiz are actually the configuration files of Openbiz classes. All Openbiz core classes are general classes. They represent different things with association to different metadata. For example, when StudentDO.xml links to BizDataObj class, this BizDataObj instance is a student object. While when SchoolDO.xml links to BizDataObj class, then this BizDataObj instance becomes a school object.

Because Openbiz classes are described with metadata, authoring metadata files is same as implementing a class. Thus, application development means authoring metadata files in most time, instead of traditional programming. Application described with the metadata files should have more clear logic and design.

What can metadata do:

- Describe the properties of objects
- Describe relationship of objects
- Describe rendering behavior of objects
- Describe validation of the data
- Describe user interaction on a page

What can't metadata do:

- Logic of function - this is implemented in real program classes. The "Class" attribute of a metadata can bind any custom class with the metadata.

Sample metadatas

Let's show two simple metadata samples - UserDO.xml represents a user data object and UserNewForm.xml represents a new event form. The the meaning the xml should be self-explained.

DO metadata

UserDO.xml (this file can be found at in Cubi under /modules/system/do/).

```
<?xml version="1.0" standalone="no"?>
<BizDataObj Name="UserDO" class="BizDataObj" DBName="Default" Table="user"
SearchRule="" SortRule="" OtherSQLRule="" IdGeneration="Identity">
    <BizFieldList>
        <BizField Name="Id" Column="id"/>
        <BizField Name="username" Column="username"/>
        <BizField Name="password" Column="password"/>
        <BizField Name="email" Column="email"/>
...
```

Form metadata

UserNewForm.xml (this file can be found at in Cubi under /modules/system/form/).

```
<?xml version="1.0" encoding="UTF-8"?>
<EasyForm Name="UserNewForm" Class="UserForm" FormType="New" Title="New user" Description="Please fill the form below to c
BizDataObj="system.do.UserDO" PageSize="10" TemplateEngine="Smarty" TemplateFile="system_right_detailform.tpl.html" Message
    <DataPanel>
        <Element Name="fld_Id" Class="LabelText" FieldName="Id" Label="Id" Hidden="Y" />
        <Element Name="fld_username" Class="InputText" FieldName="username" required="Y" Label="Username"  Validator="{@val
        <Element Name="fld_email" Class="InputText" FieldName="email" Label="Email" Validator="{@validate:email('[fld_email
...
```

Manage metadata with package

A good designed applications are usually built upon modules. Openbiz recommends developers to create their metadata under app/modules/ directory - this is what Openbiz Cubi does. Openbiz metadata files can be organized by module name and sub directory names. It is like the package concept used in Java.

For example,

```
PackageX.PackageY.metaA.xml refers to the metaA.xml under modules/PackageA/PackageB directory.
```

Metadata Simple Expression

In order to adding flexibility of metadata configuration, Openbiz simple expression is used in metadata files. If a statement has {expr} pattern, expr will be evaluated as an expression. Basically, an expression is a single PHP statement which returns a value. If users need more complicated logic which can't be put in a simple expression, they can associate an object with user-defined class where they put special code.

Expression Tags

Three expression tags are supported.

- {expr} pairs. Openbiz will do php eval on the expr string between { and }
- {fx}expr{/fx} pairs. This is the verbose version of {} pairs. Openbiz will do php eval on the expr string between {fx} and {/fx}. Example, {fx}10-1{/fx} returns "9".
- {tx}expr{/tx} pairs. This pair tells Openbiz simply returns the strings without calling eval. Example, {tx}10-1{/tx} returns "10-1".

Using Expressions

Simple expression is to support dynamic value binding of metadata attributes. Users can use simple expression in the following place in metadata files.

BizDataObj

SearchRule, SortRule, !OtherSQLRule, AccessRule, UpdateCondition,! DeleteCondition.

BizField

Required, Validator, Value, DefaultValue

EasyForm

Title

Element

Link, Style, Hidden, Enabled, SelectFrom

EventHandler

Function, RedirectPage

Literals

The simple expression language defines the following literals:

- Boolean: true and false
- Integer: as in PHP
- Floating point: as in PHP
- String: with single and double quotes; " is escaped as \", ' is escaped as \', and \ is escaped as \\.

- Null: null

Operators

The simple expression language provides the following operators:

- Arithmetic: +, - (binary), *, / and div, % and mod, - (unary)
- Logical: and, &&, or, ||, not, !
- Relational: ==, !=, <, >, <=, >=. Comparisons can be made against other values, or against boolean, string, integer, or floating point literals.
- Conditional: A ? B : C. Evaluate B or C, depending on the result of the evaluation of A.

Variables

Simple expression allows developers to use variables of openbiz metadata objects.

Syntax to get metadata object variables	Description	Sample usage
`@object_name:property_name`	get the given property of the given object.	`@EventDO:Name, @EventForm:Title`
`@object_name:*[child_name].property_name`	get the given property of the given object's child element	`@EventDO:Field[Id].Value, @EventForm:Elem[evt_id].Value`
`@:property_name or @this:property_name`	get the given property of this object ("this" object is the object defined in the metadata file)	In EventDO, `@:Name` or `@this:Name` means getting the "Name" property of EventDO.
`@:*[child_name].property_name or @this:*[child_name].property_name`	get the given property of this object's child element	In EventDO, `@:Field[Id].Value` or `@this:Field[Id].Value` means getting the "Id" field value of EventDO.
`[field_name]`	get the value of a given Field of its DO or Element of its Form	In EventDO, `[Id]` means getting the "Id" field value of EventDO.
`@profile:property_name`	get the user profile property. User profile is provided with ProfileService.	`@profile:ROLEID`
`@svcname:method(arg1, arg2 ...)`	invoke the registered plugin service method and get the returned value. Currently registered plugin services are @validation - validation service @query - query service. To register a service, $g_ServiceAlias can be defined as a global variable.	In a LabelText Element, text=" {@query:FetchField(easy.d_Event, [Id]=@:Element[fld_evtid].Value, Name)}"`

As implied from the implementation, developers can add more property support by modifying/overriding GetProperty() method. The input of GetProperty() can be either "property_name" or "*[child_name]" or something new that supported by customer code.

- Simple expression language also allows developers to use any global variables supported by PHP. Please read http://us2.php.net/manual/en/reserved.variables.php for details

Functions

Developers can invoke any PHP functions in simple expression. A user defined functions can be invoked if the file that contains such function is included. For example, if the metadata object A is based on a customer class, the class file is A.php that includes another A_help.inc. In this case, you can invoke functions defined in A_help.inc in simple expression.

Examples

```
<BizDataObj SearchRule="[Start]>'date(\"Y-m-d\")'">

<BizDataObj AccessRule="[OwnerId]='{@profile:USERID}'">

<BizDataObj UpdateCondition="[OrgId]=={@profile:ORGID}">

<BizDataObj DeleteCondition="'admin'=={@profile:ROLEID}">

<BizField Name="NeedApprove" Required="{[Amount]>1500}"/>

<BizField Name="Fee" Validator="{[Fee]>=15}"/>

<BizField Name="FullName" Value="{[LastName]}, {@:Field[FirstName].Value}"/>

<Element Name="fld_evtname" Class="LabelText" FieldName="" Label="Event Name"
text="{@query:FetchField(easy.d_Event, [Id]=@:Element[fld_evtid].Value, Name)}"/>
```

Openbiz Metadata DTD

Data Object

```
<!--OpenBiz BizDataObj metadata DTD-->

<!ELEMENT BizDataObj (BizFieldList, TableJoins, ObjRefernces, Parameters) >
<!ATTLIST BizDataObj    Name                       CDATA  #REQUIRED >
```

```
<!ATTLIST BizDataObj    Description              CDATA  #REQUIRED >
<!ATTLIST BizDataObj    Class                    CDATA  #REQUIRED >
<!ATTLIST BizDataObj    InheritFrom              CDATA  #IMPLIED >
<!ATTLIST BizDataObj    DBName                   CDATA  #IMPLIED >
<!ATTLIST BizDataObj    Table                    CDATA  #REQUIRED >
<!ATTLIST BizDataObj    IdGeneration             CDATA  #REQUIRED >
<!ATTLIST BizDataObj    SearchRule               CDATA  #IMPLIED >
<!ATTLIST BizDataObj    SortRule                 CDATA  #IMPLIED >
<!ATTLIST BizDataObj    OtherSQLRule             CDATA  #IMPLIED >
<!ATTLIST BizDataObj    AccessRule               CDATA  #IMPLIED >
<!ATTLIST BizDataObj    CreateCondition          CDATA  #IMPLIED >
<!ATTLIST BizDataObj    UpdateCondition          CDATA  #IMPLIED >
<!ATTLIST BizDataObj    DeleteCondition          CDATA  #IMPLIED >
<!ATTLIST BizDataObj    CacheLifetime            CDATA  #IMPLIED >
<!ATTLIST BizDataObj    Access                   CDATA  #IMPLIED >
<!ATTLIST BizDataObj    MessageFile              CDATA  #IMPLIED >

<!ELEMENT BizFieldList (BizField+) >
<!ELEMENT BizField EMPTY >
<!ATTLIST BizField    Name                CDATA  #REQUIRED >
<!ATTLIST BizField    Class               CDATA  #IMPLIED >
<!ATTLIST BizField    Join                CDATA  #IMPLIED >
<!ATTLIST BizField    Column              CDATA  #REQUIRED >
<!ATTLIST BizField    SQLExpr             CDATA  #IMPLIED >
<!ATTLIST BizField    Type                CDATA  #IMPLIED >
<!ATTLIST BizField    Format              CDATA  #IMPLIED >
<!ATTLIST BizField    Required            CDATA  #IMPLIED >
<!ATTLIST BizField    Validator           CDATA  #IMPLIED >
<!ATTLIST BizField    DefaulValue         CDATA  #IMPLIED >
<!ATTLIST BizField    Value               CDATA  #IMPLIED >
<!ATTLIST BizField    OnAudit             CDATA  #IMPLIED >

<!ELEMENT TableJoins (Join+) >
<!ELEMENT Join EMPTY >
<!ATTLIST Join    Name          CDATA  #REQUIRED >
<!ATTLIST Join    Table         CDATA  #REQUIRED >
<!ATTLIST Join    Column        CDATA  #REQUIRED >
<!ATTLIST Join    JoinRef       CDATA  #IMPLIED >
<!ATTLIST Join    ColumnRef     CDATA  #REQUIRED >
<!ATTLIST Join    JoinType      CDATA  #REQUIRED >

<!ELEMENT ObjReferences (Object+) >
<!ELEMENT Object  EMPTY >
<!ATTLIST Object  Name          CDATA  #REQUIRED >
<!ATTLIST Object  Description   CDATA  #IMPLIED >
<!ATTLIST Object  Relationship  CDATA  #REQUIRED >
<!ATTLIST Object  Table         CDATA  #REQUIRED >
<!ATTLIST Object  Column        CDATA  #REQUIRED >
<!ATTLIST Object  FieldRef      CDATA  #REQUIRED >
<!ATTLIST Object  OnDelete      CDATA  #IMPLIED >
<!ATTLIST Object  OnUpdate      CDATA  #IMPLIED >
<!ATTLIST Object  XDataObj      CDATA  #IMPLIED >
<!ATTLIST Object  XTable        CDATA  #IMPLIED >
<!ATTLIST Object  XColumn1      CDATA  #IMPLIED >
<!ATTLIST Object  XColumn2      CDATA  #IMPLIED >
```

View

```
<!--OpenBiz EasyView metadata DTD-->

<!ELEMENT EasyView (FormReferences) >
<!ATTLIST EasyView    Name            CDATA  #REQUIRED >
<!ATTLIST EasyView    Description     CDATA  #REQUIRED >
<!ATTLIST EasyView    Class           CDATA  #IMPLIED >
<!ATTLIST EasyView    TemplateEngine  CDATA  #IMPLIED >
<!ATTLIST EasyView    TemplateFile    CDATA  #IMPLIED >
<!ATTLIST EasyView    Access          CDATA  #IMPLIED >

<!ELEMENT FormReferences (Reference+) >
<!ELEMENT Reference EMPTY >
<!ATTLIST Reference   Name            CDATA  #REQUIRED >
<!ATTLIST Reference   SubForm         CDATA  #IMPLIED >
```

Form

```
<!--OpenBiz EasyForm metadata DTD-->
<!ELEMENT EasyForm (SearchPanel, DataPanel, ActionPanel, NavPanel) >
<!ATTLIST EasyForm    Name            CDATA  #REQUIRED >
<!ATTLIST EasyForm    Description     CDATA  #REQUIRED >
<!ATTLIST EasyForm    Class           CDATA  #REQUIRED >
<!ATTLIST EasyForm    jsClass         CDATA  #REQUIRED >
<!ATTLIST EasyForm    Title           CDATA  #REQUIRED >
<!ATTLIST EasyForm    BizDataObj      CDATA  #REQUIRED >
<!ATTLIST EasyForm    PageSize        CDATA  #REQUIRED >
<!ATTLIST EasyForm    SearchRule      CDATA  #IMPLIED >
<!ATTLIST EasyView    TemplateEngine  CDATA  #IMPLIED >
<!ATTLIST EasyView    TemplateFile    CDATA  #IMPLIED >
<!ATTLIST EasyView    Access          CDATA  #IMPLIED >
<!ATTLIST EasyVuew    MessageFile     CDATA  #IMPLIED >

<!ELEMENT SearchPanel (Element+) >
<!ELEMENT DatePanel (Element+) >
<!ELEMENT ActionPanel (Element+) >
<!ELEMENT NaviPanel (Element+) >

<!ELEMENT Element (EventHandler+) >
<!ATTLIST Element    Name                CDATA  #REQUIRED >
```

```
<!ATTLIST Element     Class           CDATA  #IMPLIED >
<!ATTLIST Element     FieldName       CDATA  #IMPLIED >
<!ATTLIST Element     Label           CDATA  #IMPLIED >
<!ATTLIST Element     Type            CDATA  #IMPLIED >
<!ATTLIST Element     Width           CDATA  #IMPLIED >
<!ATTLIST Element     Height          CDATA  #IMPLIED >
<!ATTLIST Element     HTMLAttr        CDATA  #IMPLIED >
<!ATTLIST Element     Link            CDATA  #IMPLIED >
<!ATTLIST Element     Image           CDATA  #IMPLIED >
<!ATTLIST Element     Hidden          CDATA  #IMPLIED >
<!ATTLIST Element     Enabled         CDATA  #IMPLIED >
<!ATTLIST Element     Sortable        (Y|N) "" >
<!ATTLIST Element     Style           CDATA  #IMPLIED >
<!ATTLIST Element     ValuePicker     CDATA  #IMPLIED >
<!ATTLIST Element     Pickermap       CDATA  #IMPLIED >
<!ATTLIST Element     SelectFrom      CDATA  #IMPLIED >

<!ELEMENT EventHandler EMPTY >
<!ATTLIST EventHandler Name           CDATA  #REQUIRED >
<!ATTLIST EventHandler Event          CDATA  #REQUIRED >
<!ATTLIST EventHandler Function       CDATA  #REQUIRED >
<!ATTLIST EventHandler FunctionType   CDATA  #IMPLIED >
<!ATTLIST EventHandler ShortcutKey    CDATA  #IMPLIED >
<!ATTLIST EventHandler RedirectPage   CDATA  #IMPLIED >
<!ATTLIST EventHandler consoleMenu    CDATA  #IMPLIED >
```

Service

```
<!--OpenBiz Plugin service metadata DTD-->

<!ELEMENT PluginService ANY >
<!ATTLIST PluginService Name           CDATA  #REQUIRED >
<!ATTLIST PluginService Description    CDATA  #IMPLIED >
<!ATTLIST PluginService Class          CDATA  #REQUIRED >
```

► Sign in to add a comment