Tutorial

How to create a Java class from an XML file

Sommaire

Presentation of the application:

Version 2.1/A of ZeroCouplage was finally set up. Before, coding in ZeroCouplage without strong knowledge of Java caused havoc to developers. So they couldn't go straight forward to learning ZeroCouplage, they had to learn first Java programming. In fact, this was a little bit confusing.

Solution: Coding with XML!

XML has a lot of benefits: structured data, obvious, comprehensible to anyone who understands HTML, and more powerful.

Ok! But what is the big deal???

In this tutorial, we'll explain exactly what the big deal is – how XML can be used to create easily and automatically a java class.

Before proceeding with this tutorial, let's have a look of an example of generated java class from xml file :

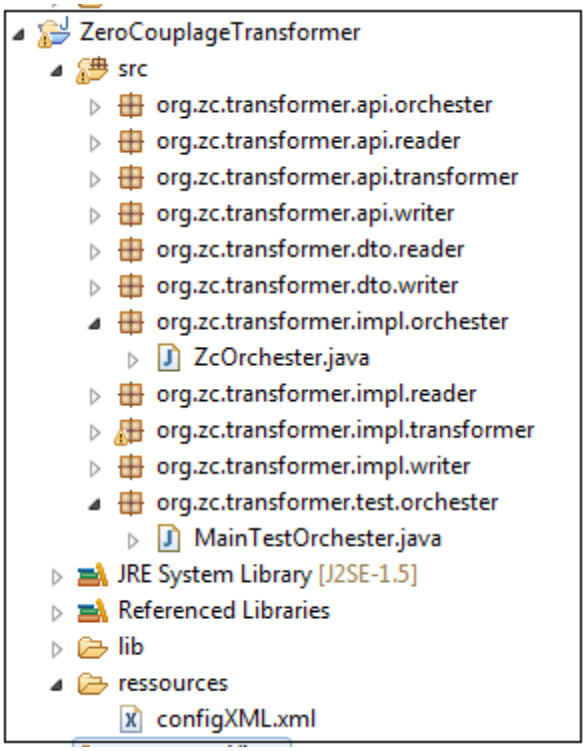
```xml
<Page name="page" text="inscription" title="Inscription">

    <GridLayout name="layout" rows="4" cols="1" text="sign in">

        <Label name="labelOfLogin" label="Entrer Login :">
        </Label>


        <Button name="valider" text="Valider" action="goTraitementInscription">
        </Button>

        <Button name="reset" text="Valider" action="goInscription">
        </Button>

    </GridLayout>

</Page>
```

```java
package ma.ensao;

import org.zerocouplage.component.api.view.ZCView;

public class Inscription implements ZCView {
    private ZCPage page;
    private ZCGridLayout layout;
    private ZCLabel labelOfLogin;
    private ZCButton valider;
    private ZCButton reset;

    @Override
    public ZCPage display() throws IOException, ZCExceptionConfig, ZCCompNotFoundException, Exception  {
        page =(ZCPage) ZCComponentFactory.newComponent(ZCPage.class);
        page.setName("inscription");
        page.setTitle("Inscription");
        layout =(ZCGridLayout) ZCComponentFactory.newComponent(ZCGridLayout.class);
        page.setBody(layout);
        layout.setName("sign in");
        layout.setRows(4);
        layout.setCols(1);
        labelOfLogin =(ZCLabel) ZCComponentFactory.newComponent(ZCLabel.class);
        layout.addComponent(labelOfLogin);
        labelOfLogin.setLabel("Entrer Login :");
        valider =(ZCButton) ZCComponentFactory.newComponent(ZCButton.class);
        layout.addComponent(valider);
        valider.setAction("goTraitementInscription",this);
        valider.setText("Valider");
        reset =(ZCButton) ZCComponentFactory.newComponent(ZCButton.class);
        layout.addComponent(reset);
        reset.setAction("goInscription",this);
        reset.setText("Valider");
        return page;
    }
}
```

How To Build Your java class with xml file ?

1) Import projet of ZeroCouplage from the following link :
(Link of SVN or Github)

The project must be like this:



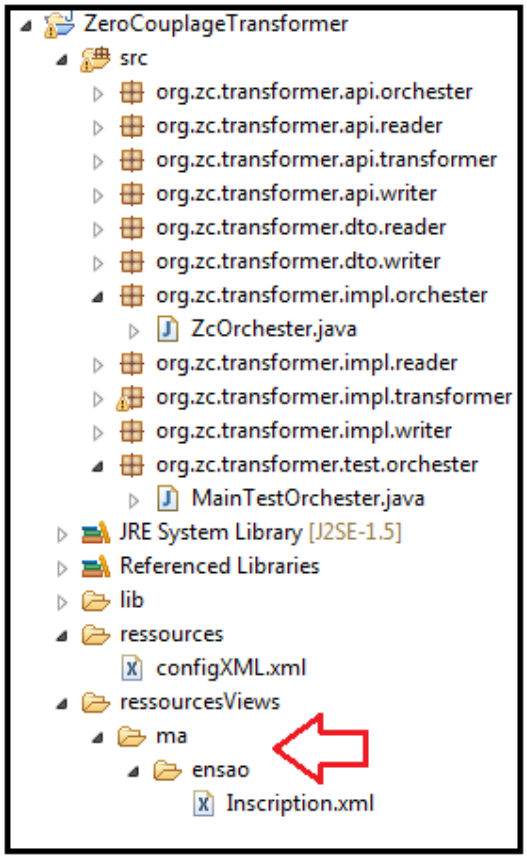
ConfigXML.xml contains all ZCcomponent with their equivalent in Java .

MainTestOrchester.java : class main that instantiates and launches the **ZcOrchester.java**

2) Create a folder called for example "**RessourceViews**" , in this folder you can create as many folder as you want , but take in consideration that folders containing in **"RessourceViews"** will be the name of package where it will be placed your java class.

Example:

RessourceViews/ma/ensao → *package*: ma.ensao



3) Open the class ZcOrchester.java and change paths : 1 and 2

ZCOrchester.java

```
package org.zc.transformer.impl.orchester;

import java.io.File;

public class ZcOrchester {

    private String[] listpathViewXML;
    //File [] listpathViewXML;
    private ReaderDTO readerDTO;
    private ConfigDTO configDTO;
    private JavaDTO javaDTO;
    private ZCReader zcreader;
    private ZCTransformer zctransformer;
    private ZcWriter zcwriter;
    private ZCReaderConfig zcReaderConfig;

    public void ochester(File pathViewFolder, String outputFolder) {

        listpathViewXML = pathViewFolder.list();
        for(int i=0;i<listpathViewXML.length;i++){
            if(listpathViewXML[i].endsWith(".xml")==true){
                String pathView2 = listpathViewXML[i];
                String path =pathViewFolder+"/"+pathView2;
                String pathZCReader = path.replace("\\", "/");

                zcreader = new ZCReader();                                                          1
                readerDTO =  zcreader.ReadView(pathZCReader,"D:/ENSAO/GI4_S2/PFA/ZC_Projects/ZeroCouplageTransformer/ressourcesViews/");
                zcReaderConfig = new ZCReaderConfig();
                configDTO =  zcReaderConfig.read("D:/ENSAO/GI4_S2/PFA/ZC_Projects/ZeroCouplageTransformer/ressources/configXML.xml");
                                                                                                    2
                zctransformer=new org.zc.transformer.impl.transformer.ZCTransformer();
                javaDTO = new JavaDTO();
                javaDTO=zctransformer.transformer(readerDTO, configDTO);
                zcwriter = new ZcWriter();

                zcwriter.write(javaDTO, outputFolder);

        }
    }
}
}
```

Path1: path of your View folder "ressourcesViews" in your disk.

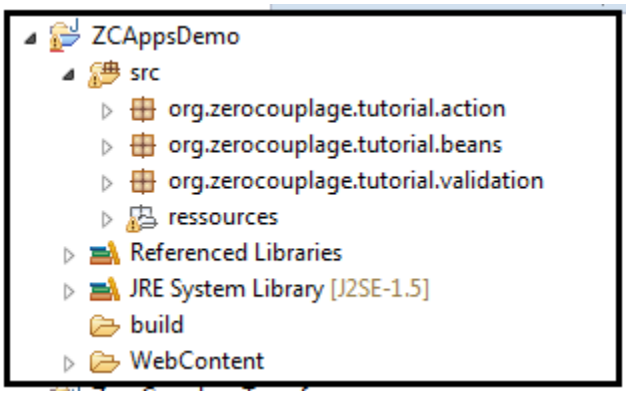
Path2: path of configXML.xml in your disk.

How to get this paths ?

-Click right on the file or folder → properties → Location → get the path !

4) Import projet of the demo from the following link :
(Link of SVN or Git)

The project must be like this:



5) Go back to the project of ZeroCouplageTransformer and then to the main class of orchester : **MainTestOrchester.java , a**fter that ,Change paths again : 1 & 2 .

-**Path1**: path of your View "**inscription.xml**" in your disk.

-**Path2**: path of "src" in your project ("ZCAppsDemo").

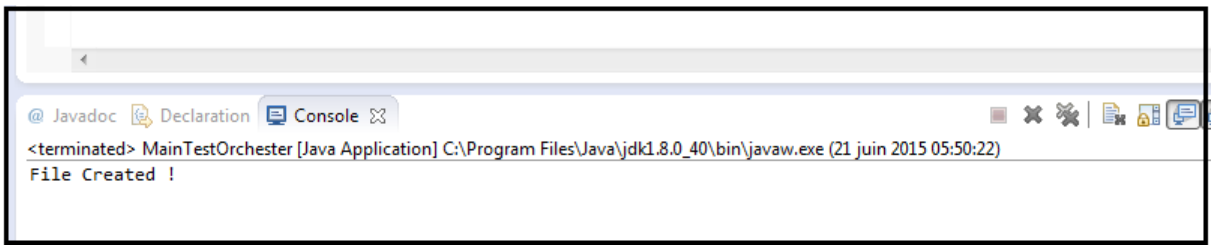
MainTestOrchester.java

```
package org.zc.transformer.test.orchester;


import java.io.File;

public class MainTestOrchester {
    public static void main(String[] args) {

        ZcOrchester zcOrchester = new ZcOrchester();
                                                                    1
        File pathview = new File("D:/ENSAO/GI4_S2/PFA/ZC_Projects/ZeroCouplageTransformer/ressourcesViews/ma/ensao/");
        zcOrchester.ochester(pathview , "D://ENSAO//GI4_S2//PFA//ZC_Projects//ZCAppsDemo//src");
                                                                    2
    }
}
```

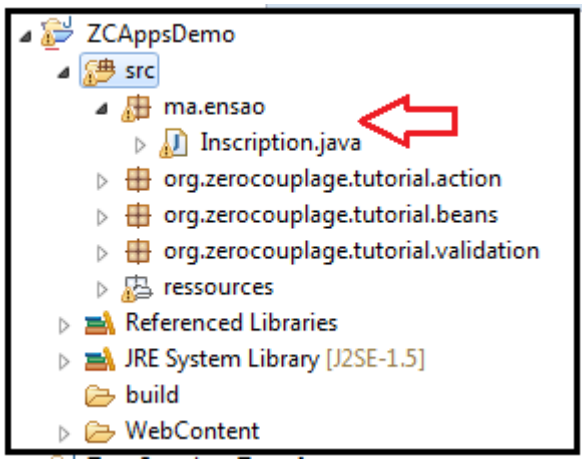
6) Go to "**MainTestOrchester.java**" and **run as Java application** .

If you get "**File created**" like above, then go to step 7 .

If you get an error, try to fix them first .



7) Go to "**src**" of your project "**ZcAppsDemo**" and then press **F5**
8)



Congratulation! your java class is created in you package **ma.ensao**

Inscription.java

```java
package ma.ensao;

import org.zerocouplage.component.api.view.ZCView;

public class Inscription implements ZCView {
    private ZCPage page;
    private ZCGridLayout layout;
    private ZCLabel labelOfLogin;
    private ZCButton valider;
    private ZCButton reset;

    @Override
    public ZCPage display() throws IOException, ZCExceptionConfig, ZCCompNotFoundException, Exception   {
        page =(ZCPage) ZCComponentFactory.newComponent(ZCPage.class);
        page.setName("inscription");
        page.setTitle("Inscription");
        layout =(ZCGridLayout) ZCComponentFactory.newComponent(ZCGridLayout.class);
        page.setBody(layout);
        layout.setName("sign in");
        layout.setRows(4);
        layout.setCols(1);
        labelOfLogin =(ZCLabel) ZCComponentFactory.newComponent(ZCLabel.class);
        layout.addComponent(labelOfLogin);
        labelOfLogin.setLabel("Entrer Login :");
        valider =(ZCButton) ZCComponentFactory.newComponent(ZCButton.class);
        layout.addComponent(valider);
        valider.setAction("goTraitementInscription",this);
        valider.setText("Valider");
        reset =(ZCButton) ZCComponentFactory.newComponent(ZCButton.class);
        layout.addComponent(reset);
        reset.setAction("goInscription",this);
        reset.setText("Valider");
        return page;
    }


}
```

The end