

```
1 package com.ustb.cn;
2 import com.ustb.cn.BussinessProcess.ProcessTree;
3 import com.ustb.cn.CreateUCCF.CreateUCCF;
4 import com.ustb.cn.CreateUCCF.Read_UCCF;
5 import com.ustb.cn.CreateUCCF.UccfInof;
6 import com.ustb.cn.DrawFMPic.DrawFMDiagram;
7 import com.ustb.cn.ModelTransformation.ModelTran;
8 import com.ustb.cn.ValidateConstraints.Validate;
9 import org.dom4j.DocumentException;
10 import javax.swing.*;
11 import javax.swing.border.TitledBorder;
12 import javax.swing.event.ListSelectionEvent;
13 import javax.swing.event.ListSelectionListener;
14 import javax.swing.filechooser.FileNameExtensionFilter;
15 import java.awt.*;
16 import java.awt.event.ActionEvent;
17 import java.awt.event.ActionListener;
18 public class Main extends JFrame{
19     FMTreeJPanel fmTree;
20     String Selected_String;
21     Font jTextArea_Font=new Font("Consolas",Font.PLAIN,12);
22     String url_FM_FilePath;
23     String url_FM_Save_Location;
24     ArrayList<String> arrayList=new ArrayList<String>();
25     String FMDiagram_Name;
26     JTextArea jTextArea_Logger_Info;
27     JTextArea jTextArea_FM_XMI;
28     String url_FM_FilePath_2_1;
29     String url_VxBPEL_Save_Location_2_2;
30     String vxbpel_name;
31     String vxbpel_file_location;
32     ArrayList<ArrayList> arrayLists_VP;
33     String UCCF_Save_Location;
34     String UCCF_File_Name;
35     public Main(){
36         this.setTitle("FM2VxBPEL");
37         this.setSize(630,760);
38         this.setLayout(new BorderLayout(2, 2));
39         JPanel jPanel_North=new JPanel(new BorderLayout());
40         jPanel_North.setPreferredSize(new Dimension(0,25));
41         JMenuBar jMenuBar=new JMenuBar();//添加菜单栏
42         jPanel_North.add(jMenuBar,BorderLayout.NORTH);
43         JMenu mFile=new JMenu("File");
44         JMenuItem mFile_Close=new JMenuItem("Close");
45         mFile.add(mFile_Close);
46         mFile_Close.addActionListener(new ActionListener() {
47             @Override
48             public void actionPerformed(ActionEvent e) {
49                 dispose();
50             }
51         })
52     }
53 }
```

```

51     });
52     jMenuBar.add(mFile);
53     JMenu mEngineManage=new JMenu("Engine");
54     JMenuItem startODE=new JMenuItem("Start ODE");
55     JMenuItem stopODE=new JMenuItem("Stop ODE");
56     mEngineManage.add(startODE);
57     mEngineManage.add(stopODE);
58     jMenuBar.add(mEngineManage);
59     JMenu mRun=new JMenu("Run");
60     JMenuItem runProcess=new JMenuItem("Run Process");
61     runProcess.addActionListener(new ActionListener() {
62         @Override
63         public void actionPerformed(ActionEvent e) {
64             RunProcess r=new RunProcess();
65         }
66     });
67     mRun.add(runProcess);
68     jMenuBar.add(mRun);
69     JMenu m_Tools=new JMenu("Tools");
70     JMenuItem businessManage=new JMenuItem("Business Manage");
71     businessManage.addActionListener(new ActionListener() {
72         @Override
73         public void actionPerformed(ActionEvent e) {
74             BusinessManage.openDefaultBrowser();
75         }
76     });
77     m_Tools.add(businessManage);
78     jMenuBar.add(m_Tools);
79     JMenu mHelp=new JMenu("Help");
80     JMenuItem metamodel=new JMenuItem("FM MetaModel");
81     metamodel.addActionListener(new ActionListener() {
82         @Override
83         public void actionPerformed(ActionEvent e) {
84             FeatureModelMatamodel fmm=new FeatureModelMatamodel();
85         }
86     });
87     mHelp.add(metamodel);
88     jMenuBar.add(mHelp);
89     final JPanel jPanel_Center=new JPanel(new BorderLayout());
90     JTabbedPane jTabbedPane1=new JTabbedPane();
91     jPanel_Center.add(jTabbedPane1);
92     final JPanel jPanel_Visual_Customizing=new JPanel(new BorderLayout());
93     jTabbedPane1.add("Visual Customizing",jPanel_Visual_Customizing);
94     JPanel jPanel_Visual_Customizing_North=new JPanel(null);
95     jPanel_Visual_Customizing_North.setBorder(new
96     TitledBorder(UIManager.getBorder("TitledBorder.border"), "Loading File",
97     TitledBorder.CENTER, TitledBorder.TOP, null, null));
98     jPanel_Visual_Customizing_North.setPreferredSize(new Dimension(0,105));
99     jPanel_Visual_Customizing.add(jPanel_Visual_Customizing_North,BorderLayout.NORTH);
100    JLabel jLabel_1=new JLabel("Select FM File:");

```

```

101     jLabel_1.setBounds(8,20,100,20);
102     jPanel_Visual_Customizing_North.add(jLabel_1);
103     final JTextField jt1=new JTextField();
104     jt1.setBounds(135,20,290,20);
105     jPanel_Visual_Customizing_North.add(jt1);
106     MyButton myButton_Select_FM_File=new MyButton("Browse",430,20,80,20);
107     jPanel_Visual_Customizing_North.add(myButton_Select_FM_File);
108     myButton_Select_FM_File.addActionListener(new ActionListener() {
109         @Override
110         public void actionPerformed(ActionEvent e) {
111             JFileChooser fcDlg = new JFileChooser();
112             fcDlg.setDialogTitle("Please choose file:");
113             FileNameExtensionFilter filter = new
114 FileNameExtensionFilter("file(*.scfeaturemodel)", "scfeaturemodel");
115             fcDlg.setFileFilter(filter);
116             int returnVal = fcDlg.showOpenDialog(null);
117             if (returnVal == JFileChooser.APPROVE_OPTION) {
118                 String filepath = fcDlg.getSelectedFile().getPath();
119                 url_FM_FilePath=filepath;
120                 jt1.setText(url_FM_FilePath);
121                 JTextArea_Logger_Info.append("Start!\r\n");
122                 JTextArea_Logger_Info.append("The location of Feature Model
123 file!" + filepath + "\r\n");
124                 File f=new File(url_FM_FilePath);
125                 System.out.println(filepath);
126             }
127         }
128     });
129     MyButton myButton_Cancel1=new MyButton("Cancel",515,20,80,20);
130     jPanel_Visual_Customizing_North.add(myButton_Cancel1);
131     myButton_Cancel1.addActionListener(new CancelActionListener(jt1));
132     JLabel jLabel_2=new JLabel("Select Save Location:");
133     jLabel_2.setBounds(8,50,130,20);
134     jPanel_Visual_Customizing_North.add(jLabel_2);
135     final JTextField jt2=new JTextField();
136     jt2.setBounds(135,50,290,20);
137     url_FM_Save_Location=jt2.getText();
138     jPanel_Visual_Customizing_North.add(jt2);
139     MyButton myButton_Select_SaveLocation=new MyButton("Browse",430,50,80,20);
140     jPanel_Visual_Customizing_North.add(myButton_Select_SaveLocation);
141     myButton_Select_SaveLocation.addActionListener(new ActionListener() {
142         public void actionPerformed(ActionEvent e) {
143             JFileChooser fcDlg = new JFileChooser();
144             fcDlg.setDialogTitle("Please select the location where the feature model
145 picture is saved");
146             fcDlg.setFileSelectionMode(JFileChooser.DIRECTORIES_ONLY);
147             int returnVal = fcDlg.showOpenDialog(null);
148             if (returnVal == JFileChooser.APPROVE_OPTION) {
149                 String filepath = fcDlg.getSelectedFile().getPath();
150                 url_FM_Save_Location=filepath;

```

```

151         jt2.setText(url_FM_Save_Location);
152         jTextArea_Logger_Info.append("The location of
153 diagram!" + filepath + "\r\n");
154     }
155 }
156 });
157 MyButton myButton_Cancel2=new MyButton("Cancel",515,50,80,20);
158 jPanel_Visual_Customizing_North.add(myButton_Cancel2);
159 myButton_Cancel2.addActionListener(new CancelActionListener(jt2));
160 JLabel jLabel_3=new JLabel("Input Diagram Name:");
161 jLabel_3.setBounds(8,80,130,20);
162 jPanel_Visual_Customizing_North.add(jLabel_3);
163 final JTextField jt3=new JTextField();
164 jt3.setBounds(135,80,290,20);
165 jPanel_Visual_Customizing_North.add(jt3);
166 MyButton myButton_Create_Diagram=new MyButton("Create",430,80,80,20);
167 jPanel_Visual_Customizing_North.add(myButton_Create_Diagram);
168 myButton_Create_Diagram.addActionListener(new ActionListener() {
169     @Override
170     public void actionPerformed(ActionEvent e) {
171         try {
172             Read_FM_XMI rfx=new Read_FM_XMI(url_FM_FilePath);
173             jTextArea_FM_XMI.append(rfx.result);
174         } catch (IOException e1) {
175             e1.printStackTrace();
176         }
177         url_FM_Save_Location=jt2.getText();
178         FMDiagram_Name=jt3.getText();
179         try {
180             DrawFMDiagram drawFMDiagram=new
181 DrawFMDiagram(url_FM_FilePath,url_FM_Save_Location,FMDiagram_Name);
182         } catch (DocumentException e1) {
183             e1.printStackTrace();
184         } catch (IOException e1) {
185             e1.printStackTrace();
186         }
187         JPanel jPanel_FM_VisualTree=new JPanel(new BorderLayout());
188         jPanel_FM_VisualTree.setBorder(new
189 TitledBorder(UIManager.getBorder("TitledBorder.border"), "Visual Tree",
190 TitledBorder.CENTER, TitledBorder.TOP, null, null));
191         jPanel_FM_VisualTree.setPreferredSize(new Dimension(0,415));
192 jPanel_Visual_Customizing.add(jPanel_FM_VisualTree,BorderLayout.SOUTH);
193         try {
194             fmTree=new FMTreeJPanel(url_FM_FilePath);
195             JPanel fmTreeJPanel=fmTree.jPanel;
196             System.out.println("Selected_String"+Selected_String);
197             fmTreeJPanel.setBorder(new
198 TitledBorder(UIManager.getBorder("TitledBorder.border"), "Configurable FM Tree",
199 TitledBorder.CENTER, TitledBorder.TOP, null, null));
200             fmTreeJPanel.setPreferredSize(new Dimension(250,375));

```

```

201         jPanel_FM_VisualTree.add(fmTreeJPanel, BorderLayout.WEST);
202         jPanel_FM_VisualTree.setVisible(true);
203         jPanel_Center.repaint();
204     } catch (IOException e1) {
205         e1.printStackTrace();
206     } catch (AWTException e1) {
207         e1.printStackTrace();
208     } catch (DocumentException e1) {
209         e1.printStackTrace();
210     }
211     JPanel jPanel_Configuration_Info=new JPanel(null);
212     jPanel_Configuration_Info.setBorder(new
213 TitledBorder(UIManager.getBorder("TitledBorder.border"), "Configuration",
214 TitledBorder.CENTER, TitledBorder.TOP, null, null));
215     jPanel_Configuration_Info.setPreferredSize(new Dimension(250,415));
216     JPanel jPanel_Configuration_Info_Table=new JPanel(null);
217     jPanel_Configuration_Info_Table.setBorder(new
218 TitledBorder(UIManager.getBorder("TitledBorder.border"), "Constraints Table",
219 TitledBorder.LEFT, TitledBorder.TOP, null, null));
220     jPanel_Configuration_Info_Table.setBounds(20,25,210,120);
221     jPanel_Configuration_Info.add(jPanel_Configuration_Info_Table);
222     try {
223         ConstraintTable ct=new ConstraintTable(url_FM_FilePath);
224         JScrollPane jScrollPane_Constraint_Table=ct.jtableJScrollPane;
225         jScrollPane_Constraint_Table.setBounds(5,20,200,90);
226         jPanel_Configuration_Info_Table.add(jScrollPane_Constraint_Table);
227     } catch (DocumentException e1) {
228         e1.printStackTrace();
229     }
230     JPanel jPanel_jList_Configuration_Info=new JPanel(null);
231     jPanel_jList_Configuration_Info.setBorder(new
232 TitledBorder(UIManager.getBorder("TitledBorder.border"), "FM Configuration Info",
233 TitledBorder.LEFT, TitledBorder.TOP, null, null));
234     jPanel_jList_Configuration_Info.setBounds(20,155,210,230);
235     jPanel_Configuration_Info.add(jPanel_jList_Configuration_Info);
236     final DefaultListModel dlm=new DefaultListModel();
237     final JList jList = new JList(dlm);
238     jList.setBounds(5,20,200,205);
239     jPanel_jList_Configuration_Info.add(jList);
240     final String[] jlist_selected_info = new String[1];
241     jList.addListSelectionListener(new ListSelectionListener(){
242         @Override
243         public void valueChanged(ListSelectionEvent e) {
244             if(!jList.getValueIsAdjusting()){
245                 System.out.println(jList.getSelectedValue());
246                 jlist_selected_info[0] = (String) jList.getSelectedValue();
247             }
248         }
249     });
250     jPanel_FM_VisualTree.add(jPanel_Configuration_Info,BorderLayout.EAST);

```

```

251         JPanel jPanel_JButton=new JPanel(null);
252         jPanel_FM_VisualTree.add(jPanel_JButton,BorderLayout.CENTER);
253         MyButton myButton_Add=new MyButton("Add",7,100,80,20);
254         myButton_Add.addActionListener(new ActionListener() {
255             @Override
256             public void actionPerformed(ActionEvent e) {
257                 Selected_String=fmTree.selected_String;
258                 System.out.println("Selected_String"+Selected_String);
259                 dlm.addElement(Selected_String);
260                 arrayList.add(Selected_String);
261             }
262         });
263         MyButton myButton_Remove=new MyButton("Remove",7,150,80,20);
264         myButton_Remove.addActionListener(new ActionListener() {
265             @Override
266             public void actionPerformed(ActionEvent e) {
267                 arrayList.remove(jlist_selected_info[0]);
268                 dlm.removeElement(jlist_selected_info[0]);
269             }
270         });
271         MyButton myButton_Validate=new MyButton("Validate",7,200,80,20);
272         myButton_Validate.addActionListener(new ActionListener() {
273             @Override
274             public void actionPerformed(ActionEvent e) {
275                 try {
276                     Validate v=new Validate(url_FM_FilePath,arrayList);
277                     if(v.flag.equals("true")){
278                         MyDialog_Confirm myDialog_confirm=new
279 MyDialog_Confirm("Confirm to FM constraints!");
280                     }else if(v.flag.equals("false")){
281                         MyDialog_Error myDialog_error=new
282 MyDialog_Error("Error!");
283                     }
284                 } catch (DocumentException e1) {
285                     e1.printStackTrace();
286                 }
287             }
288         });
289         jPanel_JButton.add(myButton_Add);
290         jPanel_JButton.add(myButton_Remove);
291         jPanel_JButton.add(myButton_Validate);
292         jPanel_Visual_Customizing.add(jPanel_FM_VisualTree,BorderLayout.SOUTH);
293         jPanel_Center.repaint();
294     }
295 };
296 MyButton myButton_Cancel3=new MyButton("Cancel",515,80,80,20);
297 jPanel_Visual_Customizing_North.add(myButton_Cancel3);
298 myButton_Cancel3.addActionListener(new CancelActionListener(jt3));
299 final JPanel jPanel_Model_Transformation=new JPanel(new BorderLayout());
300 jTabbedPane1.add("Model Transformation",jPanel_Model_Transformation);

```

```

301     JPanel jPanel_Model_Transformation_North=new JPanel(null);
302     jPanel_Model_Transformation_North.setBorder(new
303     TitledBorder(UIManager.getBorder("TitledBorder.border"), "Loading File",
304     TitledBorder.CENTER, TitledBorder.TOP, null, null));
305     jPanel_Model_Transformation_North.setPreferredSize(new Dimension(0,140));
306     jPanel_Model_Transformation.add(
307     jPanel_Model_Transformation_North, BorderLayout.NORTH);
308     JLabel jLabel_2_1=new JLabel("Select FM File:");
309     jLabel_2_1.setBounds(8,20,100,20);
310     jPanel_Model_Transformation_North.add(jLabel_2_1);
311     final JTextField jt_2_1=new JTextField();
312     jt_2_1.setBounds(135,20,290,20);
313     jPanel_Model_Transformation_North.add(jt_2_1);
314     MyButton myButton_Select_FM_File_2_1=new MyButton("Browse",430,20,80,20);
315     jPanel_Model_Transformation_North.add(myButton_Select_FM_File_2_1);
316     myButton_Select_FM_File_2_1.addActionListener(new ActionListener() {
317         @Override
318         public void actionPerformed(ActionEvent e) {
319             JFileChooser fcDlg = new JFileChooser();
320             fcDlg.setDialogTitle("Please choose file:");
321             FileNameExtensionFilter filter = new
322             FileNameExtensionFilter("file(*.scfeaturemodel)", "scfeaturemodel");
323             fcDlg.setFileFilter(filter);
324             int returnVal = fcDlg.showOpenDialog(null);
325             if (returnVal == JFileChooser.APPROVE_OPTION) {
326                 String filepath = fcDlg.getSelectedFile().getPath();
327                 url_FM_FilePath_2_1=filepath;
328                 jt_2_1.setText(url_FM_FilePath_2_1);
329                 JTextArea_Logger_Info.append("The location of Feature Model
330 file!" + filepath + "\r\n");
331                 File f=new File(url_FM_FilePath_2_1);
332                 System.out.println(filepath);
333             }
334         }
335     });
336     MyButton myButton_Cancel_2_1=new MyButton("Cancel",515,20,80,20);
337     jPanel_Model_Transformation_North.add(myButton_Cancel_2_1);
338     myButton_Cancel_2_1.addActionListener(new CancelActionListener(jt_2_1));
339     JLabel jLabel_2_2=new JLabel("Select Save Location:");
340     jLabel_2_2.setBounds(8,50,130,20);
341     jPanel_Model_Transformation_North.add(jLabel_2_2);
342     final JTextField jt_2_2=new JTextField();
343     jt_2_2.setBounds(135,50,290,20);
344     jPanel_Model_Transformation_North.add(jt_2_2);
345     MyButton myButton_Select_SaveLocation_2_2=new
346     MyButton("Browse",430,50,80,20);
347     jPanel_Model_Transformation_North.add(myButton_Select_SaveLocation_2_2);
348     myButton_Select_SaveLocation_2_2.addActionListener(new ActionListener() {
349         @Override
350         public void actionPerformed(ActionEvent e) {

```

```

351         JFileChooser fcDlg = new JFileChooser();
352         fcDlg.setDialogTitle("Please select the location where the feature model
353 picture is saved");
354         fcDlg.setFileSelectionMode(JFileChooser.DIRECTORIES_ONLY);
355         int returnVal = fcDlg.showOpenDialog(null);
356         if (returnVal == JFileChooser.APPROVE_OPTION) {
357             String filepath = fcDlg.getSelectedFile().getPath();
358             url_VxBPEL_Save_Location_2_2=filepath;
359             jt_2_2.setText(url_VxBPEL_Save_Location_2_2);
360             jTextArea_Logger_Info.append("The location of
361 VxBPEL!" + filepath + "\r\n");
362         }
363     }
364 });
365 MyButton myButton_Cancel_2_2=new MyButton("Cancel",515,50,80,20);
366 jPanel_Model_Transformation_North.add(myButton_Cancel_2_2);
367 myButton_Cancel_2_2.addActionListener(new CancelActionListener(jt_2_2));
368 JLabel jLabel_2_3=new JLabel("Input VxBPEL Name:");
369 jLabel_2_3.setBounds(8,80,130,20);
370 jPanel_Model_Transformation_North.add(jLabel_2_3);
371 final JTextField jt_2_3=new JTextField();
372 jt_2_3.setBounds(135,80,290,20);
373 jPanel_Model_Transformation_North.add(jt_2_3);
374 MyButton myButton_Model_Convert=new MyButton("Convert",430,80,80,20);
375 jPanel_Model_Transformation_North.add(myButton_Model_Convert);
376 //myButton_Model_Convert 执行事件
377 myButton_Model_Convert.addActionListener(new ActionListener() {
378     @Override
379     public void actionPerformed(ActionEvent e) {
380         vxbpel_name=jt_2_3.getText();
381         try {
382             ModelTran mt=new
383 ModelTran(url_FM_FilePath_2_1,url_VxBPEL_Save_Location_2_2,vxbpel_name);
384         } catch (DocumentException e1) {
385             e1.printStackTrace();
386         } catch (IOException e1) {
387             e1.printStackTrace();
388         }
389     }
390 });
391 MyButton myButton_Cancel_2_3=new MyButton("Cancel",515,80,80,20);
392 jPanel_Model_Transformation_North.add(myButton_Cancel_2_3);
393 myButton_Cancel_2_3.addActionListener(new CancelActionListener(jt_2_3));
394 JLabel jLabel_2_4=new JLabel("Parse Variability Info:");
395 jLabel_2_4.setBounds(8,110,130,20);
396 jPanel_Model_Transformation_North.add(jLabel_2_4);
397 final JTextField jt_2_4=new JTextField();
398 jt_2_4.setBounds(135,110,290,20);
399 jPanel_Model_Transformation_North.add(jt_2_4);
400 MyButton myButton_Browse_2_4=new MyButton("Browse",430,110,80,20);

```



```

401     jPanel_Model_Transformation_North.add(myButton_Browse_2_4);
402     myButton_Browse_2_4.addActionListener(new ActionListener() {
403         public void actionPerformed(ActionEvent e) {
404             JFileChooser fcDlg = new JFileChooser();
405             fcDlg.setDialogTitle("Please choose file:");
406             FileNameExtensionFilter filter =
407                 new FileNameExtensionFilter("file(*.bpel)", "bpel");
408             fcDlg.setFileFilter(filter);
409             int returnVal = fcDlg.showOpenDialog(null);
410             if (returnVal == JFileChooser.APPROVE_OPTION) {
411                 String filepath = fcDlg.getSelectedFile().getPath();
412                 vxbpel_file_location=filepath;
413                 jt_2_4.setText(vxbpel_file_location);
414                 jTextArea_Loger_Info.append("Start!\r\n");
415                 jTextArea_Loger_Info.append("The location of vxbpel
416 file!" + filepath + "\r\n");
417                 System.out.println(filepath);
418             }
419         }
420     });
421     MyButton myButton_Parse_2_4=new MyButton("Parse",515,110,80,20);
422     jPanel_Model_Transformation_North.add(myButton_Parse_2_4);
423     myButton_Parse_2_4.addActionListener(new ActionListener() {
424         @Override
425         public void actionPerformed(ActionEvent e) {
426             JPanel jPanel_VxBPEL=new JPanel(new BorderLayout());
427             jPanel_VxBPEL.setBorder(new
428 TitledBorder(UIManager.getBorder("TitledBorder.border"), "VxBPEL",
429 TitledBorder.CENTER, TitledBorder.TOP, null, null));
430             jPanel_VxBPEL.setPreferredSize(new Dimension(0,380));
431             jPanel_Model_Transformation.add(jPanel_VxBPEL,BorderLayout.SOUTH);
432             try {
433                 ProcessTree pt=new ProcessTree(vxbpel_file_location);
434                 JPanel jPanel_vxbpel_Tree=pt.jPanel;
435                 jPanel_vxbpel_Tree.setBorder(new
436 TitledBorder(UIManager.getBorder("TitledBorder.border"), "Process Tree",
437 TitledBorder.CENTER, TitledBorder.TOP, null, null));
438                 jPanel_vxbpel_Tree.setPreferredSize(new Dimension(250,325));
439                 jPanel_VxBPEL.add(jPanel_vxbpel_Tree,BorderLayout.WEST);
440             } catch (DocumentException e1) {
441                 e1.printStackTrace();
442             }
443             JPanel jPanel_vxbpel_configuration_info=new JPanel(null);
444             jPanel_vxbpel_configuration_info.setBorder(new
445 TitledBorder(UIManager.getBorder("TitledBorder.border"), "VxBPEL Configuration Info",
446 TitledBorder.CENTER, TitledBorder.TOP, null, null));
447             jPanel_vxbpel_configuration_info.setPreferredSize(new
448 Dimension(250,325));
449             jPanel_VxBPEL.add(jPanel_vxbpel_configuration_info,BorderLayout.EAST);
450             JPanel jPanel_fm_conf_info=new JPanel(null);

```

```

451         jPanel_fm_conf_info.setBounds(10,20,230,140);
452         jPanel_fm_conf_info.setBorder(new
453 TitledBorder(UIManager.getBorder("TitledBorder.border"), "FM Configuration Info",
454 TitledBorder.LEFT, TitledBorder.TOP, null, null));
455         JTextArea jTextArea_fm_conf_info=new JTextArea();
456         JScrollPane jScrollPane_fm_conf_info=new
457 JScrollPane(jTextArea_fm_conf_info);
458         jScrollPane_fm_conf_info.setBounds(5,20,220,110);
459         jPanel_fm_conf_info.add(jScrollPane_fm_conf_info);
460         jPanel_vxbpel_configuration_info.add(jPanel_fm_conf_info);
461         if(arrayList.size()>0){
462             for(int i=0;i<arrayList.size();i++){
463                 jTextArea_fm_conf_info.append(arrayList.get(i)+"\r\n");
464             }
465         }
466         JPanel jPanel_Match_vp_variant=new JPanel(null);
467         jPanel_Match_vp_variant.setBounds(10,170,230,180);
468         jPanel_Match_vp_variant.setBorder(new
469 TitledBorder(UIManager.getBorder("TitledBorder.border"), "VariationPoint && Variant
470 Conf", TitledBorder.LEFT, TitledBorder.TOP, null, null));
471         final JTextArea jTextArea_user_conf=new JTextArea();
472         JScrollPane jScrollPane_user_conf=new JScrollPane(jTextArea_user_conf);
473         jScrollPane_user_conf.setBounds(5,20,220,155);
474         jPanel_Match_vp_variant.add(jScrollPane_user_conf);
475         jPanel_vxbpel_configuration_info.add(jPanel_Match_vp_variant);
476         JPanel jPanel_JButton2=new JPanel(null);
477         jPanel_VxBPEL.add(jPanel_JButton2,BorderLayout.CENTER);
478         MyButton myButton_Match_VP=new MyButton("Match",7,120,80,20);
479         myButton_Match_VP.addActionListener(new ActionListener() {
480             @Override
481             public void actionPerformed(ActionEvent e) {
482                 arrayList_VP=new ArrayList<ArrayList>();
483                 for(int i=0;i<arrayList.size();i++){
484                     try {
485                         FindVP findVP=new
486 FindVP(vxbpel_file_location,arrayList.get(i));
487 jTextArea_user_conf.append("VariationPoint->"+findVP.flag1+"\r\n");
488 jTextArea_user_conf.append("
489 Variant->"+findVP.flag2+"\r\n");
490 ArrayList<String> a=new ArrayList<String>();
491 a.add(findVP.flag1);
492 a.add(findVP.flag2);
493 arrayList_VP.add(a);
494 } catch (DocumentException e1) {
495     e1.printStackTrace();
496 }
497 }
498 }
499 });
500 MyButton myButton_UCCF=new MyButton("UCCF",7,170,80,20);

```

```

501         jPanel_JButton2.add(myButton_Match_VP);
502         jPanel_Center.repaint();
503     }
504 });
505 final JPanel jPanel_UCCF=new JPanel(new BorderLayout());
506 jTabbedPane1.add("Create UCCF",jPanel_UCCF);
507 JPanel jPanel_UCCF_North=new JPanel(null);
508 jPanel_UCCF_North.setBorder(new
509 TitledBorder(UIManager.getBorder("TitledBorder.border"), "Create UCCF",
510 TitledBorder.CENTER, TitledBorder.TOP, null, null));
511 jPanel_UCCF_North.setPreferredSize(new Dimension(0,75));
512 jPanel_UCCF.add(jPanel_UCCF_North,BorderLayout.NORTH);
513 JLabel jLabel_3_1=new JLabel("Select Save Location:");
514 jLabel_3_1.setBounds(8,20,140,20);
515 jPanel_UCCF_North.add(jLabel_3_1);
516 final JTextField jt_3_1=new JTextField();
517 jt_3_1.setBounds(135,20,290,20);
518 jPanel_UCCF_North.add(jt_3_1);
519 MyButton myButton_Select_3_1=new MyButton("Browse",430,20,80,20);
520 jPanel_UCCF_North.add(myButton_Select_3_1);
521 myButton_Select_3_1.addActionListener(new ActionListener() {
522     @Override
523     public void actionPerformed(ActionEvent e) {
524         JFileChooser fcDlg = new JFileChooser();
525         fcDlg.setDialogTitle("Please select the location where the UCCF is saved");
526         fcDlg.setFileSelectionMode(JFileChooser.DIRECTORIES_ONLY);
527         int returnVal = fcDlg.showOpenDialog(null);
528         if (returnVal == JFileChooser.APPROVE_OPTION) {
529             String filepath = fcDlg.getSelectedFile().getPath();
530             UCCF_Save_Location=filepath;
531             jt_3_1.setText(UCCF_Save_Location);
532             jTextArea_Lozer_Info.append("The save location of
533 UCCF!" +filepath+"\r\n");
534         }
535     }
536 });
537 MyButton myButton_Cancel_3_1=new MyButton("Cancel",515,20,80,20);
538 jPanel_UCCF_North.add(myButton_Cancel_3_1);
539 myButton_Cancel_3_1.addActionListener(new CancelActionListener(jt_3_1));
540 JLabel jLabel_3_2=new JLabel("Input UCCF File Name:");
541 jLabel_3_2.setBounds(8,50,140,20);
542 jPanel_UCCF_North.add(jLabel_3_2);
543 final JTextField jt_3_2=new JTextField();
544 jt_3_2.setBounds(135,50,290,20);
545 jPanel_UCCF_North.add(jt_3_2);
546 MyButton myButton_create_UCCF=new MyButton("UCCF",430,50,80,20);
547 jPanel_UCCF_North.add(myButton_create_UCCF);
548 myButton_create_UCCF.addActionListener(new ActionListener() {
549     @Override
550     public void actionPerformed(ActionEvent e) {

```

```

551         JPanel jPanel_UCCF_South=new JPanel(null);
552         jPanel_UCCF_South.setPreferredSize(new Dimension(0,440));
553         jPanel_UCCF_South.setBorder(new
554 TitledBorder(UIManager.getBorder("TitledBorder.border"), "UCCF Info",
555 TitledBorder.CENTER, TitledBorder.TOP, null, null));
556         JTextArea jTextArea_UCCF_Info=new JTextArea();
557         jTextArea_UCCF_Info.setFont(jTextArea_Font);
558         JScrollPane jScrollPane_UCCF_Info=new
559 JScrollPane(jTextArea_UCCF_Info);
560         jScrollPane_UCCF_Info.setBounds(5,20,595,410);
561         jPanel_UCCF_South.add(jScrollPane_UCCF_Info);
562         jPanel_UCCF.add(jPanel_UCCF_South,BorderLayout.SOUTH);
563         UCCF_File_Name=jt_3_2.getText();
564         String
565 all_uccf_location=UCCF_Save_Location+"\\ "+UCCF_File_Name+".uccf";
566         UccfInof u=new UccfInof(arrayLists_VP);
567         try {
568             CreateUCCF c=new CreateUCCF(all_uccf_location,u.s_info);
569         } catch (IOException e1) {
570             e1.printStackTrace();
571         }
572         File f=new File(all_uccf_location);
573         try {
574             Read_UCCF ru=new Read_UCCF(f,arrayLists_VP.size()+2);
575             jTextArea_UCCF_Info.append(ru.result);
576         } catch (IOException e1) {
577             e1.printStackTrace();
578         }
579         jPanel_Center.repaint();
580     }
581 });
582 MyButton myButton_Cancel_3_2=new MyButton("Cancel",515,50,80,20);
583 jPanel_UCCF_North.add(myButton_Cancel_3_2);
584 myButton_Cancel_3_2.addActionListener(new CancelActionListener(jt_3_2));
585 JPanel jPanel_South=new JPanel(new BorderLayout());
586 jPanel_South.setPreferredSize(new Dimension(0,140));
587 JTabbedPane jTabbedPane2=new JTabbedPane();
588 jPanel_South.add(jTabbedPane2,BorderLayout.NORTH);
589 JPanel jPanel_Logger_Info=new JPanel(new BorderLayout());
590 jTabbedPane2.add("Logger Info",jPanel_Logger_Info);
591 jTextArea_Logger_Info=new JTextArea();
592 jTextArea_Logger_Info.setFont(jTextArea_Font);
593 JScrollPane jScrollPane1=new JScrollPane(jTextArea_Logger_Info);
594 jScrollPane1.setPreferredSize(new Dimension(0,110));
595 jPanel_Logger_Info.add(jScrollPane1,BorderLayout.NORTH);
596 JPanel jPanel_FM_XMI=new JPanel(new BorderLayout());
597 jTabbedPane2.add("FM XMI",jPanel_FM_XMI);
598 jTextArea_FM_XMI=new JTextArea();
599 jTextArea_FM_XMI.setFont(jTextArea_Font);
600 JScrollPane jScrollPane2=new JScrollPane(jTextArea_FM_XMI);

```

```
601         jScrollPane2.setPreferredSize(new Dimension(0,110));
602         jPanel_FM_XMI.add(jScrollPane2, BorderLayout.NORTH);
603         this.add(jPanel_North, BorderLayout.NORTH);
604         this.add(jPanel_Center, BorderLayout.CENTER);
605         this.add(jPanel_South, BorderLayout.SOUTH);
606         this.setVisible(true);
607         this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
608         this.setLocationRelativeTo(null);
609     }
610     public static void main(String[] args) {
611         Main m=new Main();
612     }
613 }
614 public class Main_ATL extends JFrame{
615     FMTreeJPanel fmTree;
616     String Selected_String;
617     Font jTextArea_Font=new Font("Consolas",Font.PLAIN,12);
618     String url_FM_FilePath;
619     String url_FM_Save_Location;
620     ArrayList<String> arrayList=new ArrayList<String>();
621     String FMDiagram_Name;
622     JTextArea jTextArea_Logger_Info;
623     JTextArea jTextArea_FM_XMI;
624     String url_FM_FilePath_2_1;
625     String url_VxBPEL_Save_Location_2_2;
626     String vxbpel_name;
627     String vxbpel_file_location;
628     ArrayList<ArrayList> arrayLists_VP;
629     String UCCF_Save_Location;
630     String UCCF_File_Name;
631     public Main_ATL(){
632         this.setTitle("FM2VxBPEL");
633         this.setSize(630,760);
634         this.setLayout(new BorderLayout(2, 2));
635         JPanel jPanel_North=new JPanel(new BorderLayout());
636         jPanel_North.setPreferredSize(new Dimension(0,25));
637         JMenuBar jMenuBar=new JMenuBar();//添加菜单栏
638         jPanel_North.add(jMenuBar, BorderLayout.NORTH);
639         JMenu mFile=new JMenu("File");
640         JMenuItem mFile_Close=new JMenuItem("Close");
641         mFile.add(mFile_Close);
642         mFile_Close.addActionListener(new ActionListener() {
643             @Override
644             public void actionPerformed(ActionEvent e) {
645                 dispose();
646             }
647         });
648         jMenuBar.add(mFile);
649         //Engine Manage
650         JMenu mEngineManage=new JMenu("Engine");
```

```
651 JMenuItem startODE=new JMenuItem("Start ODE");
652 JMenuItem stopODE=new JMenuItem("Stop ODE");
653 mEngineManage.add(startODE);
654 mEngineManage.add(stopODE);
655 jMenuBar.add(mEngineManage);
656 JMenu mRun=new JMenu("Run");
657 JMenuItem runProcess=new JMenuItem("Run Process");
658 runProcess.addActionListener(new ActionListener() {
659     @Override
660     public void actionPerformed(ActionEvent e) {
661         RunProcess r=new RunProcess();
662     }
663 });
664 mRun.add(runProcess);
665 jMenuBar.add(mRun);
666 JMenu m_Tools=new JMenu("Tools");
667 JMenuItem businessManage=new JMenuItem("Business Manage");
668 businessManage.addActionListener(new ActionListener() {
669     @Override
670     public void actionPerformed(ActionEvent e) {
671         BusinessManage.openDefaultBrowser();
672     }
673 });
674 m_Tools.add(businessManage);
675 jMenuBar.add(m_Tools);
676 JMenu mHelp=new JMenu("Help");
677 JMenuItem metamodel=new JMenuItem("FM MetaModel");
678 metamodel.addActionListener(new ActionListener() {
679     @Override
680     public void actionPerformed(ActionEvent e) {
681         FeatureModelMatamodel fmm=new FeatureModelMatamodel();
682     }
683 });
684 mHelp.add(metamodel);
685 jMenuBar.add(mHelp);
686 final JPanel jPanel_Center=new JPanel(new BorderLayout());
687 JTabbedPane jTabbedPane1=new JTabbedPane();
688 jPanel_Center.add(jTabbedPane1);
689 final JPanel jPanel_Visual_Customizing=new JPanel(new BorderLayout());
690 jTabbedPane1.add("Visual Customizing",jPanel_Visual_Customizing);
691 JPanel jPanel_Visual_Customizing_North=new JPanel(null);
692 jPanel_Visual_Customizing_North.setBorder(new
693 TitledBorder(UIManager.getBorder("TitledBorder.border"), "Loading File",
694 TitledBorder.CENTER, TitledBorder.TOP, null, null));
695 jPanel_Visual_Customizing_North.setPreferredSize(new Dimension(0,105));
696 jPanel_Visual_Customizing.add(jPanel_Visual_Customizing_North,BorderLayout.NORTH);
697 JLabel jLabel_1=new JLabel("Select FM File:");
698 jLabel_1.setBounds(8,20,100,20);
699 jPanel_Visual_Customizing_North.add(jLabel_1);
700 final JTextField jt1=new JTextField();
```

```

701         jt1.setBounds(135,20,290,20);
702         jPanel_Visual_Customizing_North.add(jt1);
703         MyButton myButton_Select_FM_File=new MyButton("Browse",430,20,80,20);
704         jPanel_Visual_Customizing_North.add(myButton_Select_FM_File);
705         myButton_Select_FM_File.addActionListener(new ActionListener() {
706             @Override
707             public void actionPerformed(ActionEvent e) {
708                 JFileChooser fcDlg = new JFileChooser();
709                 fcDlg.setDialogTitle("Please choose file:");
710                 FileNameExtensionFilter filter = new
711 FileNameExtensionFilter("file(*.scfeaturemodel)", "scfeaturemodel");
712                 fcDlg.setFileFilter(filter);
713                 int returnVal = fcDlg.showOpenDialog(null);
714                 if (returnVal == JFileChooser.APPROVE_OPTION) {
715                     String filepath = fcDlg.getSelectedFile().getPath();
716                     url_FM_FilePath=filepath;
717                     jt1.setText(url_FM_FilePath);
718                     jTextArea_Loger_Info.append("Start!\r\n");
719                     jTextArea_Loger_Info.append("The location of Feature Model
720 file!" + filepath + "\r\n");
721                     File f=new File(url_FM_FilePath);
722                     System.out.println(filepath);
723                 }
724             }
725         });
726         MyButton myButton_Cancel1=new MyButton("Cancel",515,20,80,20);
727         jPanel_Visual_Customizing_North.add(myButton_Cancel1);
728         myButton_Cancel1.addActionListener(new CancelActionListener(jt1));
729         JLabel jLabel_2=new JLabel("Select Save Location:");
730         jLabel_2.setBounds(8,50,130,20);
731         jPanel_Visual_Customizing_North.add(jLabel_2);
732         final JTextField jt2=new JTextField();
733         jt2.setBounds(135,50,290,20);
734         url_FM_Save_Location=jt2.getText();
735         jPanel_Visual_Customizing_North.add(jt2);
736         MyButton myButton_Select_SaveLocation=new MyButton("Browse",430,50,80,20);
737         jPanel_Visual_Customizing_North.add(myButton_Select_SaveLocation);
738         myButton_Select_SaveLocation.addActionListener(new ActionListener() {
739             public void actionPerformed(ActionEvent e) {
740                 JFileChooser fcDlg = new JFileChooser();
741                 fcDlg.setDialogTitle("Please select the location where the feature model
742 picture is saved");
743                 fcDlg.setFileSelectionMode(JFileChooser.DIRECTORIES_ONLY);
744                 int returnVal = fcDlg.showOpenDialog(null);
745                 if (returnVal == JFileChooser.APPROVE_OPTION) {
746                     String filepath = fcDlg.getSelectedFile().getPath();
747                     url_FM_Save_Location=filepath;
748                     jt2.setText(url_FM_Save_Location);
749                     jTextArea_Loger_Info.append("The location of
750 diagram!" + filepath + "\r\n");

```

```

751         }
752     }
753 });
754 MyButton myButton_Cancel2=new MyButton("Cancel",515,50,80,20);
755 jPanel_Visual_Customizing_North.add(myButton_Cancel2);
756 myButton_Cancel2.addActionListener(new CancelActionListener(jt2));
757 JLabel jLabel_3=new JLabel("Input Diagram Name:");
758 jLabel_3.setBounds(8,80,130,20);
759 jPanel_Visual_Customizing_North.add(jLabel_3);
760 final JTextField jt3=new JTextField();
761 jt3.setBounds(135,80,290,20);
762 jPanel_Visual_Customizing_North.add(jt3);
763 MyButton myButton_Create_Diagram=new MyButton("Create",430,80,80,20);
764 jPanel_Visual_Customizing_North.add(myButton_Create_Diagram);
765 myButton_Create_Diagram.addActionListener(new ActionListener() {
766     @Override
767     public void actionPerformed(ActionEvent e) {
768         try {
769             Read_FM_XMI rfx=new Read_FM_XMI(url_FM_FilePath);
770             jTextArea_FM_XMI.append(rfx.result);
771         } catch (IOException e1) {
772             e1.printStackTrace();
773         }
774         url_FM_Save_Location=jt2.getText();
775         FMDiagram_Name=jt3.getText();
776         try {
777             DrawFMDiagram drawFMDiagram=new
778 DrawFMDiagram(url_FM_FilePath,url_FM_Save_Location,FMDiagram_Name);
779         } catch (DocumentException e1) {
780             e1.printStackTrace();
781         } catch (IOException e1) {
782             e1.printStackTrace();
783         }
784         JPanel jPanel_FM_VisualTree=new JPanel(new BorderLayout());
785         jPanel_FM_VisualTree.setBorder(new
786 TitledBorder(UIManager.getBorder("TitledBorder.border"), "Visual Tree",
787 TitledBorder.CENTER, TitledBorder.TOP, null, null));
788         jPanel_FM_VisualTree.setPreferredSize(new Dimension(0,415));
789 jPanel_Visual_Customizing.add(jPanel_FM_VisualTree,BorderLayout.SOUTH);
790         try {
791             fmTree=new FMTreeJPanel(url_FM_FilePath);
792             JPanel fmTreeJPanel=fmTree.jPanel;
793             System.out.println("Selected_String"+Selected_String);
794             fmTreeJPanel.setBorder(new
795 TitledBorder(UIManager.getBorder("TitledBorder.border"), "Configurable FM Tree",
796 TitledBorder.CENTER, TitledBorder.TOP, null, null));
797             fmTreeJPanel.setPreferredSize(new Dimension(250,375));
798             jPanel_FM_VisualTree.add(fmTreeJPanel, BorderLayout.WEST);
799             jPanel_FM_VisualTree.setVisible(true);
800             jPanel_Center.repaint();

```



```

801         } catch (IOException e1) {
802             e1.printStackTrace();
803         } catch (AWTException e1) {
804             e1.printStackTrace();
805         } catch (DocumentException e1) {
806             e1.printStackTrace();
807         }
808         JPanel jPanel_Configuration_Info=new JPanel(null);
809         jPanel_Configuration_Info.setBorder(new
810 TitledBorder(UIManager.getBorder("TitledBorder.border"), "Configuration",
811 TitledBorder.CENTER, TitledBorder.TOP, null, null));
812         jPanel_Configuration_Info.setPreferredSize(new Dimension(250,415));
813         JPanel jPanel_Configuration_Info_Table=new JPanel(null);
814         jPanel_Configuration_Info_Table.setBorder(new
815 TitledBorder(UIManager.getBorder("TitledBorder.border"), "Constraints Table",
816 TitledBorder.LEFT, TitledBorder.TOP, null, null));
817         jPanel_Configuration_Info_Table.setBounds(20,25,210,120);
818         jPanel_Configuration_Info.add(jPanel_Configuration_Info_Table);
819         try {
820             ConstraintTable ct=new ConstraintTable(url_FM_FilePath);
821             JScrollPane jScrollPane_Constraint_Table=ct.jtableJScrollPane;
822             jScrollPane_Constraint_Table.setBounds(5,20,200,90);
823             jPanel_Configuration_Info_Table.add(jScrollPane_Constraint_Table);
824         } catch (DocumentException e1) {
825             e1.printStackTrace();
826         }
827         JPanel jPanel_jList_Configuration_Info=new JPanel(null);
828         jPanel_jList_Configuration_Info.setBorder(new
829 TitledBorder(UIManager.getBorder("TitledBorder.border"), "FM Configuration Info",
830 TitledBorder.LEFT, TitledBorder.TOP, null, null));
831         jPanel_jList_Configuration_Info.setBounds(20,155,210,230);
832         jPanel_Configuration_Info.add(jPanel_jList_Configuration_Info);
833         final DefaultListModel dlm=new DefaultListModel();
834         final JList jList = new JList(dlm);
835         jList.setBounds(5,20,200,205);
836         jPanel_jList_Configuration_Info.add(jList);
837         final String[] jlist_selected_info = new String[1];
838         jList.addListSelectionListener(new ListSelectionListener(){
839             @Override
840             public void valueChanged(ListSelectionEvent e) {
841                 if(!jList.getValueIsAdjusting()){ //设置只有释放鼠标时才触发
842                     System.out.println(jList.getSelectedValue());
843                     jlist_selected_info[0] = (String) jList.getSelectedValue();
844                 }
845             }
846         });
847         jPanel_FM_VisualTree.add(jPanel_Configuration_Info, BorderLayout.EAST);
848         JPanel jPanel_JButton=new JPanel(null);
849         jPanel_FM_VisualTree.add(jPanel_JButton, BorderLayout.CENTER);
850         MyButton myButton_Add=new MyButton("Add", 7, 100, 80, 20);

```

```

851         myButton_Add.addActionListener(new ActionListener() {
852             @Override
853             public void actionPerformed(ActionEvent e) {
854                 Selected_String=fmTree.selected_String;
855                 System.out.println("Selected_String"+Selected_String);
856                 dlm.addElement(Selected_String);
857                 arrayList.add(Selected_String);
858             }
859         });
860         MyButton myButton_Remove=new MyButton("Remove",7,150,80,20);
861         myButton_Remove.addActionListener(new ActionListener() {
862             public void actionPerformed(ActionEvent e) {
863                 arrayList.remove(jlist_selected_info[0]);
864                 dlm.removeElement(jlist_selected_info[0]);
865             }
866         });
867         MyButton myButton_Validate=new MyButton("Validate",7,200,80,20);
868         myButton_Validate.addActionListener(new ActionListener() {
869             @Override
870             public void actionPerformed(ActionEvent e) {
871                 try {
872                     Validate v=new Validate(url_FM_FilePath,arrayList);
873                     if(v.flag.equals("true")){
874                         MyDialog_Confirm myDialog_confirm=new
875 MyDialog_Confirm("Confirm to FM constraints!");
876                     }else if(v.flag.equals("false")){
877                         MyDialog_Error myDialog_error=new
878 MyDialog_Error("Error!");
879                     }
880                 } catch (DocumentException e1) {
881                     e1.printStackTrace();
882                 }
883             }
884         });
885         jPanel_JButton.add(myButton_Add);
886         jPanel_JButton.add(myButton_Remove);
887         jPanel_JButton.add(myButton_Validate);
888         jPanel_Visual_Customizing.add(jPanel_FM_VisualTree,BorderLayout.SOUTH);
889         jPanel_Center.repaint();
890     }
891 });
892 MyButton myButton_Cancel3=new MyButton("Cancel",515,80,80,20);
893 jPanel_Visual_Customizing_North.add(myButton_Cancel3);
894 myButton_Cancel3.addActionListener(new CancelActionListener(jt3));
895 final JPanel jPanel_Model_Transformation=new JPanel(new BorderLayout());
896 jTabbedPane1.add("Standard VxBPEL",jPanel_Model_Transformation);
897 JPanel jPanel_Model_Transformation_North=new JPanel(null);
898 jPanel_Model_Transformation_North.setBorder(new
899 TitledBorder(UIManager.getBorder("TitledBorder.border"), "Loading File",
900 TitledBorder.CENTER, TitledBorder.TOP, null, null));

```

```

901     jPanel_Model_Transformation_North.setPreferredSize(new Dimension(0,140));
902     jPanel_Model_Transformation.add(
903     jPanel_Model_Transformation_North, BorderLayout.NORTH);
904     JLabel jLabel_2_1=new JLabel("Select File:");
905     jLabel_2_1.setBounds(8,20,100,20);
906     jPanel_Model_Transformation_North.add(jLabel_2_1);
907     final JTextField jt_2_1=new JTextField();
908     jt_2_1.setBounds(135,20,290,20);
909     jPanel_Model_Transformation_North.add(jt_2_1);
910     MyButton myButton_Select_FM_File_2_1=new MyButton("Browse",430,20,80,20);
911     jPanel_Model_Transformation_North.add(myButton_Select_FM_File_2_1);
912     myButton_Select_FM_File_2_1.addActionListener(new ActionListener() {
913         @Override
914         public void actionPerformed(ActionEvent e) {
915             JFileChooser fcDlg = new JFileChooser();
916             fcDlg.setDialogTitle("Please choose file:");
917             FileNameExtensionFilter filter = new
918 FileNameExtensionFilter("file(*.xml,*vxbpel)", "xml", "vxbpel");
919             fcDlg.setFileFilter(filter);
920             int returnVal = fcDlg.showOpenDialog(null);
921             if (returnVal == JFileChooser.APPROVE_OPTION) {
922                 String filepath = fcDlg.getSelectedFile().getPath();
923                 url_FM_FilePath_2_1=filepath;
924                 jt_2_1.setText(url_FM_FilePath_2_1);
925                 jTextArea_Lozer_Info.append("The location of Feature Model
926 file!" +filepath+"\r\n");
927                 File f=new File(url_FM_FilePath_2_1);
928                 System.out.println(filepath);
929             }
930         }
931     });
932     MyButton myButton_Cancel_2_1=new MyButton("Cancel",515,20,80,20);
933     jPanel_Model_Transformation_North.add(myButton_Cancel_2_1);
934     myButton_Cancel_2_1.addActionListener(new CancelActionListener(jt_2_1));
935     JLabel jLabel_2_2=new JLabel("Select Save Location:");
936     jLabel_2_2.setBounds(8,50,130,20);
937     jPanel_Model_Transformation_North.add(jLabel_2_2);
938     final JTextField jt_2_2=new JTextField();
939     jt_2_2.setBounds(135,50,290,20);
940     jPanel_Model_Transformation_North.add(jt_2_2);
941     MyButton myButton_Select_SaveLocation_2_2=new
942 MyButton("Browse",430,50,80,20);
943     jPanel_Model_Transformation_North.add(myButton_Select_SaveLocation_2_2);
944     myButton_Select_SaveLocation_2_2.addActionListener(new ActionListener() {
945         @Override
946         public void actionPerformed(ActionEvent e) {
947             JFileChooser fcDlg = new JFileChooser();
948             fcDlg.setDialogTitle("Please select the location where the feature model
949 picture is saved");
950             fcDlg.setFileSelectionMode(JFileChooser.DIRECTORIES_ONLY);

```

```

951         int returnVal = fcDlg.showOpenDialog(null);
952         if (returnVal == JFileChooser.APPROVE_OPTION) {
953             String filepath = fcDlg.getSelectedFile().getPath();
954             url_VxBPEL_Save_Location_2_2=filepath;
955             jt_2_2.setText(url_VxBPEL_Save_Location_2_2);
956             jTextArea_Loget_Info.append("The location of
957 VxBPEL!" + filepath + "\r\n");
958         }
959     }
960 });
961 MyButton myButton_Cancel_2_2=new MyButton("Cancel",515,50,80,20);
962 jPanel_Model_Transformation_North.add(myButton_Cancel_2_2);
963 myButton_Cancel_2_2.addActionListener(new CancelActionListener(jt_2_2));
964 JLabel jLabel_2_3=new JLabel("Input VxBPEL Name:");
965 jLabel_2_3.setBounds(8,80,130,20);
966 jPanel_Model_Transformation_North.add(jLabel_2_3);
967 final JTextField jt_2_3=new JTextField();
968 jt_2_3.setBounds(135,80,290,20);
969 jPanel_Model_Transformation_North.add(jt_2_3);
970 MyButton myButton_Model_Convert=new MyButton("Convert",430,80,80,20);
971 jPanel_Model_Transformation_North.add(myButton_Model_Convert);
972 myButton_Model_Convert.addActionListener(new ActionListener() {
973     @Override
974     public void actionPerformed(ActionEvent e) {
975         vxbpel_name=jt_2_3.getText();
976         try {
977             ModelTran mt=new
978 ModelTran(url_FM_FilePath_2_1,url_VxBPEL_Save_Location_2_2,vxbpel_name);
979         } catch (DocumentException e1) {
980             e1.printStackTrace();
981         } catch (IOException e1) {
982             e1.printStackTrace();
983         }
984     }
985 });
986 MyButton myButton_Cancel_2_3=new MyButton("Cancel",515,80,80,20);
987 jPanel_Model_Transformation_North.add(myButton_Cancel_2_3);
988 myButton_Cancel_2_3.addActionListener(new CancelActionListener(jt_2_3));
989 JLabel jLabel_2_4=new JLabel("Parse Variability Info:");
990 jLabel_2_4.setBounds(8,110,130,20);
991 jPanel_Model_Transformation_North.add(jLabel_2_4);
992 final JTextField jt_2_4=new JTextField();
993 jt_2_4.setBounds(135,110,290,20);
994 jPanel_Model_Transformation_North.add(jt_2_4);
995 MyButton myButton_Browse_2_4=new MyButton("Browse",430,110,80,20);
996 jPanel_Model_Transformation_North.add(myButton_Browse_2_4);
997 myButton_Browse_2_4.addActionListener(new ActionListener() {
998     @Override
999     public void actionPerformed(ActionEvent e) {
1000         JFileChooser fcDlg = new JFileChooser();

```

```

1001         fcDlg.setDialogTitle("Please choose file:");
1002         FileNameExtensionFilter filter =
1003         new FileNameExtensionFilter("file(*.bpel)", "bpel");
1004         fcDlg.setFileFilter(filter);
1005         int returnVal = fcDlg.showOpenDialog(null);
1006         if (returnVal == JFileChooser.APPROVE_OPTION) {
1007             String filepath = fcDlg.getSelectedFile().getPath();
1008             vxbpel_file_location=filepath;
1009             jt_2_4.setText(vxbpel_file_location);
1010             jTextArea_Loger_Info.append("Start!\r\n");
1011             jTextArea_Loger_Info.append("The location of vxbpel
1012 file!" + filepath + "\r\n");
1013             System.out.println(filepath);
1014         }
1015     }
1016 });
1017 MyButton myButton_Parse_2_4=new MyButton("Parse",515,110,80,20);
1018 jPanel_Model_Transformation_North.add(myButton_Parse_2_4);
1019 myButton_Parse_2_4.addActionListener(new ActionListener() {
1020     @Override
1021     public void actionPerformed(ActionEvent e) {
1022         JPanel jPanel_VxBPEL=new JPanel(new BorderLayout());
1023         jPanel_VxBPEL.setBorder(new
1024 TitledBorder(UIManager.getBorder("TitledBorder.border"), "VxBPEL",
1025 TitledBorder.CENTER, TitledBorder.TOP, null, null));
1026         jPanel_VxBPEL.setPreferredSize(new Dimension(0,380));
1027         jPanel_Model_Transformation.add(jPanel_VxBPEL,BorderLayout.SOUTH);
1028         try {
1029             ProcessTree pt=new ProcessTree(vxbpel_file_location);
1030             JPanel jPanel_vxbpel_Tree=pt.jPanel;
1031             jPanel_vxbpel_Tree.setBorder(new
1032 TitledBorder(UIManager.getBorder("TitledBorder.border"), "Process Tree",
1033 TitledBorder.CENTER, TitledBorder.TOP, null, null));
1034             jPanel_vxbpel_Tree.setPreferredSize(new Dimension(250,325));
1035             jPanel_VxBPEL.add(jPanel_vxbpel_Tree,BorderLayout.WEST);
1036         } catch (DocumentException e1) {
1037             e1.printStackTrace();
1038         }
1039         JPanel jPanel_vxbpel_configuration_info=new JPanel(null);
1040         jPanel_vxbpel_configuration_info.setBorder(new
1041 TitledBorder(UIManager.getBorder("TitledBorder.border"), "VxBPEL Configuration Info",
1042 TitledBorder.CENTER, TitledBorder.TOP, null, null));
1043         jPanel_vxbpel_configuration_info.setPreferredSize(new
1044 Dimension(250,325));
1045 jPanel_VxBPEL.add(jPanel_vxbpel_configuration_info,BorderLayout.EAST);
1046         JPanel jPanel_fm_conf_info=new JPanel(null);
1047         jPanel_fm_conf_info.setBounds(10,20,230,140);
1048         jPanel_fm_conf_info.setBorder(new
1049 TitledBorder(UIManager.getBorder("TitledBorder.border"), "FM Configuration Info",
1050 TitledBorder.LEFT, TitledBorder.TOP, null, null));

```

```

1051         JTextArea jTextArea_fm_conf_info=new JTextArea();
1052         JScrollPane jScrollPane_fm_conf_info=new
1053 JScrollPane(jTextArea_fm_conf_info);
1054         jScrollPane_fm_conf_info.setBounds(5,20,220,110);
1055         jPanel_fm_conf_info.add(jScrollPane_fm_conf_info);
1056         jPanel_vxbpel_configuration_info.add(jPanel_fm_conf_info);
1057         if(arrayList.size()>0){
1058             for(int i=0;i<arrayList.size();i++){
1059                 jTextArea_fm_conf_info.append(arrayList.get(i)+"\r\n");
1060             }
1061         }
1062         JPanel jPanel_Match_vp_variant=new JPanel(null);
1063         jPanel_Match_vp_variant.setBounds(10,170,230,180);
1064         jPanel_Match_vp_variant.setBorder(new
1065 TitledBorder(UIManager.getBorder("TitledBorder.border"), "VariationPoint && Variant
1066 Conf", TitledBorder.LEFT, TitledBorder.TOP, null, null));
1067         final JTextArea jTextArea_user_conf=new JTextArea();
1068         JScrollPane jScrollPane_user_conf=new JScrollPane(jTextArea_user_conf);
1069         jScrollPane_user_conf.setBounds(5,20,220,155);
1070         jPanel_Match_vp_variant.add(jScrollPane_user_conf);
1071         jPanel_vxbpel_configuration_info.add(jPanel_Match_vp_variant);
1072         JPanel jPanel_JButton2=new JPanel(null);
1073         jPanel_VxBPEL.add(jPanel_JButton2,BorderLayout.CENTER);
1074         MyButton myButton_Match_VP=new MyButton("Match",7,120,80,20);
1075         myButton_Match_VP.addActionListener(new ActionListener() {
1076             @Override
1077             public void actionPerformed(ActionEvent e) {
1078                 arrayLists_VP=new ArrayList<ArrayList>();
1079                 for(int i=0;i<arrayList.size();i++){
1080                     try {
1081                         FindVP findVP=new
1082 FindVP(vxbpel_file_location,arrayList.get(i));
1083 jTextArea_user_conf.append("VariationPoint->"+findVP.flag1+"\r\n");
1084 jTextArea_user_conf.append("
1085 Variant->"+findVP.flag2+"\r\n");
1086                     ArrayList<String> a=new ArrayList<String>();
1087                     a.add(findVP.flag1);
1088                     a.add(findVP.flag2);
1089                     arrayLists_VP.add(a);
1090                 } catch (DocumentException e1) {
1091                     e1.printStackTrace();
1092                 }
1093             }
1094         }
1095     });
1096     jPanel_JButton2.add(myButton_Match_VP);
1097     jPanel_Center.repaint();//很重要
1098 }
1099 });
1100 final JPanel jPanel_UCCF=new JPanel(new BorderLayout());//可视化定制

```

```

1101     jTabbedPane1.add("Create UCCF",jPanel_UCCF);
1102     JPanel jPanel_UCCF_North=new JPanel(null);
1103     jPanel_UCCF_North.setBorder(new
1104 TitledBorder(UIManager.getBorder("TitledBorder.border"), "Create UCCF",
1105 TitledBorder.CENTER, TitledBorder.TOP, null, null));
1106     jPanel_UCCF_North.setPreferredSize(new Dimension(0,75));
1107     jPanel_UCCF.add(jPanel_UCCF_North,BorderLayout.NORTH);
1108     JLabel jLabel_3_1=new JLabel("Select Save Location:");
1109     jLabel_3_1.setBounds(8,20,140,20);
1110     jPanel_UCCF_North.add(jLabel_3_1);
1111     final JTextField jt_3_1=new JTextField();
1112     jt_3_1.setBounds(135,20,290,20);
1113     jPanel_UCCF_North.add(jt_3_1);
1114     MyButton myButton_Select_3_1=new MyButton("Browse",430,20,80,20);
1115     jPanel_UCCF_North.add(myButton_Select_3_1);
1116     myButton_Select_3_1.addActionListener(new ActionListener() {
1117         @Override
1118         public void actionPerformed(ActionEvent e) {
1119             JFileChooser fcDlg = new JFileChooser();
1120             fcDlg.setDialogTitle("Please select the location where the UCCF is saved");
1121             fcDlg.setFileSelectionMode(JFileChooser.DIRECTORIES_ONLY);
1122             int returnVal = fcDlg.showOpenDialog(null);
1123             if (returnVal == JFileChooser.APPROVE_OPTION) {
1124                 String filepath = fcDlg.getSelectedFile().getPath();
1125                 UCCF_Save_Location=filepath;
1126                 jt_3_1.setText(UCCF_Save_Location);
1127                 jTextArea_Loget_Info.append(
1128 "The save location of UCCF!" +filepath+"\r\n");
1129             }
1130         }
1131     });
1132     MyButton myButton_Cancel_3_1=new MyButton("Cancel",515,20,80,20);
1133     jPanel_UCCF_North.add(myButton_Cancel_3_1);
1134     myButton_Cancel_3_1.addActionListener(new CancelActionListener(jt_3_1));
1135     JLabel jLabel_3_2=new JLabel("Input UCCF File Name:");
1136     jLabel_3_2.setBounds(8,50,140,20);
1137     jPanel_UCCF_North.add(jLabel_3_2);
1138     final JTextField jt_3_2=new JTextField();
1139     jt_3_2.setBounds(135,50,290,20);
1140     jPanel_UCCF_North.add(jt_3_2);
1141     MyButton myButton_create_UCCF=new MyButton("UCCF",430,50,80,20);
1142     jPanel_UCCF_North.add(myButton_create_UCCF);
1143     myButton_create_UCCF.addActionListener(new ActionListener() {
1144         @Override
1145         public void actionPerformed(ActionEvent e) {
1146             JPanel jPanel_UCCF_South=new JPanel(null);
1147             jPanel_UCCF_South.setPreferredSize(new Dimension(0,440));
1148             jPanel_UCCF_South.setBorder(new
1149 TitledBorder(UIManager.getBorder("TitledBorder.border"), "UCCF Info",
1150 TitledBorder.CENTER, TitledBorder.TOP, null, null));

```

```

1151         JTextArea jTextArea_UCCF_Info=new JTextArea();
1152         jTextArea_UCCF_Info.setFont(jTextArea_Font);
1153         JScrollPane jScrollPane_UCCF_Info=new
1154 JScrollPane(jTextArea_UCCF_Info);
1155         jScrollPane_UCCF_Info.setBounds(5,20,595,410);
1156         jPanel_UCCF_South.add(jScrollPane_UCCF_Info);
1157         jPanel_UCCF.add(jPanel_UCCF_South, BorderLayout.SOUTH);
1158         UCCF_File_Name=jt_3_2.getText();
1159         String
1160 all_uccf_location=UCCF_Save_Location+"\\ "+UCCF_File_Name+".uccf";
1161         UccfInof u=new UccfInof(arrayLists_VP);
1162         try {
1163             CreateUCCF c=new CreateUCCF(all_uccf_location,u.s_info);
1164         } catch (IOException e1) {
1165             e1.printStackTrace();
1166         }
1167         File f=new File(all_uccf_location);
1168         try {
1169             Read_UCCF ru=new Read_UCCF(f,arrayLists_VP.size()+2);
1170             jTextArea_UCCF_Info.append(ru.result);
1171         } catch (IOException e1) {
1172             e1.printStackTrace();
1173         }
1174         jPanel_Center.repaint();
1175     }
1176 });
1177 MyButton myButton_Cancel_3_2=new MyButton("Cancel",515,50,80,20);
1178 jPanel_UCCF_North.add(myButton_Cancel_3_2);
1179 myButton_Cancel_3_2.addActionListener(new CancelActionListener(jt_3_2));
1180 JPanel jPanel_South=new JPanel(new BorderLayout());
1181 jPanel_South.setPreferredSize(new Dimension(0,140));
1182 JTabbedPane jTabbedPane2=new JTabbedPane();
1183 jPanel_South.add(jTabbedPane2, BorderLayout.NORTH);
1184 JPanel jPanel_Lozer_Info=new JPanel(new BorderLayout());
1185 jTabbedPane2.add("Logger Info",jPanel_Lozer_Info);
1186 jTextArea_Lozer_Info=new JTextArea();
1187 jTextArea_Lozer_Info.setFont(jTextArea_Font);
1188 JScrollPane jScrollPane1=new JScrollPane(jTextArea_Lozer_Info);
1189 jScrollPane1.setPreferredSize(new Dimension(0,110));
1190 jPanel_Lozer_Info.add(jScrollPane1, BorderLayout.NORTH);
1191 JPanel jPanel_FM_XMI=new JPanel(new BorderLayout());
1192 jTabbedPane2.add("FM XMI",jPanel_FM_XMI);
1193 jTextArea_FM_XMI=new JTextArea();
1194 jTextArea_FM_XMI.setFont(jTextArea_Font);
1195 JScrollPane jScrollPane2=new JScrollPane(jTextArea_FM_XMI);
1196 jScrollPane2.setPreferredSize(new Dimension(0,110));
1197 jPanel_FM_XMI.add(jScrollPane2, BorderLayout.NORTH);
1198 this.add(jPanel_North, BorderLayout.NORTH);
1199 this.add(jPanel_Center, BorderLayout.CENTER);
1200 this.add(jPanel_South, BorderLayout.SOUTH);

```



```
1201         this.setVisible(true);
1202         this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
1203         this.setLocationRelativeTo(null);
1204     }
1205     public static void main(String[] args) {
1206         Main_ATL m=new Main_ATL();
1207     }
1208 }
1209 package com.ustb.cn;
1210 import java.io.IOException;
1211 import java.util.ArrayList;
1212 import java.util.Iterator;
1213 import java.util.List;
1214 import java.util.Map;
1215 import java.util.Set;
1216 import javax.servlet.http.HttpServlet;
1217 import javax.servlet.http.HttpServletRequest;
1218 import javax.servlet.http.HttpServletResponse;
1219 import javax.servlet.http.HttpSession;
1220 public class MRIter extends HttpServlet{
1221     private void autoMethodIter(HttpServletRequest request,
1222         HttpServletResponse response, HttpSession session,
1223         String operationName, WsdlOperationFormat opFormat,
1224         List<MetamorphicRelation> mrList) throws IOException {
1225         String numStr = request.getParameter("num_text");
1226         int num = Integer.parseInt(numStr);
1227         if (((String) session.getAttribute("operationName"))
1228             .equals("transfer") && num > 0) {
1229             int mrID = 0;
1230             int id = 1;
1231             List<TCTransfer> tcsTransferList = null;
1232             List<TCTransferF> tcsTransferFList= null;
1233             List<List<String>> tcsListS = null;// 原始测试用例
1234             List<List<String>> tcsListF = null;//衍生测试用例
1235             if (session.getAttribute("tcsTransferList") == null) {
1236                 System.out.println("第一个 tc");
1237                 tcsTransferList = new ArrayList<TCTransfer>();
1238                 tcsTransferFList = new ArrayList<TCTransferF>();
1239                 tcsListS = new ArrayList<List<String>>();
1240                 tcsListF = new ArrayList<List<String>>();
1241             } else {
1242                 tcsTransferList = (List<TCTransfer>) session
1243                     .getAttribute("tcsTransferList");
1244                 tcsTransferFList = (List<TCTransferF>) session
1245                     .getAttribute("tcsTransferFList");
1246                 tcsListS = (List<List<String>>) session.getAttribute("tcsList");
1247                 // TODO
1248                 tcsListF = (List<List<String>>) session
1249                     .getAttribute("tcsFList");
1250                 id = tcsTransferList.size() + 1;
```

```

1251     }
1252     XmlTestCaseGenerator tcGen = new XmlTestCaseGenerator();
1253     for (int mrCount = 0; mrCount < num; mrCount++) {
1254         TCTransfer tempTC=null;
1255         TCTransferF tempTCF=null;
1256         mrID=0;
1257         Iterator<MetamorphicRelation> iterMR = mrList.iterator();
1258         while (iterMR.hasNext()) {
1259             mrID++;
1260             MetamorphicRelation mr = iterMR.next();
1261             if (mrID == 1 && mrCount == 0) {
1262                 XmlTestCase newTC = tcGen.genWithLimitation(
1263                     operationName, opFormat.getInput().format, mr
1264                         .getRelationOfInput(), mrID);
1265                 Set<Map.Entry<XmlVariable, String>> entry = newTC
1266                     .getInput().entrySet();
1267                 Iterator<Map.Entry<XmlVariable, String>> iter = entry
1268                     .iterator();
1269                 Map.Entry<XmlVariable, String> myentry = null;
1270                 int i = 0;
1271                 String values[] = new String[entry.size()];
1272                 while (iter.hasNext()) {
1273                     myentry = iter.next();
1274                     values[i] = myentry.getValue();
1275                     i++;
1276                 }
1277                 TCTransfer tcTransfer=new TCTransfer(id, mrID, values[0],
1278                     values[1], values[2], values[3]);
1279                 tempTC=new TCTransfer(id, mrID, values[0],
1280                     values[1], values[2], values[3]);
1281                 tcsTransferList.add(tcTransfer);
1282                 TCTransferF
1283                 tcTransferF=TCTransferF.generateFollowTC(tcTransfer,opFormat,mr);
1284                 tempTCF=tcTransferF;
1285                 tcsTransferFList.add(tcTransferF);
1286             }else if(mrID==1){
1287                 TCTransferF tcTransferF=tcsTransferFList.get(mrCount-1);
1288                 TCTransfer tcTransfer=tcTransferF.transferToS();
1289                 tcTransfer.setId(id);
1290                 tcTransfer.setMrID(mrID);
1291                 tempTC = new TCTransfer(tcTransfer.getId(), tcTransfer
1292                     .getMrID(), tcTransfer.getAccountFrom(),
1293                     tcTransfer.getAccountTo(),
1294                     tcTransfer.getMode(), tcTransfer.getAmount());
1295                 tcsTransferList.add(tcTransfer);
1296                 tcTransferF=TCTransferF.generateFollowTC(tcTransfer,opFormat,mr);
1297                 tempTCF=tcTransferF;
1298                 tcsTransferFList.add(tcTransferF);
1299             }else {
1300                 tempTC.setMrID(mrID);

```

```

1301         tempTC.setId(id);
1302         TCTransfer tcTransfer = new TCTransfer(tempTC.getId(),
1303             tempTC.getMrID(), tempTC.getAccountFrom(),
1304             tempTC.getAccountTo(), tempTC.getMode(), tempTC
1305                 .getAmount());
1306         tcsTransferList.add(tcTransfer);
1307         TCTransferF tcTransferF = TCTransferF.generateFollowTC(
1308             tcTransfer, opFormat, mr);
1309         tempTCF = tcTransferF;
1310         tcsTransferFList.add(tcTransferF);
1311     }
1312     List<String> tcList = new ArrayList<String>();
1313     tcList.add(String.valueOf(id));
1314     tcList.add(String.valueOf(mrID));
1315     tcList.add(tempTC.getAccountFrom());
1316     tcList.add(tempTC.getAccountTo());
1317     tcList.add(tempTC.getMode());
1318     tcList.add(tempTC.getAmount());
1319     tcsListS.add(tcList);
1320     List<String> tcListF = new ArrayList<String>();
1321     tcListF.add(String.valueOf(tempTCF.getID()));//有待改进
1322     tcListF.add(String.valueOf(mrID));
1323     tcListF.add(tempTCF.getAccountFrom());
1324     tcListF.add(tempTCF.getAccountTo());
1325     tcListF.add(tempTCF.getMode());
1326     tcListF.add(tempTCF.getAmount());
1327     tcListF.add(String.valueOf(tempTCF.getID()));
1328     tcsListF.add(tcListF);
1329     id++;
1330 }
1331 }
1332 session.setAttribute("tcsList", tcsListS);
1333 session.setAttribute("tcsTransferList", tcsTransferList);
1334 session.setAttribute("tcsFList", tcsListF);
1335 session.setAttribute("tcsTransferFList", tcsTransferFList);
1336 SqlUtils sqlUtils=SqlUtils.getInstance();
1337 sqlUtils.insertToTransferS(tcsTransferList);
1338 sqlUtils.insertToTransferF(tcsTransferFList);
1339 response.sendRedirect("4.start_test.jsp");
1340 } else if (((String) session.getAttribute("operationName"))
1341     .equals("matchQuakes") && num > 0) {
1342 } else if (((String) session.getAttribute("operationName"))
1343     .equals("rmbToGbbob") && num > 0) {
1344 } else {
1345     System.out.println("跳转错误!!! ");
1346 }
1347
1348 }
1349 private void autoMethodIter_1(HttpServletRequest request,
1350     HttpServletResponse response, HttpSession session,

```

```

1351         String operationName, WsdlOperationFormat opFormat,
1352         List<MetamorphicRelation> mrList) throws IOException {
1353
1354         String numStr = request.getParameter("num_text");
1355         int num = Integer.parseInt(numStr);
1356         if (((String) session.getAttribute("operationName"))
1357             .equals("transfer") && num > 0) {
1358             int mrID = 0;
1359             int id = 1;
1360             List<TCTransfer> tcsTransferList = null;
1361             List<TCTransferF> tcsTransferFList = null;
1362             List<List<String>> tcsListS = null;
1363             List<List<String>> tcsListF = null;
1364             if (session.getAttribute("tcsTransferList") == null) {
1365                 System.out.println("第一个 tc");
1366                 tcsTransferList = new ArrayList<TCTransfer>();
1367                 tcsTransferFList = new ArrayList<TCTransferF>();
1368                 tcsListS = new ArrayList<List<String>>();
1369                 tcsListF = new ArrayList<List<String>>();
1370             } else {
1371                 tcsTransferList = (List<TCTransfer>) session
1372                     .getAttribute("tcsTransferList");
1373                 tcsTransferFList = new ArrayList<TCTransferF>();
1374                 tcsListS = (List<List<String>>) session.getAttribute("tcsList");
1375                 tcsListF = new ArrayList<List<String>>();
1376                 id = tcsTransferList.size() + 1;
1377             }
1378             XmlTestCaseGenerator tcGen = new XmlTestCaseGenerator();
1379             for (int mrCount = 0; mrCount < num; mrCount++) {
1380                 TCTransfer tempTC=null;
1381                 TCTransferF tempTCF=null;
1382                 mrID=0;
1383                 Iterator<MetamorphicRelation> iterMR = mrList.iterator();
1384                 while (iterMR.hasNext()) {
1385                     mrID++;
1386                     MetamorphicRelation mr = iterMR.next();
1387                     if (mrID == 1 && mrCount == 0) {
1388                         TCTransfer tcTransfer=null;
1389                         if(tcsTransferList.size()!=0){
1390                             tcTransfer=tcsTransferList.get(0);
1391                         } else {
1392                             XmlTestCase newTC = tcGen.genWithLimitation(
1393                                 operationName, opFormat.getInput().format,
1394                                 mr.getRelationOfInput(), mrID);
1395                             Set<Map.Entry<XmlVariable, String>> entry = newTC
1396                                 .getInput().entrySet();
1397                             Iterator<Map.Entry<XmlVariable, String>> iter = entry
1398                                 .iterator();
1399                             Map.Entry<XmlVariable, String> myentry = null;
1400                             int i = 0;

```

```

1401         String values[] = new String[entry.size()];
1402         while (iter.hasNext()) {
1403             myentry = iter.next();
1404             values[i] = myentry.getValue();
1405             i++;
1406         }
1407         tcTransfer = new TCTransfer(id, mrID, values[0],
1408             values[1], values[2], values[3]);
1409         tempTC = new TCTransfer(id, mrID, values[0],
1410             values[1], values[2], values[3]);
1411         tcsTransferList.add(tcTransfer);
1412     }
1413     TCTransferF
1414     tcTransferF=TCTransferF.generateFollowTC(tcTransfer,opFormat,mr);
1415     tempTCF=tcTransferF;
1416     tcsTransferFList.add(tcTransferF);
1417 }else if(mrID==1){
1418     TCTransferF tcTransferF=tcsTransferFList.get(mrCount-1);
1419     TCTransfer tcTransfer=tcTransferF.tranferToS();
1420
1421     tcTransfer.setId(id);
1422     tcTransfer.setMrID(mrID);
1423     tempTC = new TCTransfer(tcTransfer.getId(), tcTransfer
1424         .getMrID(), tcTransfer.getAccountFrom(),
1425         tcTransfer.getAccountTo(),
1426         tcTransfer.getMode(), tcTransfer.getAmount());
1427     tcsTransferList.add(tcTransfer);
1428     tcTransferF=TCTransferF.generateFollowTC(tcTransfer,opFormat,mr);
1429     tempTCF=tcTransferF;
1430     tcsTransferFList.add(tcTransferF);
1431 }else {
1432     tempTC.setMrID(mrID);
1433     tempTC.setId(id);
1434     TCTransfer tcTransfer = new TCTransfer(tempTC.getId(),
1435         tempTC.getMrID(), tempTC.getAccountFrom(),
1436         tempTC.getAccountTo(), tempTC.getMode(), tempTC
1437         .getAmount());
1438     tcsTransferList.add(tcTransfer);
1439     TCTransferF tcTransferF = TCTransferF.generateFollowTC(
1440         tcTransfer, opFormat, mr);
1441     tempTCF = tcTransferF;
1442     tcsTransferFList.add(tcTransferF);
1443 }
1444 List<String> tcList = new ArrayList<String>();
1445 tcListF.add(tempTCF.getAmount());
1446 tcListF.add(String.valueOf(tempTCF.getsID()));
1447 tcsListF.add(tcListF);
1448 tcList.add(String.valueOf(id));
1449 tcList.add(String.valueOf(mrID));
1450 tcList.add(tempTC.getAccountFrom());

```

```

1451         tcList.add(tempTC.getAccountTo());
1452         tcList.add(tempTC.getMode());
1453         tcList.add(tempTC.getAmount());
1454         tcsListS.add(tcList);
1455         List<String> tcListF = new ArrayList<String>();
1456         tcListF.add(String.valueOf(tempTCF.getsID()));//有待改进
1457         tcListF.add(String.valueOf(mrID));
1458         tcListF.add(tempTCF.getAccountFrom());
1459         tcListF.add(tempTCF.getAccountTo());
1460         tcListF.add(tempTCF.getMode());
1461         tcListF.add(tempTCF.getAmount());
1462         tcListF.add(String.valueOf(tempTCF.getsID()));
1463         tcsListF.add(tcListF);
1464         id++;
1465     }
1466 }
1467 session.setAttribute("tcsList", tcsListS);
1468 session.setAttribute("tcsTransferList", tcsTransferList);
1469 session.setAttribute("tcsFList", tcsListF);
1470 session.setAttribute("tcsTransferFList", tcsTransferFList);
1471 SqlUtils sqlUtils=SqlUtils.getInstance();
1472 sqlUtils.insertToTransferS(tcsTransferList);
1473 sqlUtils.insertToTransferF(tcsTransferFList);
1474 response.sendRedirect("4.start_test.jsp");
1475 } else if (((String) session.getAttribute("operationName"))
1476     .equals("matchQuakes") && num > 0) {
1477 } else if (((String) session.getAttribute("operationName"))
1478     .equals("rmbToGbbob") && num > 0) {
1479 } else {
1480     System.out.println("跳转错误!!! ");
1481 }
1482 }
1483 }
1484 package com.ustb.cn;
1485 import com.ustb.cn.BussinessProcess.ProcessTree;
1486 import com.ustb.cn.CreateUCCF.CreateUCCF;
1487 import com.ustb.cn.CreateUCCF.Read_UCCF;
1488 import com.ustb.cn.CreateUCCF.UccfInof;
1489 import com.ustb.cn.DrawFMPic.DrawFMDiagram;
1490 import com.ustb.cn.ModelTransformation.ModelTran;
1491 import com.ustb.cn.ValidateConstraints.Validate;
1492 import org.dom4j.DocumentException;
1493 public class Main_Test2 extends JFrame{
1494     FMTreeJPanel fmTree;
1495     String Selected_String;
1496     Font jTextArea_Font=new Font("Consolas",Font.PLAIN,12);
1497     String url_FM_FilePath;
1498     String url_FM_Save_Location;
1499     String FMDiagram_Name;
1500     JTextArea jTextArea_Logger_Info;

```