package com.ustb.cn;

import com.ustb.cn.BussinessProcess.ProcessTree;

import com.ustb.cn.CreateUCCF.CreateUCCF;

import com.ustb.cn.CreateUCCF.Read\_UCCF;

import com.ustb.cn.CreateUCCF.UccfInof;

import com.ustb.cn.DrawFMPic.DrawFMDiagram;

import com.ustb.cn.ModelTransformation.ModelTran;

import com.ustb.cn.ValidateConstraints.Validate;

import org.dom4j.DocumentException;

import javax.swing.\*;

import javax.swing.border.TitledBorder;

import javax.swing.event.ListSelectionEvent;

import javax.swing.event.ListSelectionListener;

import javax.swing.filechooser.FileNameExtensionFilter;

import java.awt.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

public class Main extends JFrame{

FMTreeJPanel fmTree;

String Selected\_String;

Font jTextArea\_Font=new Font("Consolas",Font.PLAIN,12);

String url\_FM\_FilePath;

String url\_FM\_Save\_Location;

ArrayList<String> arrayList=new ArrayList<String>();

String FMDiagram\_Name;

JTextArea jTextArea\_Loger\_Info;

JTextArea jTextArea\_FM\_XMI;

String url\_FM\_FilePath\_2\_1;

String url\_VxBPEL\_Save\_Location\_2\_2;

String vxbpel\_name;

String vxbpel\_file\_location;

ArrayList<ArrayList> arrayLists\_VP;

String UCCF\_Save\_Location;

String UCCF\_File\_Name;

public Main(){

this.setTitle("FM2VxBPEL");

this.setSize(630,760);

this.setLayout(new BorderLayout(2, 2));

JPanel jPanel\_North=new JPanel(new BorderLayout());

jPanel\_North.setPreferredSize(new Dimension(0,25));

JMenuBar jMenuBar=new JMenuBar();//添加菜单栏

jPanel\_North.add(jMenuBar,BorderLayout.NORTH);

JMenu mFile=new JMenu("File");

JMenuItem mFile\_Close=new JMenuItem("Close");

mFile.add(mFile\_Close);

mFile\_Close.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

dispose();

}

});

jMenuBar.add(mFile);

JMenu mEngineManage=new JMenu("Engine");

JMenuItem startODE=new JMenuItem("Start ODE");

JMenuItem stopODE=new JMenuItem("Stop ODE");

mEngineManage.add(startODE);

mEngineManage.add(stopODE);

jMenuBar.add(mEngineManage);

JMenu mRun=new JMenu("Run");

JMenuItem runProcess=new JMenuItem("Run Process");

runProcess.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

RunProcess r=new RunProcess();

}

});

mRun.add(runProcess);

jMenuBar.add(mRun);

JMenu m\_Tools=new JMenu("Tools");

JMenuItem businessManage=new JMenuItem("Business Manage");

businessManage.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

BusinessManage.openDefaultBrowser();

}

});

m\_Tools.add(businessManage);

jMenuBar.add(m\_Tools);

JMenu mHelp=new JMenu("Help");

JMenuItem metamodel=new JMenuItem("FM MetaModel");

metamodel.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

FeatureModelMatamodel fmm=new FeatureModelMatamodel();

}

});

mHelp.add(metamodel);

jMenuBar.add(mHelp);

final JPanel jPanel\_Center=new JPanel(new BorderLayout());

JTabbedPane jTabbedPane1=new JTabbedPane();

jPanel\_Center.add(jTabbedPane1);

final JPanel jPanel\_Visual\_Customizing=new JPanel(new BorderLayout());

jTabbedPane1.add("Visual Customizing",jPanel\_Visual\_Customizing);

JPanel jPanel\_Visual\_Customizing\_North=new JPanel(null);

jPanel\_Visual\_Customizing\_North.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "Loading File", TitledBorder.CENTER, TitledBorder.TOP, null, null));

jPanel\_Visual\_Customizing\_North.setPreferredSize(new Dimension(0,105)); jPanel\_Visual\_Customizing.add(jPanel\_Visual\_Customizing\_North,BorderLayout.NORTH);

JLabel jLabel\_1=new JLabel("Select FM File:");

jLabel\_1.setBounds(8,20,100,20);

jPanel\_Visual\_Customizing\_North.add(jLabel\_1);

final JTextField jt1=new JTextField();

jt1.setBounds(135,20,290,20);

jPanel\_Visual\_Customizing\_North.add(jt1);

MyButton myButton\_Select\_FM\_File=new MyButton("Browse",430,20,80,20);

jPanel\_Visual\_Customizing\_North.add(myButton\_Select\_FM\_File);

myButton\_Select\_FM\_File.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

JFileChooser fcDlg = new JFileChooser();

fcDlg.setDialogTitle("Please choose file:");

FileNameExtensionFilter filter = new FileNameExtensionFilter("file(\*.scfeaturemodel)", "scfeaturemodel");

fcDlg.setFileFilter(filter);

int returnVal = fcDlg.showOpenDialog(null);

if (returnVal == JFileChooser.APPROVE\_OPTION) {

String filepath = fcDlg.getSelectedFile().getPath();

url\_FM\_FilePath=filepath;

jt1.setText(url\_FM\_FilePath);

jTextArea\_Loger\_Info.append("Start!\r\n");

jTextArea\_Loger\_Info.append("The location of Feature Model file!"+filepath+"\r\n");

File f=new File(url\_FM\_FilePath);

System.out.println(filepath);

}

}

});

MyButton myButton\_Cancel1=new MyButton("Cancel",515,20,80,20);

jPanel\_Visual\_Customizing\_North.add(myButton\_Cancel1);

myButton\_Cancel1.addActionListener(new CancelActionListener(jt1));

JLabel jLabel\_2=new JLabel("Select Save Location:");

jLabel\_2.setBounds(8,50,130,20);

jPanel\_Visual\_Customizing\_North.add(jLabel\_2);

final JTextField jt2=new JTextField();

jt2.setBounds(135,50,290,20);

url\_FM\_Save\_Location=jt2.getText();

jPanel\_Visual\_Customizing\_North.add(jt2);

MyButton myButton\_Select\_SaveLocation=new MyButton("Browse",430,50,80,20);

jPanel\_Visual\_Customizing\_North.add(myButton\_Select\_SaveLocation);

myButton\_Select\_SaveLocation.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

JFileChooser fcDlg = new JFileChooser();

fcDlg.setDialogTitle("Please select the location where the feature model picture is saved");

fcDlg.setFileSelectionMode(JFileChooser.DIRECTORIES\_ONLY);

int returnVal = fcDlg.showOpenDialog(null);

if (returnVal == JFileChooser.APPROVE\_OPTION) {

String filepath = fcDlg.getSelectedFile().getPath();

url\_FM\_Save\_Location=filepath;

jt2.setText(url\_FM\_Save\_Location);

jTextArea\_Loger\_Info.append("The location of diagram!"+filepath+"\r\n");

}

}

});

MyButton myButton\_Cancel2=new MyButton("Cancel",515,50,80,20);

jPanel\_Visual\_Customizing\_North.add(myButton\_Cancel2);

myButton\_Cancel2.addActionListener(new CancelActionListener(jt2));

JLabel jLabel\_3=new JLabel("Input Diagram Name:");

jLabel\_3.setBounds(8,80,130,20);

jPanel\_Visual\_Customizing\_North.add(jLabel\_3);

final JTextField jt3=new JTextField();

jt3.setBounds(135,80,290,20);

jPanel\_Visual\_Customizing\_North.add(jt3);

MyButton myButton\_Create\_Diagram=new MyButton("Create",430,80,80,20);

jPanel\_Visual\_Customizing\_North.add(myButton\_Create\_Diagram);

myButton\_Create\_Diagram.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

try {

Read\_FM\_XMI rfx=new Read\_FM\_XMI(url\_FM\_FilePath);

jTextArea\_FM\_XMI.append(rfx.result);

} catch (IOException e1) {

e1.printStackTrace();

}

url\_FM\_Save\_Location=jt2.getText();

FMDiagram\_Name=jt3.getText();

try {

DrawFMDiagram drawFMDiagram=new DrawFMDiagram(url\_FM\_FilePath,url\_FM\_Save\_Location,FMDiagram\_Name);

} catch (DocumentException e1) {

e1.printStackTrace();

} catch (IOException e1) {

e1.printStackTrace();

}

JPanel jPanel\_FM\_VisualTree=new JPanel(new BorderLayout());

jPanel\_FM\_VisualTree.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "Visual Tree", TitledBorder.CENTER, TitledBorder.TOP, null, null));

jPanel\_FM\_VisualTree.setPreferredSize(new Dimension(0,415)); jPanel\_Visual\_Customizing.add(jPanel\_FM\_VisualTree,BorderLayout.SOUTH);

try {

fmTree=new FMTreeJPanel(url\_FM\_FilePath);

JPanel fmTreeJPanel=fmTree.jPanel;

System.out.println("Selected\_String"+Selected\_String);

fmTreeJPanel.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "Configurable FM Tree", TitledBorder.CENTER, TitledBorder.TOP, null, null));

fmTreeJPanel.setPreferredSize(new Dimension(250,375));

jPanel\_FM\_VisualTree.add(fmTreeJPanel, BorderLayout.WEST);

jPanel\_FM\_VisualTree.setVisible(true);

jPanel\_Center.repaint();

} catch (IOException e1) {

e1.printStackTrace();

} catch (AWTException e1) {

e1.printStackTrace();

} catch (DocumentException e1) {

e1.printStackTrace();

}

JPanel jPanel\_Configuration\_Info=new JPanel(null);

jPanel\_Configuration\_Info.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "Configuration", TitledBorder.CENTER, TitledBorder.TOP, null, null));

jPanel\_Configuration\_Info.setPreferredSize(new Dimension(250,415));

JPanel jPanel\_Configuration\_Info\_Table=new JPanel(null);

jPanel\_Configuration\_Info\_Table.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "Constraints Table", TitledBorder.LEFT, TitledBorder.TOP, null, null));

jPanel\_Configuration\_Info\_Table.setBounds(20,25,210,120);

jPanel\_Configuration\_Info.add(jPanel\_Configuration\_Info\_Table);

try {

ConstraintTable ct=new ConstraintTable(url\_FM\_FilePath);

JScrollPane jScrollPane\_Constraint\_Table=ct.jtableJScrollPane;

jScrollPane\_Constraint\_Table.setBounds(5,20,200,90);

jPanel\_Configuration\_Info\_Table.add(jScrollPane\_Constraint\_Table);

} catch (DocumentException e1) {

e1.printStackTrace();

}

JPanel jPanel\_jList\_Configuration\_Info=new JPanel(null);

jPanel\_jList\_Configuration\_Info.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "FM Configuration Info", TitledBorder.LEFT, TitledBorder.TOP, null, null));

jPanel\_jList\_Configuration\_Info.setBounds(20,155,210,230);

jPanel\_Configuration\_Info.add(jPanel\_jList\_Configuration\_Info);

final DefaultListModel dlm=new DefaultListModel();

final JList jList = new JList(dlm);

jList.setBounds(5,20,200,205);

jPanel\_jList\_Configuration\_Info.add(jList);

final String[] jlist\_selected\_info = new String[1];

jList.addListSelectionListener(new ListSelectionListener(){

@Override

public void valueChanged(ListSelectionEvent e) {

if(!jList.getValueIsAdjusting()){

System.out.println(jList.getSelectedValue());

jlist\_selected\_info[0] = (String) jList.getSelectedValue();

}

}

});

jPanel\_FM\_VisualTree.add(jPanel\_Configuration\_Info,BorderLayout.EAST);

JPanel jPanel\_JButton=new JPanel(null);

jPanel\_FM\_VisualTree.add(jPanel\_JButton,BorderLayout.CENTER);

MyButton myButton\_Add=new MyButton("Add",7,100,80,20);

myButton\_Add.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

Selected\_String=fmTree.selected\_String;

System.out.println("Selected\_String"+Selected\_String);

dlm.addElement(Selected\_String);

arrayList.add(Selected\_String);

}

});

MyButton myButton\_Remove=new MyButton("Remove",7,150,80,20);

myButton\_Remove.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

arrayList.remove(jlist\_selected\_info[0]);

dlm.removeElement(jlist\_selected\_info[0]);

}

});

MyButton myButton\_Validate=new MyButton("Validate",7,200,80,20);

myButton\_Validate.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

try {

Validate v=new Validate(url\_FM\_FilePath,arrayList);

if(v.flag.equals("true")){

MyDialog\_Confirm myDialog\_confirm=new MyDialog\_Confirm("Confirm to FM constraints!");

}else if(v.flag.equals("false")){

MyDialog\_Error myDialog\_error=new MyDialog\_Error("Error!");

}

} catch (DocumentException e1) {

e1.printStackTrace();

}

}

});

jPanel\_JButton.add(myButton\_Add);

jPanel\_JButton.add(myButton\_Remove);

jPanel\_JButton.add(myButton\_Validate);

jPanel\_Visual\_Customizing.add(jPanel\_FM\_VisualTree,BorderLayout.SOUTH);

jPanel\_Center.repaint();

}

});

MyButton myButton\_Cancel3=new MyButton("Cancel",515,80,80,20);

jPanel\_Visual\_Customizing\_North.add(myButton\_Cancel3);

myButton\_Cancel3.addActionListener(new CancelActionListener(jt3));

final JPanel jPanel\_Model\_Transformation=new JPanel(new BorderLayout());

jTabbedPane1.add("Model Transformation",jPanel\_Model\_Transformation);

JPanel jPanel\_Model\_Transformation\_North=new JPanel(null);

jPanel\_Model\_Transformation\_North.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "Loading File", TitledBorder.CENTER, TitledBorder.TOP, null, null));

jPanel\_Model\_Transformation\_North.setPreferredSize(new Dimension(0,140));

jPanel\_Model\_Transformation.add(

jPanel\_Model\_Transformation\_North,BorderLayout.NORTH);

JLabel jLabel\_2\_1=new JLabel("Select FM File:");

jLabel\_2\_1.setBounds(8,20,100,20);

jPanel\_Model\_Transformation\_North.add(jLabel\_2\_1);

final JTextField jt\_2\_1=new JTextField();

jt\_2\_1.setBounds(135,20,290,20);

jPanel\_Model\_Transformation\_North.add(jt\_2\_1);

MyButton myButton\_Select\_FM\_File\_2\_1=new MyButton("Browse",430,20,80,20);

jPanel\_Model\_Transformation\_North.add(myButton\_Select\_FM\_File\_2\_1);

myButton\_Select\_FM\_File\_2\_1.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

JFileChooser fcDlg = new JFileChooser();

fcDlg.setDialogTitle("Please choose file:");

FileNameExtensionFilter filter = new FileNameExtensionFilter("file(\*.scfeaturemodel)", "scfeaturemodel");

fcDlg.setFileFilter(filter);

int returnVal = fcDlg.showOpenDialog(null);

if (returnVal == JFileChooser.APPROVE\_OPTION) {

String filepath = fcDlg.getSelectedFile().getPath();

url\_FM\_FilePath\_2\_1=filepath;

jt\_2\_1.setText(url\_FM\_FilePath\_2\_1);

jTextArea\_Loger\_Info.append("The location of Feature Model file!"+filepath+"\r\n");

File f=new File(url\_FM\_FilePath\_2\_1);

System.out.println(filepath);

}

}

});

MyButton myButton\_Cancel\_2\_1=new MyButton("Cancel",515,20,80,20);

jPanel\_Model\_Transformation\_North.add(myButton\_Cancel\_2\_1);

myButton\_Cancel\_2\_1.addActionListener(new CancelActionListener(jt\_2\_1));

JLabel jLabel\_2\_2=new JLabel("Select Save Location:");

jLabel\_2\_2.setBounds(8,50,130,20);

jPanel\_Model\_Transformation\_North.add(jLabel\_2\_2);

final JTextField jt\_2\_2=new JTextField();

jt\_2\_2.setBounds(135,50,290,20);

jPanel\_Model\_Transformation\_North.add(jt\_2\_2);

MyButton myButton\_Select\_SaveLocation\_2\_2=new MyButton("Browse",430,50,80,20);

jPanel\_Model\_Transformation\_North.add(myButton\_Select\_SaveLocation\_2\_2);

myButton\_Select\_SaveLocation\_2\_2.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

JFileChooser fcDlg = new JFileChooser();

fcDlg.setDialogTitle("Please select the location where the feature model picture is saved");

fcDlg.setFileSelectionMode(JFileChooser.DIRECTORIES\_ONLY);

int returnVal = fcDlg.showOpenDialog(null);

if (returnVal == JFileChooser.APPROVE\_OPTION) {

String filepath = fcDlg.getSelectedFile().getPath();

url\_VxBPEL\_Save\_Location\_2\_2=filepath;

jt\_2\_2.setText(url\_VxBPEL\_Save\_Location\_2\_2);

jTextArea\_Loger\_Info.append("The location of VxBPEL!"+filepath+"\r\n");

}

}

});

MyButton myButton\_Cancel\_2\_2=new MyButton("Cancel",515,50,80,20);

jPanel\_Model\_Transformation\_North.add(myButton\_Cancel\_2\_2);

myButton\_Cancel\_2\_2.addActionListener(new CancelActionListener(jt\_2\_2));

JLabel jLabel\_2\_3=new JLabel("Input VxBPEL Name:");

jLabel\_2\_3.setBounds(8,80,130,20);

jPanel\_Model\_Transformation\_North.add(jLabel\_2\_3);

final JTextField jt\_2\_3=new JTextField();

jt\_2\_3.setBounds(135,80,290,20);

jPanel\_Model\_Transformation\_North.add(jt\_2\_3);

MyButton myButton\_Model\_Convert=new MyButton("Convert",430,80,80,20);

jPanel\_Model\_Transformation\_North.add(myButton\_Model\_Convert);

//myButton\_Model\_Convert执行事件

myButton\_Model\_Convert.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

vxbpel\_name=jt\_2\_3.getText();

try {

ModelTran mt=new ModelTran(url\_FM\_FilePath\_2\_1,url\_VxBPEL\_Save\_Location\_2\_2,vxbpel\_name);

} catch (DocumentException e1) {

e1.printStackTrace();

} catch (IOException e1) {

e1.printStackTrace();

}

}

});

MyButton myButton\_Cancel\_2\_3=new MyButton("Cancel",515,80,80,20);

jPanel\_Model\_Transformation\_North.add(myButton\_Cancel\_2\_3);

myButton\_Cancel\_2\_3.addActionListener(new CancelActionListener(jt\_2\_3));

JLabel jLabel\_2\_4=new JLabel("Parse Variability Info:");

jLabel\_2\_4.setBounds(8,110,130,20);

jPanel\_Model\_Transformation\_North.add(jLabel\_2\_4);

final JTextField jt\_2\_4=new JTextField();

jt\_2\_4.setBounds(135,110,290,20);

jPanel\_Model\_Transformation\_North.add(jt\_2\_4);

MyButton myButton\_Browse\_2\_4=new MyButton("Browse",430,110,80,20);

jPanel\_Model\_Transformation\_North.add(myButton\_Browse\_2\_4);

myButton\_Browse\_2\_4.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

JFileChooser fcDlg = new JFileChooser();

fcDlg.setDialogTitle("Please choose file:");

FileNameExtensionFilter filter =

new FileNameExtensionFilter("file(\*.bpel)", "bpel");

fcDlg.setFileFilter(filter);

int returnVal = fcDlg.showOpenDialog(null);

if (returnVal == JFileChooser.APPROVE\_OPTION) {

String filepath = fcDlg.getSelectedFile().getPath();

vxbpel\_file\_location=filepath;

jt\_2\_4.setText(vxbpel\_file\_location);

jTextArea\_Loger\_Info.append("Start!\r\n");

jTextArea\_Loger\_Info.append("The location of vxbpel file!"+filepath+"\r\n");

System.out.println(filepath);

}

}

});

MyButton myButton\_Parse\_2\_4=new MyButton("Parse",515,110,80,20);

jPanel\_Model\_Transformation\_North.add(myButton\_Parse\_2\_4);

myButton\_Parse\_2\_4.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

JPanel jPanel\_VxBPEL=new JPanel(new BorderLayout());

jPanel\_VxBPEL.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "VxBPEL", TitledBorder.CENTER, TitledBorder.TOP, null, null));

jPanel\_VxBPEL.setPreferredSize(new Dimension(0,380));

jPanel\_Model\_Transformation.add(jPanel\_VxBPEL,BorderLayout.SOUTH);

try {

ProcessTree pt=new ProcessTree(vxbpel\_file\_location);

JPanel jPanel\_vxbpel\_Tree=pt.jPanel;

jPanel\_vxbpel\_Tree.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "Process Tree", TitledBorder.CENTER, TitledBorder.TOP, null, null));

jPanel\_vxbpel\_Tree.setPreferredSize(new Dimension(250,325));

jPanel\_VxBPEL.add(jPanel\_vxbpel\_Tree,BorderLayout.WEST);

} catch (DocumentException e1) {

e1.printStackTrace();

}

JPanel jPanel\_vxbpel\_configuration\_info=new JPanel(null);

jPanel\_vxbpel\_configuration\_info.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "VxBPEL Configuration Info", TitledBorder.CENTER, TitledBorder.TOP, null, null));

jPanel\_vxbpel\_configuration\_info.setPreferredSize(new Dimension(250,325));

jPanel\_VxBPEL.add(jPanel\_vxbpel\_configuration\_info,BorderLayout.EAST);

JPanel jPanel\_fm\_conf\_info=new JPanel(null);

jPanel\_fm\_conf\_info.setBounds(10,20,230,140);

jPanel\_fm\_conf\_info.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "FM Configuration Info", TitledBorder.LEFT, TitledBorder.TOP, null, null));

JTextArea jTextArea\_fm\_conf\_info=new JTextArea();

JScrollPane jScrollPane\_fm\_conf\_info=new JScrollPane(jTextArea\_fm\_conf\_info);

jScrollPane\_fm\_conf\_info.setBounds(5,20,220,110);

jPanel\_fm\_conf\_info.add(jScrollPane\_fm\_conf\_info);

jPanel\_vxbpel\_configuration\_info.add(jPanel\_fm\_conf\_info);

if(arrayList.size()>0){

for(int i=0;i<arrayList.size();i++){

jTextArea\_fm\_conf\_info.append(arrayList.get(i)+"\r\n");

}

}

JPanel jPanel\_Match\_vp\_variant=new JPanel(null);

jPanel\_Match\_vp\_variant.setBounds(10,170,230,180);

jPanel\_Match\_vp\_variant.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "VariationPoint && Variant Conf", TitledBorder.LEFT, TitledBorder.TOP, null, null));

final JTextArea jTextArea\_user\_conf=new JTextArea();

JScrollPane jScrollPane\_user\_conf=new JScrollPane(jTextArea\_user\_conf);

jScrollPane\_user\_conf.setBounds(5,20,220,155);

jPanel\_Match\_vp\_variant.add(jScrollPane\_user\_conf);

jPanel\_vxbpel\_configuration\_info.add(jPanel\_Match\_vp\_variant);

JPanel jPanel\_JButton2=new JPanel(null);

jPanel\_VxBPEL.add(jPanel\_JButton2,BorderLayout.CENTER);

MyButton myButton\_Match\_VP=new MyButton("Match",7,120,80,20);

myButton\_Match\_VP.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

arrayLists\_VP=new ArrayList<ArrayList>();

for(int i=0;i<arrayList.size();i++){

try {

FindVP findVP=new FindVP(vxbpel\_file\_location,arrayList.get(i));

jTextArea\_user\_conf.append("VariationPoint->"+findVP.flag1+"\r\n");

jTextArea\_user\_conf.append(" Variant->"+findVP.flag2+"\r\n");

ArrayList<String> a=new ArrayList<String>();

a.add(findVP.flag1);

a.add(findVP.flag2);

arrayLists\_VP.add(a);

} catch (DocumentException e1) {

e1.printStackTrace();

}

}

}

});

MyButton myButton\_UCCF=new MyButton("UCCF",7,170,80,20);

jPanel\_JButton2.add(myButton\_Match\_VP);

jPanel\_Center.repaint();

}

});

final JPanel jPanel\_UCCF=new JPanel(new BorderLayout());

jTabbedPane1.add("Create UCCF",jPanel\_UCCF);

JPanel jPanel\_UCCF\_North=new JPanel(null);

jPanel\_UCCF\_North.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "Create UCCF", TitledBorder.CENTER, TitledBorder.TOP, null, null));

jPanel\_UCCF\_North.setPreferredSize(new Dimension(0,75));

jPanel\_UCCF.add(jPanel\_UCCF\_North,BorderLayout.NORTH);

JLabel jLabel\_3\_1=new JLabel("Select Save Location:");

jLabel\_3\_1.setBounds(8,20,140,20);

jPanel\_UCCF\_North.add(jLabel\_3\_1);

final JTextField jt\_3\_1=new JTextField();

jt\_3\_1.setBounds(135,20,290,20);

jPanel\_UCCF\_North.add(jt\_3\_1);

MyButton myButton\_Select\_3\_1=new MyButton("Browse",430,20,80,20);

jPanel\_UCCF\_North.add(myButton\_Select\_3\_1);

myButton\_Select\_3\_1.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

JFileChooser fcDlg = new JFileChooser();

fcDlg.setDialogTitle("Please select the location where the UCCF is saved");

fcDlg.setFileSelectionMode(JFileChooser.DIRECTORIES\_ONLY);

int returnVal = fcDlg.showOpenDialog(null);

if (returnVal == JFileChooser.APPROVE\_OPTION) {

String filepath = fcDlg.getSelectedFile().getPath();

UCCF\_Save\_Location=filepath;

jt\_3\_1.setText(UCCF\_Save\_Location);

jTextArea\_Loger\_Info.append("The save location of UCCF!"+filepath+"\r\n");

}

}

});

MyButton myButton\_Cancel\_3\_1=new MyButton("Cancel",515,20,80,20);

jPanel\_UCCF\_North.add(myButton\_Cancel\_3\_1);

myButton\_Cancel\_3\_1.addActionListener(new CancelActionListener(jt\_3\_1));

JLabel jLabel\_3\_2=new JLabel("Input UCCF File Name:");

jLabel\_3\_2.setBounds(8,50,140,20);

jPanel\_UCCF\_North.add(jLabel\_3\_2);

final JTextField jt\_3\_2=new JTextField();

jt\_3\_2.setBounds(135,50,290,20);

jPanel\_UCCF\_North.add(jt\_3\_2);

MyButton myButton\_create\_UCCF=new MyButton("UCCF",430,50,80,20);

jPanel\_UCCF\_North.add(myButton\_create\_UCCF);

myButton\_create\_UCCF.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

JPanel jPanel\_UCCF\_South=new JPanel(null);

jPanel\_UCCF\_South.setPreferredSize(new Dimension(0,440));

jPanel\_UCCF\_South.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "UCCF Info", TitledBorder.CENTER, TitledBorder.TOP, null, null));

JTextArea jTextArea\_UCCF\_Info=new JTextArea();

jTextArea\_UCCF\_Info.setFont(jTextArea\_Font);

JScrollPane jScrollPane\_UCCF\_Info=new JScrollPane(jTextArea\_UCCF\_Info);

jScrollPane\_UCCF\_Info.setBounds(5,20,595,410);

jPanel\_UCCF\_South.add(jScrollPane\_UCCF\_Info);

jPanel\_UCCF.add(jPanel\_UCCF\_South,BorderLayout.SOUTH);

UCCF\_File\_Name=jt\_3\_2.getText();

String all\_uccf\_location=UCCF\_Save\_Location+"\\"+UCCF\_File\_Name+".uccf";

UccfInof u=new UccfInof(arrayLists\_VP);

try {

CreateUCCF c=new CreateUCCF(all\_uccf\_location,u.s\_info);

} catch (IOException e1) {

e1.printStackTrace();

}

File f=new File(all\_uccf\_location);

try {

Read\_UCCF ru=new Read\_UCCF(f,arrayLists\_VP.size()+2);

jTextArea\_UCCF\_Info.append(ru.result);

} catch (IOException e1) {

e1.printStackTrace();

}

jPanel\_Center.repaint();

}

});

MyButton myButton\_Cancel\_3\_2=new MyButton("Cancel",515,50,80,20);

jPanel\_UCCF\_North.add(myButton\_Cancel\_3\_2);

myButton\_Cancel\_3\_2.addActionListener(new CancelActionListener(jt\_3\_2));

JPanel jPanel\_South=new JPanel(new BorderLayout());

jPanel\_South.setPreferredSize(new Dimension(0,140));

JTabbedPane jTabbedPane2=new JTabbedPane();

jPanel\_South.add(jTabbedPane2,BorderLayout.NORTH);

JPanel jPanel\_Loger\_Info=new JPanel(new BorderLayout());

jTabbedPane2.add("Logger Info",jPanel\_Loger\_Info);

jTextArea\_Loger\_Info=new JTextArea();

jTextArea\_Loger\_Info.setFont(jTextArea\_Font);

JScrollPane jScrollPane1=new JScrollPane(jTextArea\_Loger\_Info);

jScrollPane1.setPreferredSize(new Dimension(0,110));

jPanel\_Loger\_Info.add(jScrollPane1,BorderLayout.NORTH);

JPanel jPanel\_FM\_XMI=new JPanel(new BorderLayout());

jTabbedPane2.add("FM XMI",jPanel\_FM\_XMI);

jTextArea\_FM\_XMI=new JTextArea();

jTextArea\_FM\_XMI.setFont(jTextArea\_Font);

JScrollPane jScrollPane2=new JScrollPane(jTextArea\_FM\_XMI);

jScrollPane2.setPreferredSize(new Dimension(0,110));

jPanel\_FM\_XMI.add(jScrollPane2,BorderLayout.NORTH);

this.add(jPanel\_North,BorderLayout.NORTH);

this.add(jPanel\_Center,BorderLayout.CENTER);

this.add(jPanel\_South,BorderLayout.SOUTH);

this.setVisible(true);

this.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

this.setLocationRelativeTo(null);

}

public static void main(String[] args) {

Main m=new Main();

}

}

public class Main\_ATL extends JFrame{

FMTreeJPanel fmTree;

String Selected\_String;

Font jTextArea\_Font=new Font("Consolas",Font.PLAIN,12);

String url\_FM\_FilePath;

String url\_FM\_Save\_Location;

ArrayList<String> arrayList=new ArrayList<String>();

String FMDiagram\_Name;

JTextArea jTextArea\_Loger\_Info;

JTextArea jTextArea\_FM\_XMI;

String url\_FM\_FilePath\_2\_1;

String url\_VxBPEL\_Save\_Location\_2\_2;

String vxbpel\_name;

String vxbpel\_file\_location;

ArrayList<ArrayList> arrayLists\_VP;

String UCCF\_Save\_Location;

String UCCF\_File\_Name;

public Main\_ATL(){

this.setTitle("FM2VxBPEL");

this.setSize(630,760);

this.setLayout(new BorderLayout(2, 2));

JPanel jPanel\_North=new JPanel(new BorderLayout());

jPanel\_North.setPreferredSize(new Dimension(0,25));

JMenuBar jMenuBar=new JMenuBar();//添加菜单栏

jPanel\_North.add(jMenuBar,BorderLayout.NORTH);

JMenu mFile=new JMenu("File");

JMenuItem mFile\_Close=new JMenuItem("Close");

mFile.add(mFile\_Close);

mFile\_Close.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

dispose();

}

});

jMenuBar.add(mFile);

//Engine Manage

JMenu mEngineManage=new JMenu("Engine");

JMenuItem startODE=new JMenuItem("Start ODE");

JMenuItem stopODE=new JMenuItem("Stop ODE");

mEngineManage.add(startODE);

mEngineManage.add(stopODE);

jMenuBar.add(mEngineManage);

JMenu mRun=new JMenu("Run");

JMenuItem runProcess=new JMenuItem("Run Process");

runProcess.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

RunProcess r=new RunProcess();

}

});

mRun.add(runProcess);

jMenuBar.add(mRun);

JMenu m\_Tools=new JMenu("Tools");

JMenuItem businessManage=new JMenuItem("Business Manage");

businessManage.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

BusinessManage.openDefaultBrowser();

}

});

m\_Tools.add(businessManage);

jMenuBar.add(m\_Tools);

JMenu mHelp=new JMenu("Help");

JMenuItem metamodel=new JMenuItem("FM MetaModel");

metamodel.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

FeatureModelMatamodel fmm=new FeatureModelMatamodel();

}

});

mHelp.add(metamodel);

jMenuBar.add(mHelp);

final JPanel jPanel\_Center=new JPanel(new BorderLayout());

JTabbedPane jTabbedPane1=new JTabbedPane();

jPanel\_Center.add(jTabbedPane1);

final JPanel jPanel\_Visual\_Customizing=new JPanel(new BorderLayout());

jTabbedPane1.add("Visual Customizing",jPanel\_Visual\_Customizing);

JPanel jPanel\_Visual\_Customizing\_North=new JPanel(null);

jPanel\_Visual\_Customizing\_North.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "Loading File", TitledBorder.CENTER, TitledBorder.TOP, null, null));

jPanel\_Visual\_Customizing\_North.setPreferredSize(new Dimension(0,105));

jPanel\_Visual\_Customizing.add(jPanel\_Visual\_Customizing\_North,BorderLayout.NORTH);

JLabel jLabel\_1=new JLabel("Select FM File:");

jLabel\_1.setBounds(8,20,100,20);

jPanel\_Visual\_Customizing\_North.add(jLabel\_1);

final JTextField jt1=new JTextField();

jt1.setBounds(135,20,290,20);

jPanel\_Visual\_Customizing\_North.add(jt1);

MyButton myButton\_Select\_FM\_File=new MyButton("Browse",430,20,80,20);

jPanel\_Visual\_Customizing\_North.add(myButton\_Select\_FM\_File);

myButton\_Select\_FM\_File.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

JFileChooser fcDlg = new JFileChooser();

fcDlg.setDialogTitle("Please choose file:");

FileNameExtensionFilter filter = new FileNameExtensionFilter("file(\*.scfeaturemodel)", "scfeaturemodel");

fcDlg.setFileFilter(filter);

int returnVal = fcDlg.showOpenDialog(null);

if (returnVal == JFileChooser.APPROVE\_OPTION) {

String filepath = fcDlg.getSelectedFile().getPath();

url\_FM\_FilePath=filepath;

jt1.setText(url\_FM\_FilePath);

jTextArea\_Loger\_Info.append("Start!\r\n");

jTextArea\_Loger\_Info.append("The location of Feature Model file!"+filepath+"\r\n");

File f=new File(url\_FM\_FilePath);

System.out.println(filepath);

}

}

});

MyButton myButton\_Cancel1=new MyButton("Cancel",515,20,80,20);

jPanel\_Visual\_Customizing\_North.add(myButton\_Cancel1);

myButton\_Cancel1.addActionListener(new CancelActionListener(jt1));

JLabel jLabel\_2=new JLabel("Select Save Location:");

jLabel\_2.setBounds(8,50,130,20);

jPanel\_Visual\_Customizing\_North.add(jLabel\_2);

final JTextField jt2=new JTextField();

jt2.setBounds(135,50,290,20);

url\_FM\_Save\_Location=jt2.getText();

jPanel\_Visual\_Customizing\_North.add(jt2);

MyButton myButton\_Select\_SaveLocation=new MyButton("Browse",430,50,80,20);

jPanel\_Visual\_Customizing\_North.add(myButton\_Select\_SaveLocation);

myButton\_Select\_SaveLocation.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

JFileChooser fcDlg = new JFileChooser();

fcDlg.setDialogTitle("Please select the location where the feature model picture is saved");

fcDlg.setFileSelectionMode(JFileChooser.DIRECTORIES\_ONLY);

int returnVal = fcDlg.showOpenDialog(null);

if (returnVal == JFileChooser.APPROVE\_OPTION) {

String filepath = fcDlg.getSelectedFile().getPath();

url\_FM\_Save\_Location=filepath;

jt2.setText(url\_FM\_Save\_Location);

jTextArea\_Loger\_Info.append("The location of diagram!"+filepath+"\r\n");

}

}

});

MyButton myButton\_Cancel2=new MyButton("Cancel",515,50,80,20);

jPanel\_Visual\_Customizing\_North.add(myButton\_Cancel2);

myButton\_Cancel2.addActionListener(new CancelActionListener(jt2));

JLabel jLabel\_3=new JLabel("Input Diagram Name:");

jLabel\_3.setBounds(8,80,130,20);

jPanel\_Visual\_Customizing\_North.add(jLabel\_3);

final JTextField jt3=new JTextField();

jt3.setBounds(135,80,290,20);

jPanel\_Visual\_Customizing\_North.add(jt3);

MyButton myButton\_Create\_Diagram=new MyButton("Create",430,80,80,20);

jPanel\_Visual\_Customizing\_North.add(myButton\_Create\_Diagram);

myButton\_Create\_Diagram.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

try {

Read\_FM\_XMI rfx=new Read\_FM\_XMI(url\_FM\_FilePath);

jTextArea\_FM\_XMI.append(rfx.result);

} catch (IOException e1) {

e1.printStackTrace();

}

url\_FM\_Save\_Location=jt2.getText();

FMDiagram\_Name=jt3.getText();

try {

DrawFMDiagram drawFMDiagram=new DrawFMDiagram(url\_FM\_FilePath,url\_FM\_Save\_Location,FMDiagram\_Name);

} catch (DocumentException e1) {

e1.printStackTrace();

} catch (IOException e1) {

e1.printStackTrace();

}

JPanel jPanel\_FM\_VisualTree=new JPanel(new BorderLayout());

jPanel\_FM\_VisualTree.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "Visual Tree", TitledBorder.CENTER, TitledBorder.TOP, null, null));

jPanel\_FM\_VisualTree.setPreferredSize(new Dimension(0,415));

jPanel\_Visual\_Customizing.add(jPanel\_FM\_VisualTree,BorderLayout.SOUTH);

try {

fmTree=new FMTreeJPanel(url\_FM\_FilePath);

JPanel fmTreeJPanel=fmTree.jPanel;

System.out.println("Selected\_String"+Selected\_String);

fmTreeJPanel.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "Configurable FM Tree", TitledBorder.CENTER, TitledBorder.TOP, null, null));

fmTreeJPanel.setPreferredSize(new Dimension(250,375));

jPanel\_FM\_VisualTree.add(fmTreeJPanel, BorderLayout.WEST);

jPanel\_FM\_VisualTree.setVisible(true);

jPanel\_Center.repaint();

} catch (IOException e1) {

e1.printStackTrace();

} catch (AWTException e1) {

e1.printStackTrace();

} catch (DocumentException e1) {

e1.printStackTrace();

}

JPanel jPanel\_Configuration\_Info=new JPanel(null);

jPanel\_Configuration\_Info.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "Configuration", TitledBorder.CENTER, TitledBorder.TOP, null, null));

jPanel\_Configuration\_Info.setPreferredSize(new Dimension(250,415));

JPanel jPanel\_Configuration\_Info\_Table=new JPanel(null);

jPanel\_Configuration\_Info\_Table.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "Constraints Table", TitledBorder.LEFT, TitledBorder.TOP, null, null));

jPanel\_Configuration\_Info\_Table.setBounds(20,25,210,120);

jPanel\_Configuration\_Info.add(jPanel\_Configuration\_Info\_Table);

try {

ConstraintTable ct=new ConstraintTable(url\_FM\_FilePath);

JScrollPane jScrollPane\_Constraint\_Table=ct.jtableJScrollPane;

jScrollPane\_Constraint\_Table.setBounds(5,20,200,90);

jPanel\_Configuration\_Info\_Table.add(jScrollPane\_Constraint\_Table);

} catch (DocumentException e1) {

e1.printStackTrace();

}

JPanel jPanel\_jList\_Configuration\_Info=new JPanel(null);

jPanel\_jList\_Configuration\_Info.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "FM Configuration Info", TitledBorder.LEFT, TitledBorder.TOP, null, null));

jPanel\_jList\_Configuration\_Info.setBounds(20,155,210,230);

jPanel\_Configuration\_Info.add(jPanel\_jList\_Configuration\_Info);

final DefaultListModel dlm=new DefaultListModel();

final JList jList = new JList(dlm);

jList.setBounds(5,20,200,205);

jPanel\_jList\_Configuration\_Info.add(jList);

final String[] jlist\_selected\_info = new String[1];

jList.addListSelectionListener(new ListSelectionListener(){

@Override

public void valueChanged(ListSelectionEvent e) {

if(!jList.getValueIsAdjusting()){ //设置只有释放鼠标时才触发

System.out.println(jList.getSelectedValue());

jlist\_selected\_info[0] = (String) jList.getSelectedValue();

}

}

});

jPanel\_FM\_VisualTree.add(jPanel\_Configuration\_Info,BorderLayout.EAST);

JPanel jPanel\_JButton=new JPanel(null);

jPanel\_FM\_VisualTree.add(jPanel\_JButton,BorderLayout.CENTER);

MyButton myButton\_Add=new MyButton("Add",7,100,80,20);

myButton\_Add.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

Selected\_String=fmTree.selected\_String;

System.out.println("Selected\_String"+Selected\_String);

dlm.addElement(Selected\_String);

arrayList.add(Selected\_String);

}

});

MyButton myButton\_Remove=new MyButton("Remove",7,150,80,20);

myButton\_Remove.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

arrayList.remove(jlist\_selected\_info[0]);

dlm.removeElement(jlist\_selected\_info[0]);

}

});

MyButton myButton\_Validate=new MyButton("Validate",7,200,80,20);

myButton\_Validate.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

try {

Validate v=new Validate(url\_FM\_FilePath,arrayList);

if(v.flag.equals("true")){

MyDialog\_Confirm myDialog\_confirm=new MyDialog\_Confirm("Confirm to FM constraints!");

}else if(v.flag.equals("false")){

MyDialog\_Error myDialog\_error=new MyDialog\_Error("Error!");

}

} catch (DocumentException e1) {

e1.printStackTrace();

}

}

});

jPanel\_JButton.add(myButton\_Add);

jPanel\_JButton.add(myButton\_Remove);

jPanel\_JButton.add(myButton\_Validate);

jPanel\_Visual\_Customizing.add(jPanel\_FM\_VisualTree,BorderLayout.SOUTH);

jPanel\_Center.repaint();

}

});

MyButton myButton\_Cancel3=new MyButton("Cancel",515,80,80,20);

jPanel\_Visual\_Customizing\_North.add(myButton\_Cancel3);

myButton\_Cancel3.addActionListener(new CancelActionListener(jt3));

final JPanel jPanel\_Model\_Transformation=new JPanel(new BorderLayout());

jTabbedPane1.add("Standard VxBPEL",jPanel\_Model\_Transformation);

JPanel jPanel\_Model\_Transformation\_North=new JPanel(null);

jPanel\_Model\_Transformation\_North.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "Loading File", TitledBorder.CENTER, TitledBorder.TOP, null, null));

jPanel\_Model\_Transformation\_North.setPreferredSize(new Dimension(0,140));

jPanel\_Model\_Transformation.add(

jPanel\_Model\_Transformation\_North,BorderLayout.NORTH);

JLabel jLabel\_2\_1=new JLabel("Select File:");

jLabel\_2\_1.setBounds(8,20,100,20);

jPanel\_Model\_Transformation\_North.add(jLabel\_2\_1);

final JTextField jt\_2\_1=new JTextField();

jt\_2\_1.setBounds(135,20,290,20);

jPanel\_Model\_Transformation\_North.add(jt\_2\_1);

MyButton myButton\_Select\_FM\_File\_2\_1=new MyButton("Browse",430,20,80,20);

jPanel\_Model\_Transformation\_North.add(myButton\_Select\_FM\_File\_2\_1);

myButton\_Select\_FM\_File\_2\_1.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

JFileChooser fcDlg = new JFileChooser();

fcDlg.setDialogTitle("Please choose file:");

FileNameExtensionFilter filter = new FileNameExtensionFilter("file(\*.xmi,\*vxbpel)", "xmi","vxbpel");

fcDlg.setFileFilter(filter);

int returnVal = fcDlg.showOpenDialog(null);

if (returnVal == JFileChooser.APPROVE\_OPTION) {

String filepath = fcDlg.getSelectedFile().getPath();

url\_FM\_FilePath\_2\_1=filepath;

jt\_2\_1.setText(url\_FM\_FilePath\_2\_1);

jTextArea\_Loger\_Info.append("The location of Feature Model file!"+filepath+"\r\n");

File f=new File(url\_FM\_FilePath\_2\_1);

System.out.println(filepath);

}

}

});

MyButton myButton\_Cancel\_2\_1=new MyButton("Cancel",515,20,80,20);

jPanel\_Model\_Transformation\_North.add(myButton\_Cancel\_2\_1);

myButton\_Cancel\_2\_1.addActionListener(new CancelActionListener(jt\_2\_1));

JLabel jLabel\_2\_2=new JLabel("Select Save Location:");

jLabel\_2\_2.setBounds(8,50,130,20);

jPanel\_Model\_Transformation\_North.add(jLabel\_2\_2);

final JTextField jt\_2\_2=new JTextField();

jt\_2\_2.setBounds(135,50,290,20);

jPanel\_Model\_Transformation\_North.add(jt\_2\_2);

MyButton myButton\_Select\_SaveLocation\_2\_2=new MyButton("Browse",430,50,80,20);

jPanel\_Model\_Transformation\_North.add(myButton\_Select\_SaveLocation\_2\_2);

myButton\_Select\_SaveLocation\_2\_2.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

JFileChooser fcDlg = new JFileChooser();

fcDlg.setDialogTitle("Please select the location where the feature model picture is saved");

fcDlg.setFileSelectionMode(JFileChooser.DIRECTORIES\_ONLY);

int returnVal = fcDlg.showOpenDialog(null);

if (returnVal == JFileChooser.APPROVE\_OPTION) {

String filepath = fcDlg.getSelectedFile().getPath();

url\_VxBPEL\_Save\_Location\_2\_2=filepath;

jt\_2\_2.setText(url\_VxBPEL\_Save\_Location\_2\_2);

jTextArea\_Loger\_Info.append("The location of VxBPEL!"+filepath+"\r\n");

}

}

});

MyButton myButton\_Cancel\_2\_2=new MyButton("Cancel",515,50,80,20);

jPanel\_Model\_Transformation\_North.add(myButton\_Cancel\_2\_2);

myButton\_Cancel\_2\_2.addActionListener(new CancelActionListener(jt\_2\_2));

JLabel jLabel\_2\_3=new JLabel("Input VxBPEL Name:");

jLabel\_2\_3.setBounds(8,80,130,20);

jPanel\_Model\_Transformation\_North.add(jLabel\_2\_3);

final JTextField jt\_2\_3=new JTextField();

jt\_2\_3.setBounds(135,80,290,20);

jPanel\_Model\_Transformation\_North.add(jt\_2\_3);

MyButton myButton\_Model\_Convert=new MyButton("Convert",430,80,80,20);

jPanel\_Model\_Transformation\_North.add(myButton\_Model\_Convert);

myButton\_Model\_Convert.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

vxbpel\_name=jt\_2\_3.getText();

try {

ModelTran mt=new ModelTran(url\_FM\_FilePath\_2\_1,url\_VxBPEL\_Save\_Location\_2\_2,vxbpel\_name);

} catch (DocumentException e1) {

e1.printStackTrace();

} catch (IOException e1) {

e1.printStackTrace();

}

}

});

MyButton myButton\_Cancel\_2\_3=new MyButton("Cancel",515,80,80,20);

jPanel\_Model\_Transformation\_North.add(myButton\_Cancel\_2\_3);

myButton\_Cancel\_2\_3.addActionListener(new CancelActionListener(jt\_2\_3));

JLabel jLabel\_2\_4=new JLabel("Parse Variability Info:");

jLabel\_2\_4.setBounds(8,110,130,20);

jPanel\_Model\_Transformation\_North.add(jLabel\_2\_4);

final JTextField jt\_2\_4=new JTextField();

jt\_2\_4.setBounds(135,110,290,20);

jPanel\_Model\_Transformation\_North.add(jt\_2\_4);

MyButton myButton\_Browse\_2\_4=new MyButton("Browse",430,110,80,20);

jPanel\_Model\_Transformation\_North.add(myButton\_Browse\_2\_4);

myButton\_Browse\_2\_4.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

JFileChooser fcDlg = new JFileChooser();

fcDlg.setDialogTitle("Please choose file:");

FileNameExtensionFilter filter =

new FileNameExtensionFilter("file(\*.bpel)", "bpel");

fcDlg.setFileFilter(filter);

int returnVal = fcDlg.showOpenDialog(null);

if (returnVal == JFileChooser.APPROVE\_OPTION) {

String filepath = fcDlg.getSelectedFile().getPath();

vxbpel\_file\_location=filepath;

jt\_2\_4.setText(vxbpel\_file\_location);

jTextArea\_Loger\_Info.append("Start!\r\n");

jTextArea\_Loger\_Info.append("The location of vxbpel file!"+filepath+"\r\n");

System.out.println(filepath);

}

}

});

MyButton myButton\_Parse\_2\_4=new MyButton("Parse",515,110,80,20);

jPanel\_Model\_Transformation\_North.add(myButton\_Parse\_2\_4);

myButton\_Parse\_2\_4.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

JPanel jPanel\_VxBPEL=new JPanel(new BorderLayout());

jPanel\_VxBPEL.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "VxBPEL", TitledBorder.CENTER, TitledBorder.TOP, null, null));

jPanel\_VxBPEL.setPreferredSize(new Dimension(0,380));

jPanel\_Model\_Transformation.add(jPanel\_VxBPEL,BorderLayout.SOUTH);

try {

ProcessTree pt=new ProcessTree(vxbpel\_file\_location);

JPanel jPanel\_vxbpel\_Tree=pt.jPanel;

jPanel\_vxbpel\_Tree.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "Process Tree", TitledBorder.CENTER, TitledBorder.TOP, null, null));

jPanel\_vxbpel\_Tree.setPreferredSize(new Dimension(250,325));

jPanel\_VxBPEL.add(jPanel\_vxbpel\_Tree,BorderLayout.WEST);

} catch (DocumentException e1) {

e1.printStackTrace();

}

JPanel jPanel\_vxbpel\_configuration\_info=new JPanel(null);

jPanel\_vxbpel\_configuration\_info.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "VxBPEL Configuration Info", TitledBorder.CENTER, TitledBorder.TOP, null, null));

jPanel\_vxbpel\_configuration\_info.setPreferredSize(new Dimension(250,325));

jPanel\_VxBPEL.add(jPanel\_vxbpel\_configuration\_info,BorderLayout.EAST);

JPanel jPanel\_fm\_conf\_info=new JPanel(null);

jPanel\_fm\_conf\_info.setBounds(10,20,230,140);

jPanel\_fm\_conf\_info.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "FM Configuration Info", TitledBorder.LEFT, TitledBorder.TOP, null, null));

JTextArea jTextArea\_fm\_conf\_info=new JTextArea();

JScrollPane jScrollPane\_fm\_conf\_info=new JScrollPane(jTextArea\_fm\_conf\_info);

jScrollPane\_fm\_conf\_info.setBounds(5,20,220,110);

jPanel\_fm\_conf\_info.add(jScrollPane\_fm\_conf\_info);

jPanel\_vxbpel\_configuration\_info.add(jPanel\_fm\_conf\_info);

if(arrayList.size()>0){

for(int i=0;i<arrayList.size();i++){

jTextArea\_fm\_conf\_info.append(arrayList.get(i)+"\r\n");

}

}

JPanel jPanel\_Match\_vp\_variant=new JPanel(null);

jPanel\_Match\_vp\_variant.setBounds(10,170,230,180);

jPanel\_Match\_vp\_variant.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "VariationPoint && Variant Conf", TitledBorder.LEFT, TitledBorder.TOP, null, null));

final JTextArea jTextArea\_user\_conf=new JTextArea();

JScrollPane jScrollPane\_user\_conf=new JScrollPane(jTextArea\_user\_conf);

jScrollPane\_user\_conf.setBounds(5,20,220,155);

jPanel\_Match\_vp\_variant.add(jScrollPane\_user\_conf);

jPanel\_vxbpel\_configuration\_info.add(jPanel\_Match\_vp\_variant);

JPanel jPanel\_JButton2=new JPanel(null);

jPanel\_VxBPEL.add(jPanel\_JButton2,BorderLayout.CENTER);

MyButton myButton\_Match\_VP=new MyButton("Match",7,120,80,20);

myButton\_Match\_VP.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

arrayLists\_VP=new ArrayList<ArrayList>();

for(int i=0;i<arrayList.size();i++){

try {

FindVP findVP=new FindVP(vxbpel\_file\_location,arrayList.get(i));

jTextArea\_user\_conf.append("VariationPoint->"+findVP.flag1+"\r\n");

jTextArea\_user\_conf.append(" Variant->"+findVP.flag2+"\r\n");

ArrayList<String> a=new ArrayList<String>();

a.add(findVP.flag1);

a.add(findVP.flag2);

arrayLists\_VP.add(a);

} catch (DocumentException e1) {

e1.printStackTrace();

}

}

}

});

jPanel\_JButton2.add(myButton\_Match\_VP);

jPanel\_Center.repaint();//很重要

}

});

final JPanel jPanel\_UCCF=new JPanel(new BorderLayout());//可视化定制

jTabbedPane1.add("Create UCCF",jPanel\_UCCF);

JPanel jPanel\_UCCF\_North=new JPanel(null);

jPanel\_UCCF\_North.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "Create UCCF", TitledBorder.CENTER, TitledBorder.TOP, null, null));

jPanel\_UCCF\_North.setPreferredSize(new Dimension(0,75));

jPanel\_UCCF.add(jPanel\_UCCF\_North,BorderLayout.NORTH);

JLabel jLabel\_3\_1=new JLabel("Select Save Location:");

jLabel\_3\_1.setBounds(8,20,140,20);

jPanel\_UCCF\_North.add(jLabel\_3\_1);

final JTextField jt\_3\_1=new JTextField();

jt\_3\_1.setBounds(135,20,290,20);

jPanel\_UCCF\_North.add(jt\_3\_1);

MyButton myButton\_Select\_3\_1=new MyButton("Browse",430,20,80,20);

jPanel\_UCCF\_North.add(myButton\_Select\_3\_1);

myButton\_Select\_3\_1.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

JFileChooser fcDlg = new JFileChooser();

fcDlg.setDialogTitle("Please select the location where the UCCF is saved");

fcDlg.setFileSelectionMode(JFileChooser.DIRECTORIES\_ONLY);

int returnVal = fcDlg.showOpenDialog(null);

if (returnVal == JFileChooser.APPROVE\_OPTION) {

String filepath = fcDlg.getSelectedFile().getPath();

UCCF\_Save\_Location=filepath;

jt\_3\_1.setText(UCCF\_Save\_Location);

jTextArea\_Loger\_Info.append(

"The save location of UCCF!"+filepath+"\r\n");

}

}

});

MyButton myButton\_Cancel\_3\_1=new MyButton("Cancel",515,20,80,20);

jPanel\_UCCF\_North.add(myButton\_Cancel\_3\_1);

myButton\_Cancel\_3\_1.addActionListener(new CancelActionListener(jt\_3\_1));

JLabel jLabel\_3\_2=new JLabel("Input UCCF File Name:");

jLabel\_3\_2.setBounds(8,50,140,20);

jPanel\_UCCF\_North.add(jLabel\_3\_2);

final JTextField jt\_3\_2=new JTextField();

jt\_3\_2.setBounds(135,50,290,20);

jPanel\_UCCF\_North.add(jt\_3\_2);

MyButton myButton\_create\_UCCF=new MyButton("UCCF",430,50,80,20);

jPanel\_UCCF\_North.add(myButton\_create\_UCCF);

myButton\_create\_UCCF.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

JPanel jPanel\_UCCF\_South=new JPanel(null);

jPanel\_UCCF\_South.setPreferredSize(new Dimension(0,440));

jPanel\_UCCF\_South.setBorder(new TitledBorder(UIManager.getBorder("TitledBorder.border"), "UCCF Info", TitledBorder.CENTER, TitledBorder.TOP, null, null));

JTextArea jTextArea\_UCCF\_Info=new JTextArea();

jTextArea\_UCCF\_Info.setFont(jTextArea\_Font);

JScrollPane jScrollPane\_UCCF\_Info=new JScrollPane(jTextArea\_UCCF\_Info);

jScrollPane\_UCCF\_Info.setBounds(5,20,595,410);

jPanel\_UCCF\_South.add(jScrollPane\_UCCF\_Info);

jPanel\_UCCF.add(jPanel\_UCCF\_South,BorderLayout.SOUTH);

UCCF\_File\_Name=jt\_3\_2.getText();

String all\_uccf\_location=UCCF\_Save\_Location+"\\"+UCCF\_File\_Name+".uccf";

UccfInof u=new UccfInof(arrayLists\_VP);

try {

CreateUCCF c=new CreateUCCF(all\_uccf\_location,u.s\_info);

} catch (IOException e1) {

e1.printStackTrace();

}

File f=new File(all\_uccf\_location);

try {

Read\_UCCF ru=new Read\_UCCF(f,arrayLists\_VP.size()+2);

jTextArea\_UCCF\_Info.append(ru.result);

} catch (IOException e1) {

e1.printStackTrace();

}

jPanel\_Center.repaint();

}

});

MyButton myButton\_Cancel\_3\_2=new MyButton("Cancel",515,50,80,20);

jPanel\_UCCF\_North.add(myButton\_Cancel\_3\_2);

myButton\_Cancel\_3\_2.addActionListener(new CancelActionListener(jt\_3\_2));

JPanel jPanel\_South=new JPanel(new BorderLayout());

jPanel\_South.setPreferredSize(new Dimension(0,140));

JTabbedPane jTabbedPane2=new JTabbedPane();

jPanel\_South.add(jTabbedPane2,BorderLayout.NORTH);

JPanel jPanel\_Loger\_Info=new JPanel(new BorderLayout());

jTabbedPane2.add("Logger Info",jPanel\_Loger\_Info);

jTextArea\_Loger\_Info=new JTextArea();

jTextArea\_Loger\_Info.setFont(jTextArea\_Font);

JScrollPane jScrollPane1=new JScrollPane(jTextArea\_Loger\_Info);

jScrollPane1.setPreferredSize(new Dimension(0,110));

jPanel\_Loger\_Info.add(jScrollPane1,BorderLayout.NORTH);

JPanel jPanel\_FM\_XMI=new JPanel(new BorderLayout());

jTabbedPane2.add("FM XMI",jPanel\_FM\_XMI);

jTextArea\_FM\_XMI=new JTextArea();

jTextArea\_FM\_XMI.setFont(jTextArea\_Font);

JScrollPane jScrollPane2=new JScrollPane(jTextArea\_FM\_XMI);

jScrollPane2.setPreferredSize(new Dimension(0,110));

jPanel\_FM\_XMI.add(jScrollPane2,BorderLayout.NORTH);

this.add(jPanel\_North,BorderLayout.NORTH);

this.add(jPanel\_Center,BorderLayout.CENTER);

this.add(jPanel\_South,BorderLayout.SOUTH);

this.setVisible(true);

this.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

this.setLocationRelativeTo(null);

}

public static void main(String[] args) {

Main\_ATL m=new Main\_ATL();

}

}

package com.ustb.cn;

import java.io.IOException;

import java.util.ArrayList;

import java.util.Iterator;

import java.util.List;

import java.util.Map;

import java.util.Set;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import javax.servlet.http.HttpSession;

public class MRIter extends HttpServlet{

private void autoMethodIter(HttpServletRequest request,

HttpServletResponse response, HttpSession session,

String operationName, WsdlOperationFormat opFormat,

List<MetamorphicRelation> mrList) throws IOException {

String numStr = request.getParameter("num\_text");

int num = Integer.parseInt(numStr);

if (((String) session.getAttribute("operationName"))

.equals("transfer") && num > 0) {

int mrID = 0;

int id = 1;

List<TCTransfer> tcsTransferList = null;

List<TCTransferF> tcsTransferFList= null;

List<List<String>> tcsListS = null;// 原始测试用例

List<List<String>> tcsListF = null;//衍生测试用例

if (session.getAttribute("tcsTransferList") == null) {

System.out.println("第一个tc");

tcsTransferList = new ArrayList<TCTransfer>();

tcsTransferFList = new ArrayList<TCTransferF>();

tcsListS = new ArrayList<List<String>>();

tcsListF = new ArrayList<List<String>>();

} else {

tcsTransferList = (List<TCTransfer>) session

.getAttribute("tcsTransferList");

tcsTransferFList = (List<TCTransferF>) session

.getAttribute("tcsTransferFList");

tcsListS = (List<List<String>>) session.getAttribute("tcsList");

// TODO

tcsListF = (List<List<String>>) session

.getAttribute("tcsFList");

id = tcsTransferList.size() + 1;

}

XmlTestCaseGenerator tcGen = new XmlTestCaseGenerator();

for (int mrCount = 0; mrCount < num; mrCount++) {

TCTransfer tempTC=null;

TCTransferF tempTCF=null;

mrID=0;

Iterator<MetamorphicRelation> iterMR = mrList.iterator();

while (iterMR.hasNext()) {

mrID++;

MetamorphicRelation mr = iterMR.next();

if (mrID == 1 && mrCount == 0) {

XmlTestCase newTC = tcGen.genWithLimitation(

operationName, opFormat.getInput().format, mr

.getRelationOfInput(), mrID);

Set<Map.Entry<XmlVariable, String>> entry = newTC

.getInput().entrySet();

Iterator<Map.Entry<XmlVariable, String>> iter = entry

.iterator();

Map.Entry<XmlVariable, String> myentry = null;

int i = 0;

String values[] = new String[entry.size()];

while (iter.hasNext()) {

myentry = iter.next();

values[i] = myentry.getValue();

i++;

}

TCTransfer tcTransfer=new TCTransfer(id, mrID, values[0],

values[1], values[2], values[3]);

tempTC=new TCTransfer(id, mrID, values[0],

values[1], values[2], values[3]);

tcsTransferList.add(tcTransfer);

TCTransferF tcTransferF=TCTransferF.generateFollowTC(tcTransfer,opFormat,mr);

tempTCF=tcTransferF;

tcsTransferFList.add(tcTransferF);

}else if(mrID==1){

TCTransferF tcTransferF=tcsTransferFList.get(mrCount-1);

TCTransfer tcTransfer=tcTransferF.tranferToS();

tcTransfer.setId(id);

tcTransfer.setMrID(mrID);

tempTC = new TCTransfer(tcTransfer.getId(), tcTransfer

.getMrID(), tcTransfer.getAccountFrom(),

tcTransfer.getAccountTo(),

tcTransfer.getMode(), tcTransfer.getAmount());

tcsTransferList.add(tcTransfer); tcTransferF=TCTransferF.generateFollowTC(tcTransfer,opFormat,mr);

tempTCF=tcTransferF;

tcsTransferFList.add(tcTransferF);

}else {

tempTC.setMrID(mrID);

tempTC.setId(id);

TCTransfer tcTransfer = new TCTransfer(tempTC.getId(),

tempTC.getMrID(), tempTC.getAccountFrom(),

tempTC.getAccountTo(), tempTC.getMode(), tempTC

.getAmount());

tcsTransferList.add(tcTransfer);

TCTransferF tcTransferF = TCTransferF.generateFollowTC(

tcTransfer, opFormat, mr);

tempTCF = tcTransferF;

tcsTransferFList.add(tcTransferF);

}

List<String> tcList = new ArrayList<String>();

tcList.add(String.valueOf(id));

tcList.add(String.valueOf(mrID));

tcList.add(tempTC.getAccountFrom());

tcList.add(tempTC.getAccountTo());

tcList.add(tempTC.getMode());

tcList.add(tempTC.getAmount());

tcsListS.add(tcList);

List<String> tcListF = new ArrayList<String>();

tcListF.add(String.valueOf(tempTCF.getsID()));//有待改进

tcListF.add(String.valueOf(mrID));

tcListF.add(tempTCF.getAccountFrom());

tcListF.add(tempTCF.getAccountTo());

tcListF.add(tempTCF.getMode());

tcListF.add(tempTCF.getAmount());

tcListF.add(String.valueOf(tempTCF.getsID()));

tcsListF.add(tcListF);

id++;

}

}

session.setAttribute("tcsList", tcsListS);

session.setAttribute("tcsTransferList", tcsTransferList);

session.setAttribute("tcsFList", tcsListF);

session.setAttribute("tcsTransferFList", tcsTransferFList);

SqlUtils sqlUtils=SqlUtils.getInstance();

sqlUtils.insertToTransferS(tcsTransferList);

sqlUtils.insertToTransferF(tcsTransferFList);

response.sendRedirect("4.start\_test.jsp");

} else if (((String) session.getAttribute("operationName"))

.equals("matchQuakes") && num > 0) {

} else if (((String) session.getAttribute("operationName")) .equals("rmbToGbbob") && num > 0) {

} else {

System.out.println("跳转错误！！！");

}

}

private void autoMethodIter\_1(HttpServletRequest request,

HttpServletResponse response, HttpSession session,

String operationName, WsdlOperationFormat opFormat,

List<MetamorphicRelation> mrList) throws IOException {

String numStr = request.getParameter("num\_text");

int num = Integer.parseInt(numStr);

if (((String) session.getAttribute("operationName"))

.equals("transfer") && num > 0) {

int mrID = 0;

int id = 1;

List<TCTransfer> tcsTransferList = null;

List<TCTransferF> tcsTransferFList= null;

List<List<String>> tcsListS = null;

List<List<String>> tcsListF = null;

if (session.getAttribute("tcsTransferList") == null) {

System.out.println("第一个tc");

tcsTransferList = new ArrayList<TCTransfer>();

tcsTransferFList = new ArrayList<TCTransferF>();

tcsListS = new ArrayList<List<String>>();

tcsListF = new ArrayList<List<String>>();

} else {

tcsTransferList = (List<TCTransfer>) session

.getAttribute("tcsTransferList");

tcsTransferFList = new ArrayList<TCTransferF>();

tcsListS = (List<List<String>>) session.getAttribute("tcsList");

tcsListF = new ArrayList<List<String>>();

id = tcsTransferList.size() + 1;

}

XmlTestCaseGenerator tcGen = new XmlTestCaseGenerator();

for (int mrCount = 0; mrCount < num; mrCount++) {

TCTransfer tempTC=null;

TCTransferF tempTCF=null;

mrID=0;

Iterator<MetamorphicRelation> iterMR = mrList.iterator();

while (iterMR.hasNext()) {

mrID++;

MetamorphicRelation mr = iterMR.next();

if (mrID == 1 && mrCount == 0) {

TCTransfer tcTransfer=null;

if(tcsTransferList.size()!=0){

tcTransfer=tcsTransferList.get(0);

} else {

XmlTestCase newTC = tcGen.genWithLimitation(

operationName, opFormat.getInput().format,

mr.getRelationOfInput(), mrID);

Set<Map.Entry<XmlVariable, String>> entry = newTC

.getInput().entrySet();

Iterator<Map.Entry<XmlVariable, String>> iter = entry

.iterator();

Map.Entry<XmlVariable, String> myentry = null;

int i = 0;

String values[] = new String[entry.size()];

while (iter.hasNext()) {

myentry = iter.next();

values[i] = myentry.getValue();

i++;

}

tcTransfer = new TCTransfer(id, mrID, values[0],

values[1], values[2], values[3]);

tempTC = new TCTransfer(id, mrID, values[0],

values[1], values[2], values[3]);

tcsTransferList.add(tcTransfer);

}

TCTransferF tcTransferF=TCTransferF.generateFollowTC(tcTransfer,opFormat,mr);

tempTCF=tcTransferF;

tcsTransferFList.add(tcTransferF);

}else if(mrID==1){

TCTransferF tcTransferF=tcsTransferFList.get(mrCount-1);

TCTransfer tcTransfer=tcTransferF.tranferToS();

tcTransfer.setId(id);

tcTransfer.setMrID(mrID);

tempTC = new TCTransfer(tcTransfer.getId(), tcTransfer

.getMrID(), tcTransfer.getAccountFrom(),

tcTransfer.getAccountTo(),

tcTransfer.getMode(), tcTransfer.getAmount());

tcsTransferList.add(tcTransfer);

tcTransferF=TCTransferF.generateFollowTC(tcTransfer,opFormat,mr);

tempTCF=tcTransferF;

tcsTransferFList.add(tcTransferF);

}else {

tempTC.setMrID(mrID);

tempTC.setId(id);

TCTransfer tcTransfer = new TCTransfer(tempTC.getId(),

tempTC.getMrID(), tempTC.getAccountFrom(),

tempTC.getAccountTo(), tempTC.getMode(), tempTC

.getAmount());

tcsTransferList.add(tcTransfer);

TCTransferF tcTransferF = TCTransferF.generateFollowTC(

tcTransfer, opFormat, mr);

tempTCF = tcTransferF;

tcsTransferFList.add(tcTransferF);

}

List<String> tcList = new ArrayList<String>();

tcListF.add(tempTCF.getAmount());

tcListF.add(String.valueOf(tempTCF.getsID()));

tcsListF.add(tcListF);

tcList.add(String.valueOf(id));

tcList.add(String.valueOf(mrID));

tcList.add(tempTC.getAccountFrom());

tcList.add(tempTC.getAccountTo());

tcList.add(tempTC.getMode());

tcList.add(tempTC.getAmount());

tcsListS.add(tcList);

List<String> tcListF = new ArrayList<String>();

tcListF.add(String.valueOf(tempTCF.getsID()));//有待改进

tcListF.add(String.valueOf(mrID));

tcListF.add(tempTCF.getAccountFrom());

tcListF.add(tempTCF.getAccountTo());

tcListF.add(tempTCF.getMode());

tcListF.add(tempTCF.getAmount());

tcListF.add(String.valueOf(tempTCF.getsID()));

tcsListF.add(tcListF);

id++;

}

}

session.setAttribute("tcsList", tcsListS);

session.setAttribute("tcsTransferList", tcsTransferList);

session.setAttribute("tcsFList", tcsListF);

session.setAttribute("tcsTransferFList", tcsTransferFList);

SqlUtils sqlUtils=SqlUtils.getInstance();

sqlUtils.insertToTransferS(tcsTransferList);

sqlUtils.insertToTransferF(tcsTransferFList);

response.sendRedirect("4.start\_test.jsp");

}else if (((String) session.getAttribute("operationName"))

.equals("matchQuakes") && num > 0) {

} else if (((String) session.getAttribute("operationName")

.equals("rmbToGbbob") && num > 0) {

} else {

System.out.println("跳转错误！！！");

}

}

}

package com.ustb.cn;

import com.ustb.cn.BussinessProcess.ProcessTree;

import com.ustb.cn.CreateUCCF.CreateUCCF;

import com.ustb.cn.CreateUCCF.Read\_UCCF;

import com.ustb.cn.CreateUCCF.UccfInof;

import com.ustb.cn.DrawFMPic.DrawFMDiagram;

import com.ustb.cn.ModelTransformation.ModelTran;

import com.ustb.cn.ValidateConstraints.Validate;

import org.dom4j.DocumentException;

public class Main\_Test2 extends JFrame{

FMTreeJPanel fmTree;

String Selected\_String;

Font jTextArea\_Font=new Font("Consolas",Font.PLAIN,12);

String url\_FM\_FilePath;

String url\_FM\_Save\_Location;

String FMDiagram\_Name;

JTextArea jTextArea\_Loger\_Info;