Names and NSIDs: Ruida Xie (rux793); Hao Li (hal356); Yinsheng Dong (yid164)

Project: jEdit

Change Number: [5]

Date: 04/02/2018

*Everything in italic should be changed as appropriate by you and should not be italic when submitted.*

(Duplicate data when creating a new view)

1. **Change Request*:***

Currently clicking View | New View will create a new view for the same data; which means that modification in one view will affect the other one. Add menu item New View&Buffer under menu View to allow the user to duplicate data for the current shown view only.Locate where the activity.log is and add timestamps to all kinds of messages. Currently there are no timestamps in the log file.

1. **Concept Location:**

**Table 1. The list of all the classes visited during concept location.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **File name** | **Tool used** | **Located?** | **Comments** |
| 1 | Jedit\_gui.props | Read Source | Propagating & Changed | GUI related actions stored here |
| 2 | Actions.xml | Read Source | Changed | Methods linked to java code |
| 3 | Jedit.java | Read Source & Search Method | Changed | Create new view with buffer similar to openfile() |
|  |  |  |  |  |
|  |  |  |  |  |

1. **Impact Analysis:**

**Table 2. The list of all the classes visited during impact analysis.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Class name** | **Tool used** | **Impacted?** | **Comments** |
| 1 | Jedit\_gui.props | Read Source | Changed | Added menu items |
| 2 | Actions.xml | Read source | Changed | Added gui method link |
| 3 | Jedit.java | Read source & find usage | Changed | Added new method |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

1. **Learning process:** *Include a partial UML* ***class diagram.*** *The class diagram must contain all the classes that you visited and all the relations (dependencies) between the classes that you understood. (see Figure 1). Mark the classes that were changed with red text as in the example.*

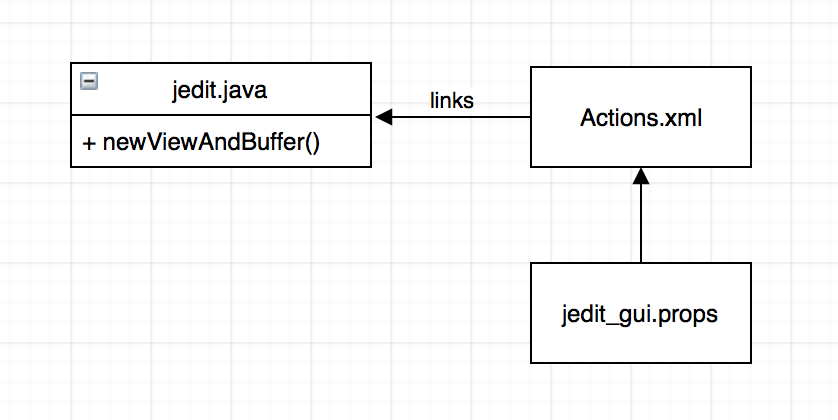
****

Figure class diagram

1. **Description of the implementation:**

I added newViewAndBuffer() method in jedit.java.

1. **Sources:** openFile() method in jedit.java*.*
2. **Highlighted Source Code:**

Actions.xml:

<ACTION NAME="new-view-and-buffer">

<CODE>

jEdit.newViewAndBuffer(view);

</CODE>

</ACTION>

Jedit\_gui.props:

new-view-and-buffer \

…

new-view-and-buffer.label=New View with Buffer

jedit.java:

//{{{ newViewAndBuffer() method

/\*\*

\* Creates a new view with a new buffer.

\* @param view An existing view

\*

\* rux793

\*/

public static View newViewAndBuffer(View view)

{

// Very similar to open file, the only differences are

// always create new view and not load from disk

String path = view.getBuffer().getDirectory();

Hashtable props = new Hashtable();

Buffer newBuffer;

synchronized (editBusOrderingLock)

{

synchronized(bufferListLock)

{

Buffer buffer = getBuffer(path);

// if(buffer != null)

// {

// if(view != null)

// view.setBuffer(buffer);

//

// return buffer;

// }

//

// if(props == null)

// props = new Hashtable();

BufferHistory.Entry entry = BufferHistory.getEntry(path);

if(entry != null && saveCaret && props.get(Buffer.CARET) == null)

{

props.put(Buffer.CARET, entry.caret);

/\* if(entry.selection != null)

{

// getSelection() converts from string to

// Selection[]

props.put(Buffer.SELECTION,entry.getSelection());

} \*/

}

if(entry != null && props.get(JEditBuffer.ENCODING) == null)

{

if(entry.encoding != null)

props.put(JEditBuffer.ENCODING,entry.encoding);

}

newBuffer = new Buffer(path,true,false,props);

if(!newBuffer.load(view,false))

return null;

addBufferToList(newBuffer);

}

EditBus.send(new BufferUpdate(newBuffer,view,BufferUpdate.CREATED));

}

// Create a new view and return

View newViewWithBuffer = newView(view, null, false);

newViewWithBuffer.setBuffer(newBuffer);

return newViewWithBuffer;

} //}}}