

Change request log

1. Team

Nick Lawrence and Satya Kalyan

2. Change Request

PDFsam Change Request #1

The Alternate Mix and Merge modules of PDFsam provide buttons to move a document up or down in their respective lists of files. However, these buttons are not ideal when the list of documents is long, and the user wants to move one of them to the top or the bottom. You are requested to add two new buttons in these modules, to allow the user to move a selected document to the top and bottom of the list.

3. Concept Location

Step #	Description	Rationale
1	<i>Found SelectionTableToolbar from "Move Up" text search</i>	
2	<i>Discovered the MoveUpButton and MoveDownButton subclasses and where they were used.</i>	
3	<i>It was determined new classes must be added and the SelectionTableToolbar constructor must be changed to add the new buttons</i>	

Time spent (in minutes): 15

4. Impact Analysis

Step #	Description	Rationale
1	<i>Found all uses of SelectionTableToolbar which led to MultipleSelectionPane.</i>	<i>The SelectionTableToolbar will be modified, so anywhere that uses it may need to be changed.</i>
2	<i>MultipleSelectionPane was discovered to not need changes, but could propagate changes, so all uses of MultipleSelectionPane were discovered.</i>	
3	<i>The four uses of MultipleSelectionPane were AlternateMixSelectionPane, ExtractSelectionPane, MergeSelectionPane, and RotateSelectionPane.</i>	
4	<i>The desired changes were on the AlternateMix module and the Merge module, and those were determined to be the only ones that would be changed. Therefore no further changes needed to be made in any other classes.</i>	

Time spent (in minutes): 20

5. Prefactoring

Step #	Description	Rationale
1	<i>No classes were modified or methods changed in the prefactoring step.</i>	<i>It was determined that the changes would be minimal and that no other areas needed to be modified.</i>

Time spent (in minutes): 0

6. Actualization

Step #	Description	Rationale
1	<i>The MoveTopButton subclass was added by copying and modifying MoveUpButton</i>	<i>The MoveTopButton would need to be almost exactly the same as MoveUpButton, but with different text and a different MoveType</i>
2	<i>The MoveTopButton subclass was changed to have the text "Move Top" and use the MoveType MoveType.TOP</i>	
3	<i>The MoveBottomButton subclass was added by copying and modifying MoveDownButton</i>	<i>The MoveBottomButton would need to be almost exactly the same as MoveDownButton, but with different text and a different MoveType</i>
4	<i>The MoveBottomButton subclass was changed to have the text "Move Bottom" and use the MoveType MoveType.BOTTOM</i>	
5	<i>The SelectionTableToolbar constructor was modified so it added an instance of MoveTopButton and an instance of MoveBottomButton to the toolbar</i>	

Time spent (in minutes): 15

7. Postfactoring

Step #	Description	Rationale

1	<i>No classes were modified or methods changed in the postfactoring step.</i>	
----------	---	--

Time spent (in minutes): 0

8. Validation

Step #	Description	Rationale
1	<i>All tests provided with PDFsam were run as a regression test.</i>	<i>This is the regular expected behavior. The test passed.</i>
2	<i>Manual tests were done by creating merge requests with many files and using the “Move Top” and “Move Bottom” buttons to verify they worked correctly.</i>	<i>This was to verify the changes made had the expected result, which was the case in all manual tests.</i>

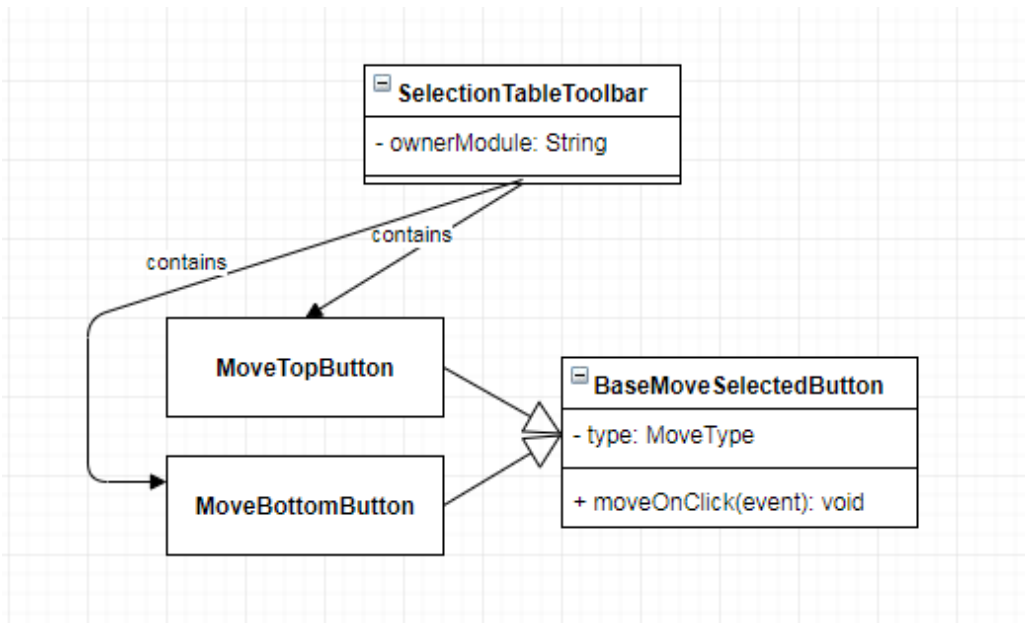
Time spent (in minutes): 40

9. Timing

Phase Name	Time (in minutes)
Concept location	15
Impact Analysis	20
Prefactoring	0
Actualization	15
Postfactoring	0

Verification	40
Total	90

10. Reverse engineering



11. Conclusions

For this change request, all steps were relatively simple due to the easy nature of the change and the layout of the code. Concept location was made easy due to the ability to text search using the text inside the nearby buttons. Impact analysis was also relatively simple due to the layout of the project and the ability for eclipse to find all uses of a specific class in a project. The actual modifications to the code were trivial since the “Move Top” and “Move Bottom” operations were already implemented in MoveType. All that needed to be done was the creation of new buttons that used these existing MoveTypes. Testing the changes was also simple due to the limited possible impact and our experience in testing software professionally.

Classes and methods changed:

- `org/pdfsam/ui/selection/multiple/SelectionTableToolbar`
 - Created Class `MoveTopButton`
 - Created Class `MoveBottomButton`
 - Modified `SelectionTableToolbar(String ownerModule, boolean canMove)`

Final Notes

Nick Lawrence implemented the changes in the code and Satya Kalyan handled the documentation.

In the PDFsam repository all the changes are implemented in the v3.3.5-branch branch, this was due to editing the code from the v3.3.5 Tag.