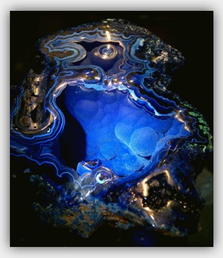
Better Backtracking Support for Programmers (www.cs.cmu.edu/~azurite/)

YoungSeok Yoon (youngseok@cs.cmu.edu)
Advised by Brad Myers (bam@cs.cmu.edu)

institute for SOFTWARE RESEARCH
Carnegie Mellon University

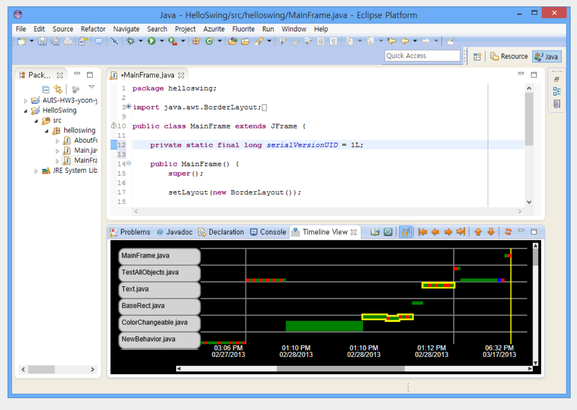
WHAT it is



AZURITE: A Selective Undo Tool for Code Editors

Adding **Z**est to **U**ndoing and **R**estoring **I**mproves **T**extual **E**xploration

Programmers need to ***backtrack*** often, yet there is only limited support for backtracking besides the conventional undo command and version control.



AZURITE is an Eclipse plug-in that allows programmers to ***selectively undo*** fine-grained code changes made in the code editor. With AZURITE, programmers can easily perform backtracking tasks, such as reverting some code to an earlier state, and restoring a block of code that was deleted a while ago.

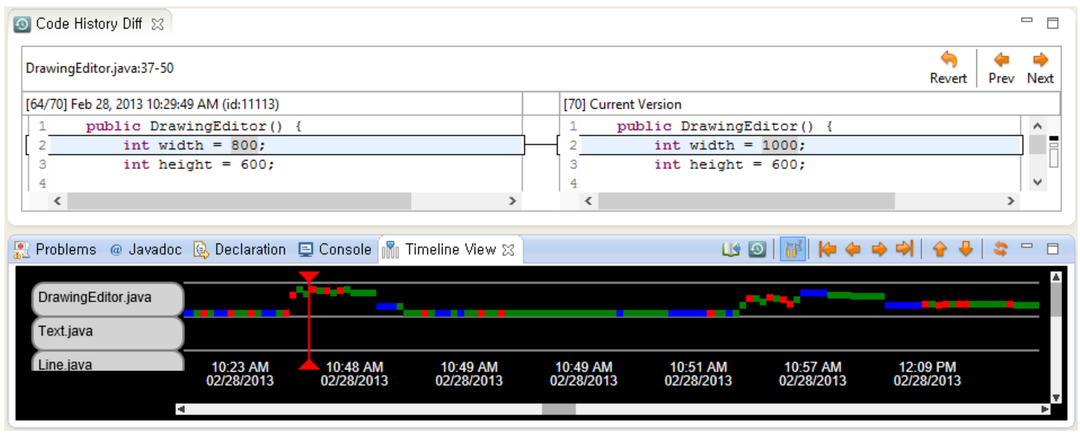
AZURITE can be used even when the desired code is not in the undo stack or the version control repository.

Backtracking: Going back to the way things were previously, for example by undoing some or all operations

Selective Undo: Undoing only the selected changes while keeping the other changes unaffected

HOW it works

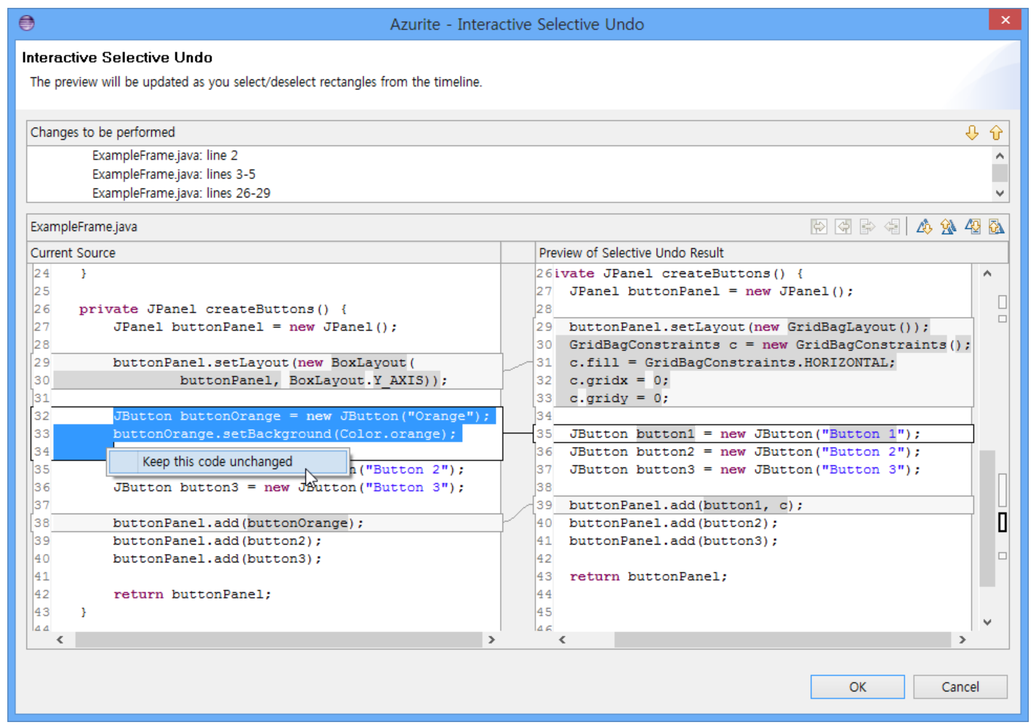
Timeline Visualization & Code History Diff View



The *timeline visualization* (bottom) is the basic UI for interacting with the code change history. The history is displayed in a two-dimensional space, and each rectangle represents an edit performed in the past. Whenever a new change is made (even by undo), the corresponding rectangle appears at the right end, so that change history is never lost. Users can select one or more rectangles in the timeline and invoke “selective undo”.

Users can select an arbitrary region of code and launch the *code history diff view* (top) to see how the selected code has changed over time, and can revert to one of the previous versions.

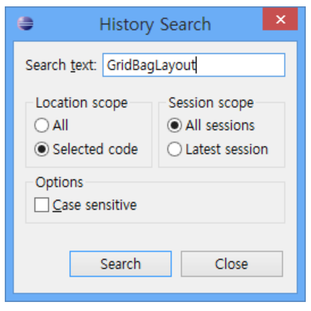
Interactive Selective Undo Dialog



The *Interactive selective undo* dialog of AZURITE. The left panel shows the current code, and the right panel shows the preview of the selective undo result. The user can also modify the outcome by marking some code in the left panel, and telling AZURITE to “keep the code unchanged”.

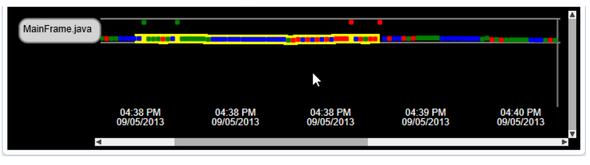
History Search

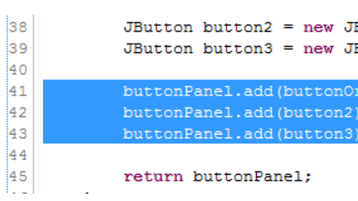
AZURITE provides various ways to *search* through the history. The goal is to enable users to express whatever they remember about the previous edits or situations that they want to select in the history.



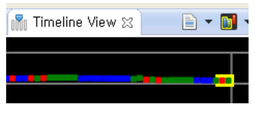
History search can even search for text which is not in the current code any more

Highlights the time interval when the given text existed in the code





Users can select an arbitrary region of code and “select corresponding rectangles”



Highlights all the edits performed on the selected region

Behind the Scenes: Selective Undo Mechanism

- Keeps track of all the fine-grained code changes
- Maintains the correct locations of the previous edits *in the current state*
- Detects conflicts among the edits and asks the user what she really wants, in case those conflicts cannot be automatically resolved

Example:

“myFontSize” —*R1*→ “myRectangleSize” —*R2*→ “myRegionArea”

What should be the result of selectively undoing only R1? myFontgionArea? myFontSize? Or do nothing and keep it myRegionArea?

WHAT we expect

- Programmers will be able to perform their daily backtracking tasks more easily
- Programmers will be more comfortable exploring, because they know they can revert incorrect changes at any time

Relevant Publications

Yoon, Y., Myers, B., and Koo, S. (2013) “Visualization of Fine-Grained Code Change History,” *IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC 2013).* **Presentation on Wednesday**
Yoon, Y. and Myers, B. (2012). “An Exploratory Study of Backtracking Strategies Used by Developers,” *International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE 2012).*