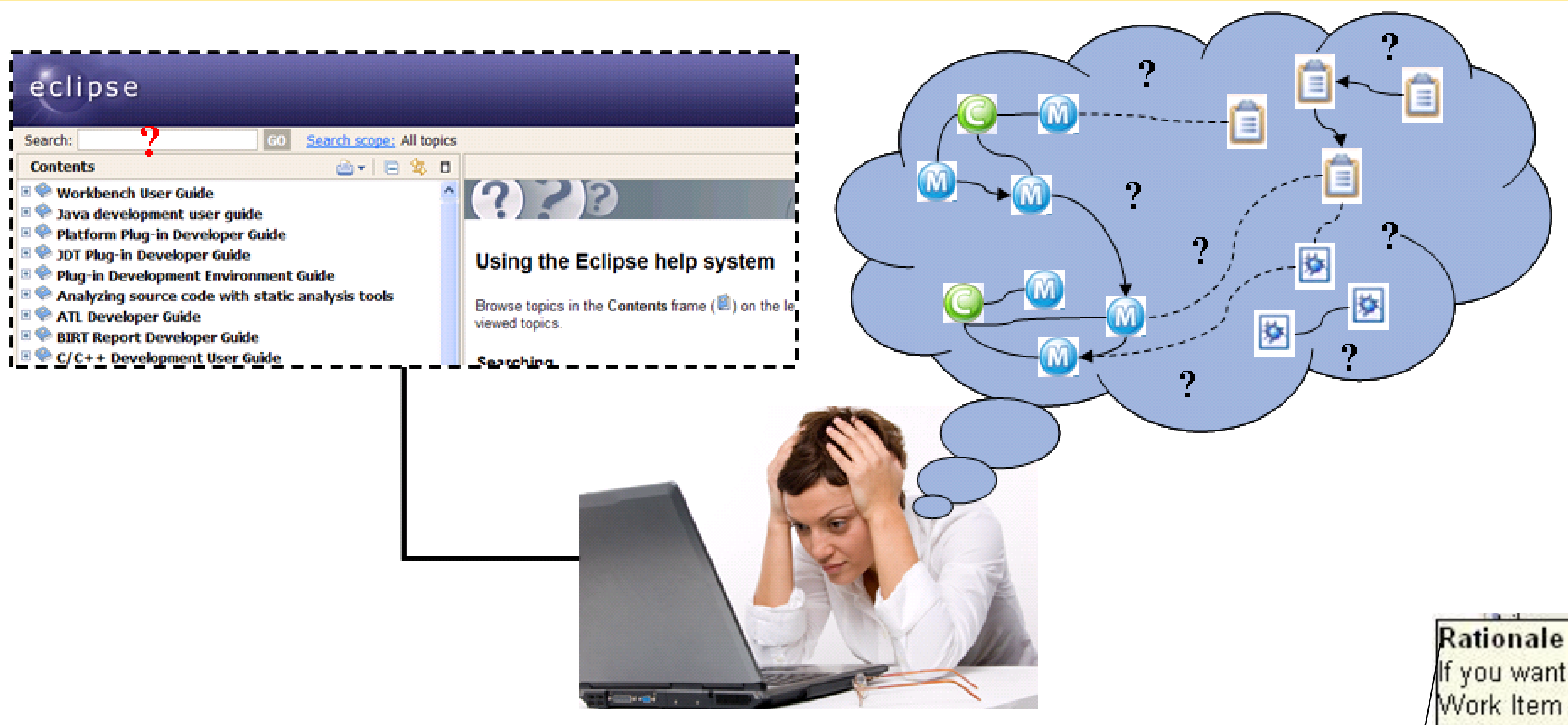
An Active Help System to Improve Program Navigation

Petcharat Viriyakattiyaporn and Gail C. Murphy
Software Practices Lab, University of British Columbia, Vancouver, Canada

Problem

- ❑ Development environments (IDEs) include many tools to aid software developers.
- ❑ Developers often do not know about all tools available.
- ❑ IDEs provide passive help systems. A developer can query for an available tool, but only if the developer knows the right keywords to express what the developer needs.


eclipse
Search: ? GO Search scope: All topics
Contents
Workbench User Guide
Java development user guide
Platform Plug-in Developer Guide
JDT Plug-in Developer Guide
Plug-in Development Environment Guide
Analyzing source code with static analysis tools
ATL Developer Guide
BIRT Report Developer Guide
C/C++ Development User Guide

Using the Eclipse help system
Browse topics in the Contents frame ( ) on the le
viewed topics.
Searching

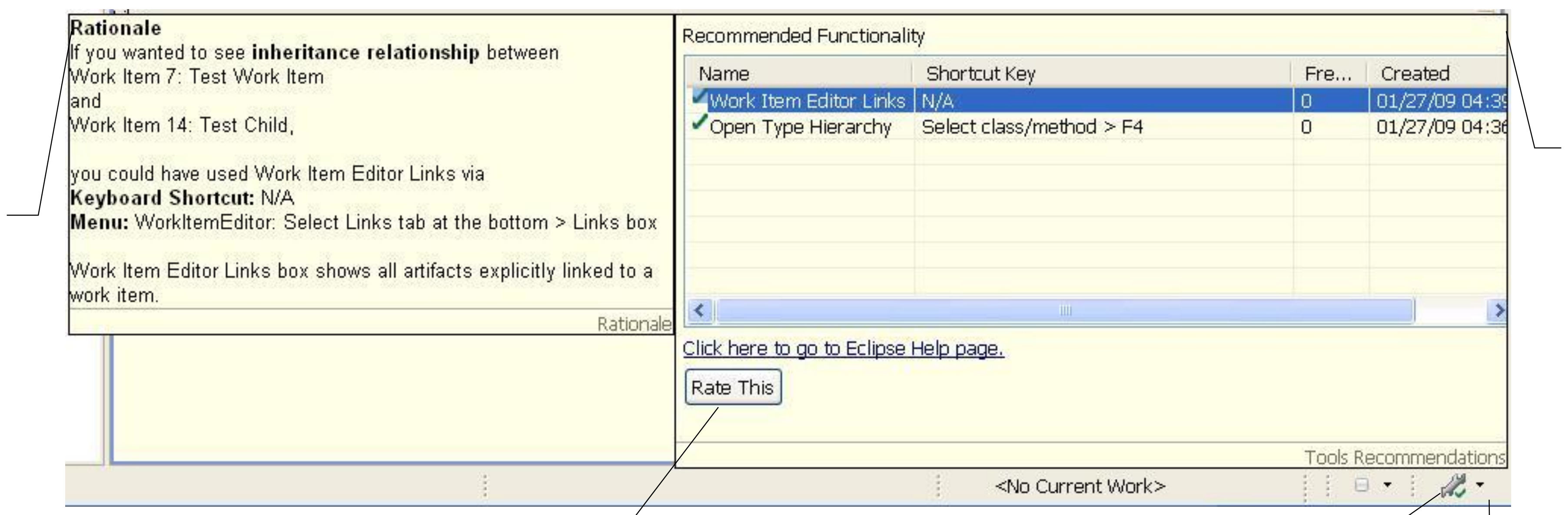
Proposed Solution: *Spyglass*

- ❑ Our Spyglass recommender system suggests appropriate navigational tools within IBM's Rationale Team Concert IDE.
- ❑ Spyglass infers navigation between Java code elements, work assignments (work items), and sets of program revisions (changesets).
- ❑ Spyglass makes recommendations when it infers that a developer is navigating sub-optimally between structurally related pieces of information.

Spyglass UI

- ❑ We carefully designed Spyglass' UI so that it presents its recommendations politely.
- ❑ The Spyglass icon, which is the entry point to all recommendations, is located in the bottom-right corner of the main IDE window.

Rationale View
This view reveals the reasoning behind a recommendation.
It appears only if the developer selects the recommendation in the Recommendation View.


Rationale
If you wanted to see **inheritance relationship** between
Work Item 7: Test Work Item
and
Work Item 14: Test Child,

you could have used Work Item Editor Links via
Keyboard Shortcut: N/A
Menu: WorkItemEditor: Select Links tab at the bottom > Links box

Work Item Editor Links box shows all artifacts explicitly linked to a work item.

Rationale

Recommended Functionality

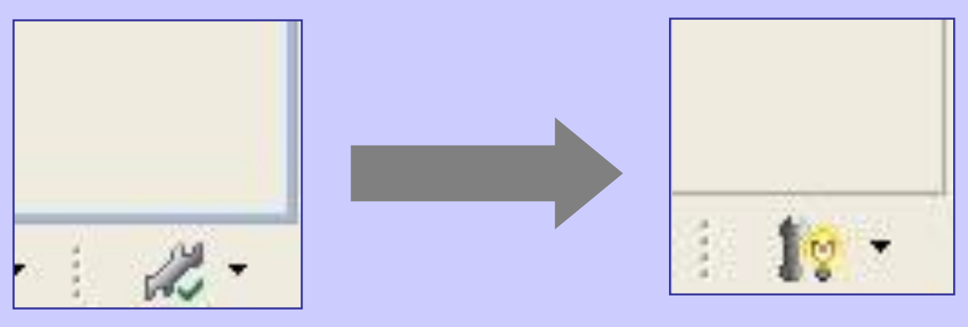
Name	Shortcut Key	Fre...	Created
✓Work Item Editor Links	N/A	0	01/27/09 04:3
✓Open Type Hierarchy	Select class/method > F4	0	01/27/09 04:3

Click here to go to Eclipse Help page.
Rate This
Tools Recommendations
<No Current Work>

Recommendation View
This view shows all unique tools recommended. It appears whenever the developer clicks on the Spyglass icon (the wrench/light bulb icon below) or the Popup View.

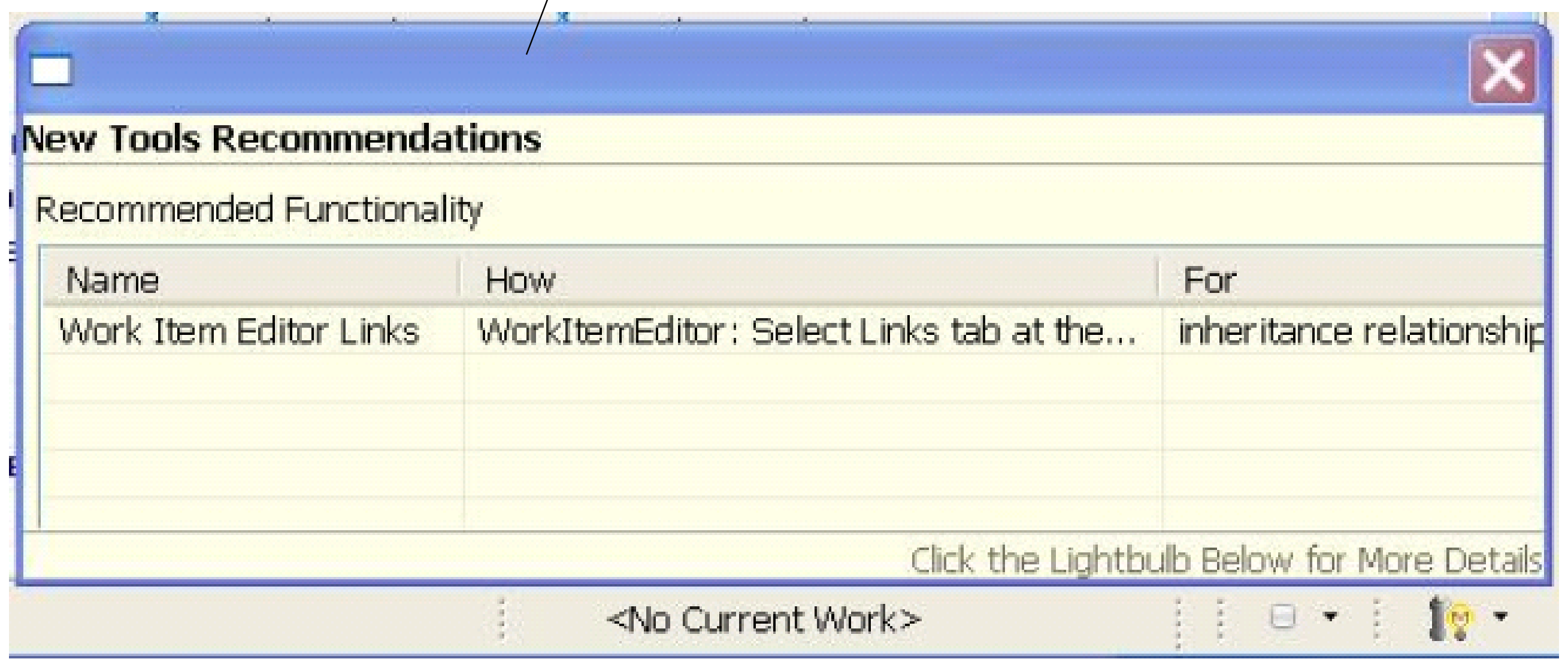
The developer can use this button to rate the usefulness of the currently selected tool.

Spyglass Icon
This icon is the entry point to Spyglass recommendations. It changes to a light bulb when there is a new recommendation for the developer.



This drop-down menu allows the developer to disable/enable the popup notification.

Popup View
When there is a new recommendation, Spyglass slides up this view above the Spyglass icon. Clicking on it opens up the Recommendation View. It closes itself in 15 seconds.


New Tools Recommendations
Recommended Functionality

Name	How	For
Work Item Editor Links	WorkItemEditor: Select Links tab at the...	inheritance relationship

Click the Lightbulb Below for More Details
<No Current Work>

Lab User Study

Participants

- ❑ 18 participants with
  - 8 months of Java
  - 2 months of Eclipse.
- ❑ Divided participants into a tutorial group and a Spyglass group.
- ❑ Independent-measures design.

Tutorial Group	Spyglass Group
❑ Read the given passive tutorial document. ❑ Completed two training tasks on Paint program.	❑ Read the given passive tutorial document (excluding the target tools). ❑ Completed two training tasks on Paint program.
❑ Completed two change tasks on JFreeChart program (with Spyglass UI disabled).	❑ Read the given Spyglass introduction. ❑ Completed two change tasks on JFreeChart program (with Spyglass UI enabled).
❑ Filled out a post-test questionnaire and had interview.	❑ Filled out a post-test questionnaire and had interview.

Results

✗ 1. Can Spyglass helps developers complete tasks more successfully?

✓ 2. Can Spyglass help navigate through pieces of information more efficiently?

○ 3. Spyglass group were aware of and used the target tools more?
- As much as the passive tutorial.

○ 4. Developers felt the recommended tools were useful?
- Yes for Open Call Hierarchy; Neutral for Open Type Hierarchy.

✗ 5. Developers felt Spyglass suggested the right tool at the right time?

○ 6. How often does Spyglass suggest the right tool at the right time?
- Half of the time (precision ~ 0.53).

✓ 7. Spyglass is not disruptive?

Conclusion & Future Work

- ❑ Spyglass needs to provide more meaningful and accurate recommendations.
- ❑ May be achievable with a more complete user model, finding the best configuration parameters for the system, and revising our algorithm.
- ❑ Explore having Spyglass recommend other types of tools.
- ❑ Thorough comparative studies between existing approaches on active help systems and our approach in a software development environment are needed.