SemDiff: Analysis & Recommendation Support for API Evolution
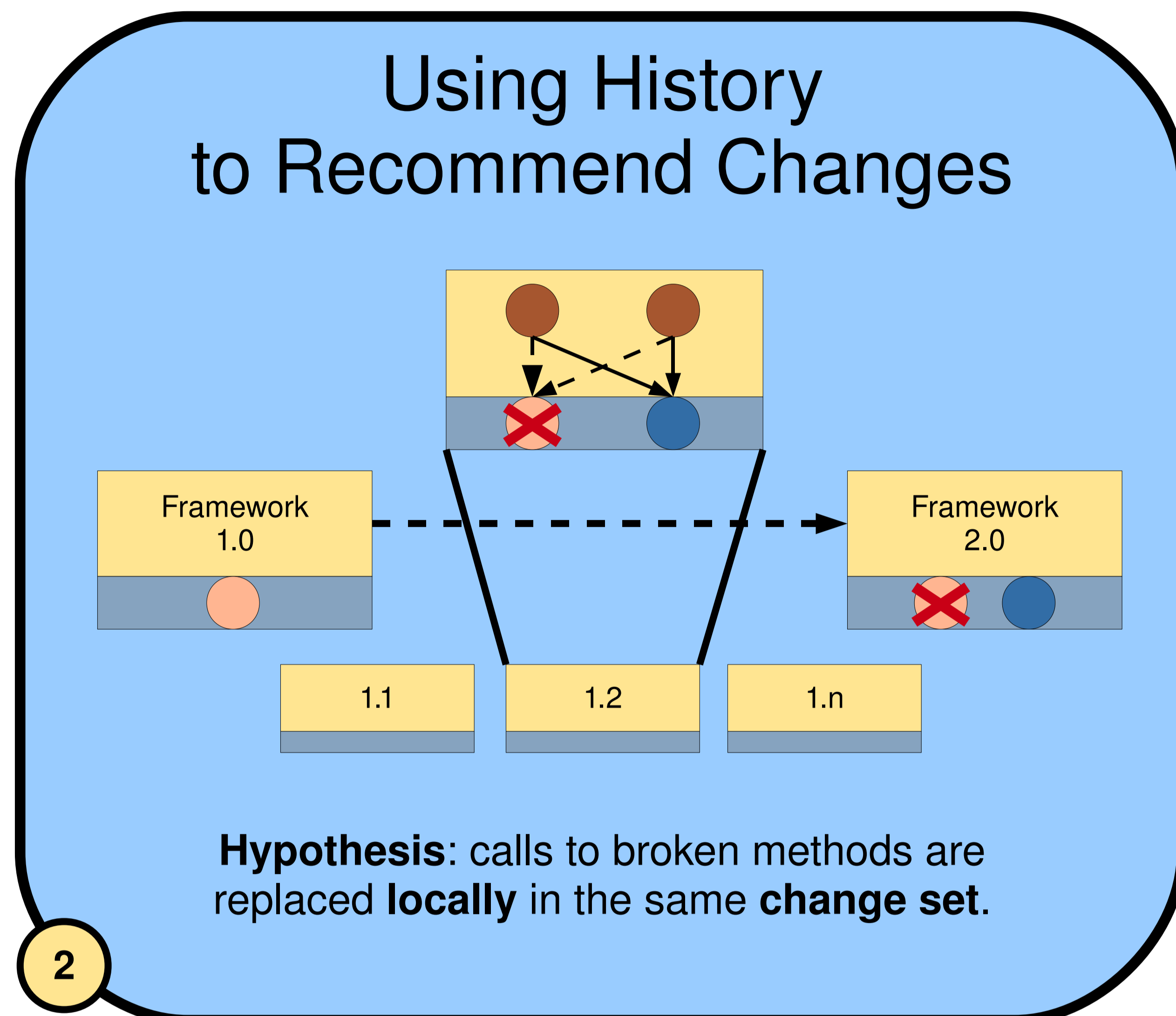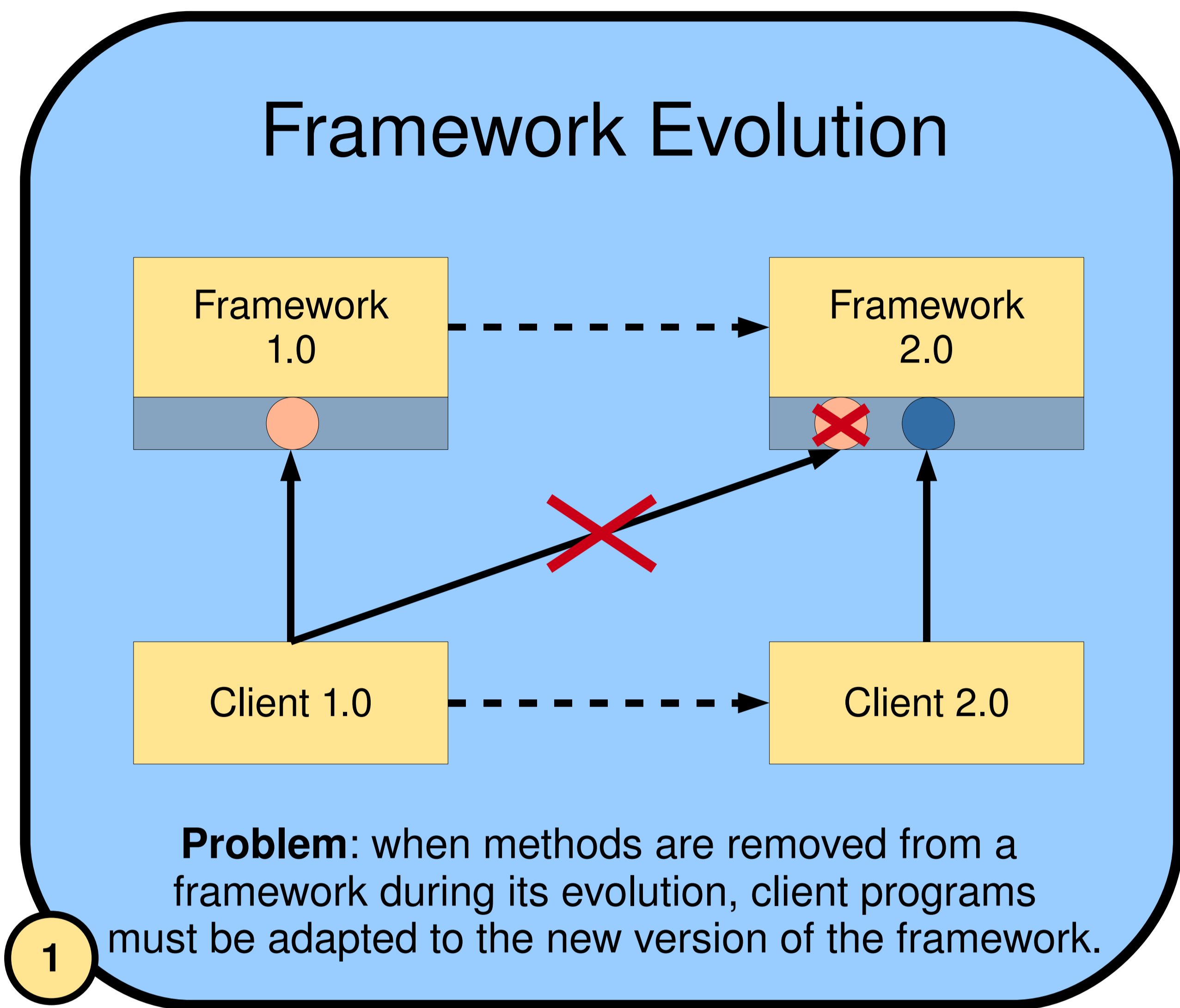
Barthélémy Dagenais and Martin P. Robillard – McGill University
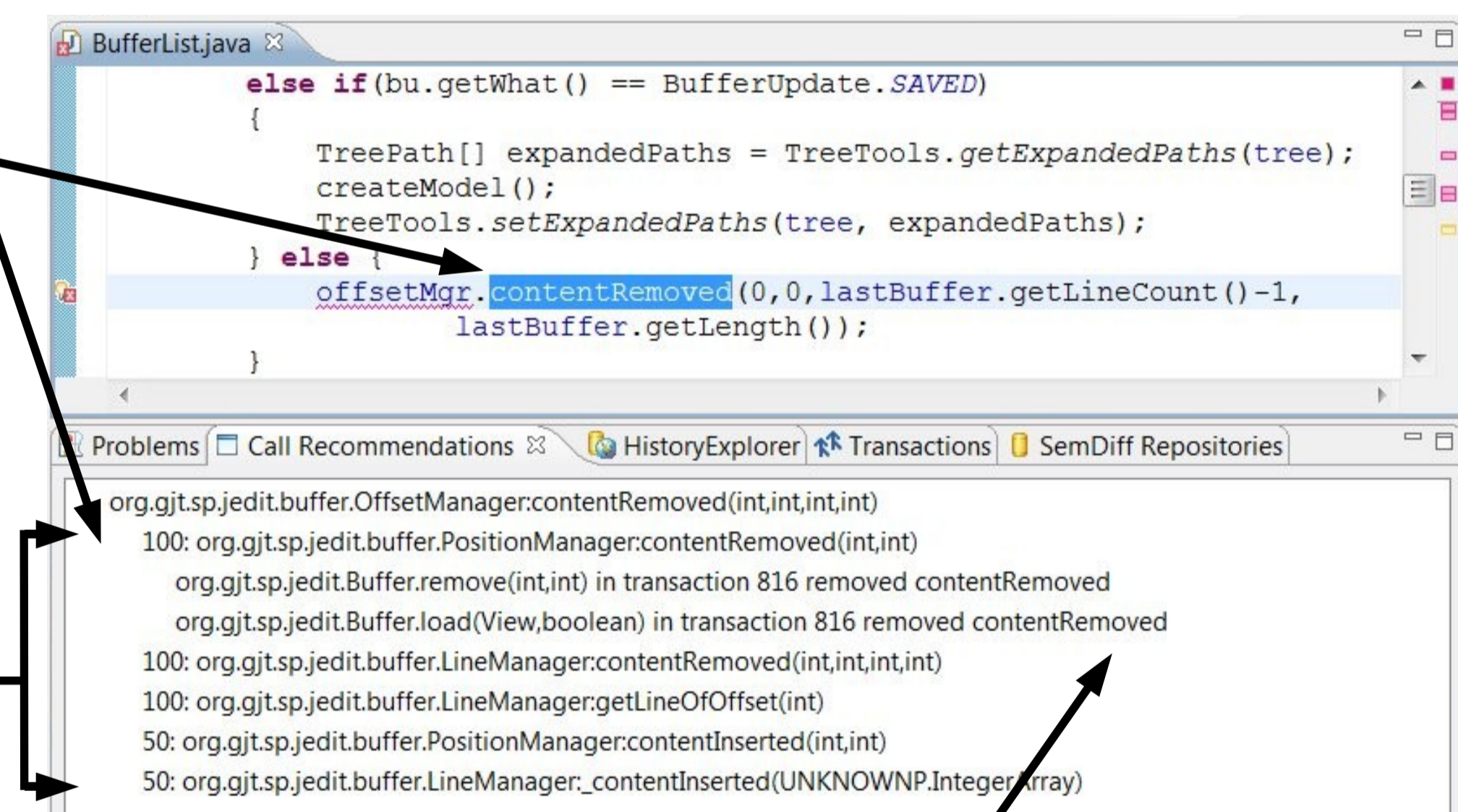http://www.cs.mcgill.ca/~swevo/semdiff/

Framework Evolution

1 **Problem**: when methods are removed from a framework during its evolution, client programs must be adapted to the new version of the framework.

Using History to Recommend Changes

2 **Hypothesis**: calls to broken methods are replaced **locally** in the same **change set**.

Requesting Recommendations

Easy Request:
Just right-click on the broken method and let SemDiff recommend relevant replacements.
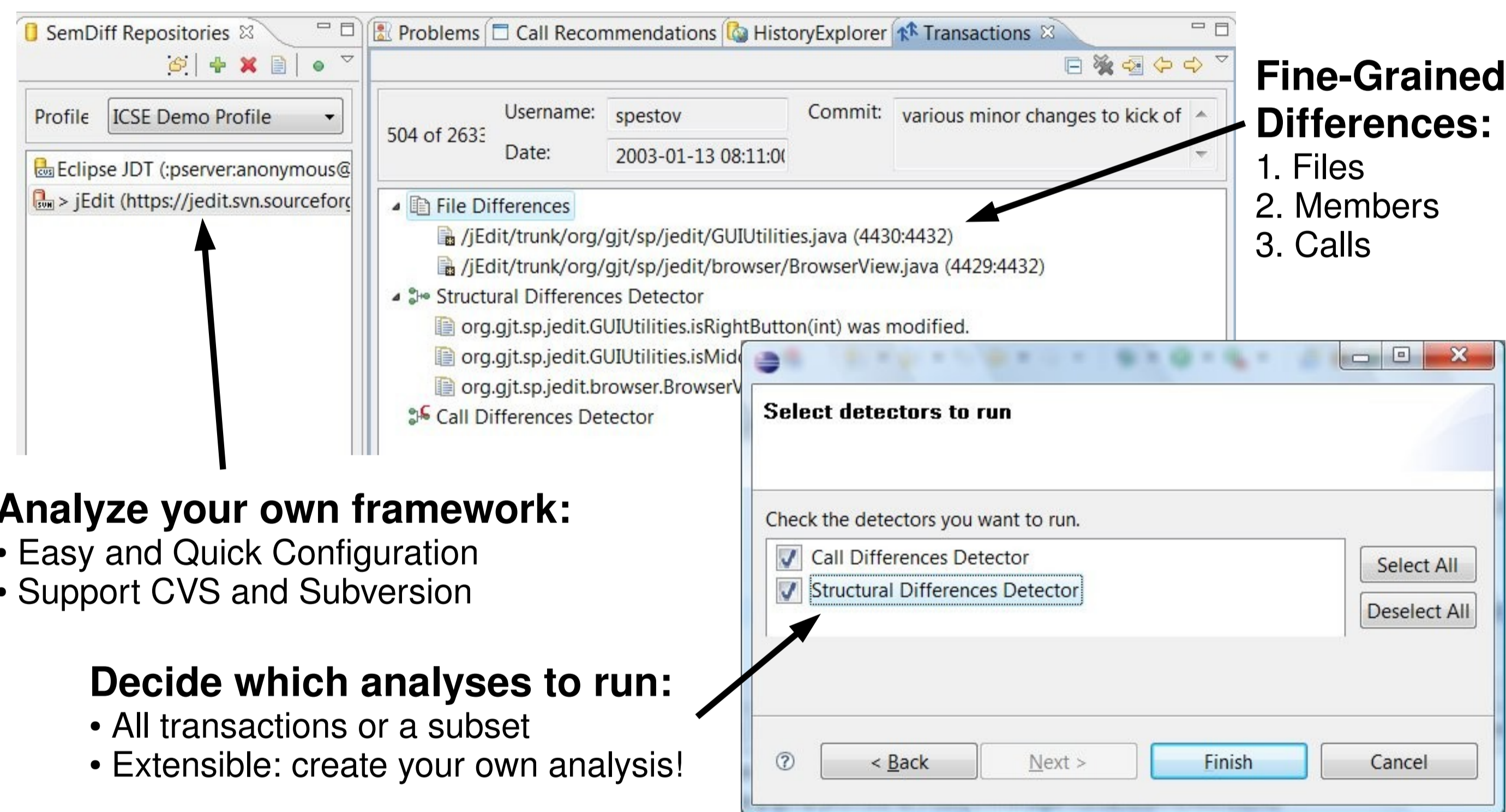

Many Alternatives:
• SemDiff detects many-to-one changes.
• SemDiff recommends potential alternatives.

Understand the recommendation:
• Double-click on the explanation to see when the call was removed and how it was replaced.
• Right-click to see the transaction in which the change happened.

3

Analyzing Source Code History

Fine-Grained Differences:
1. Files
2. Members
3. Calls

Analyze your own framework:
• Easy and Quick Configuration
• Support CVS and Subversion

Decide which analyses to run:
• All transactions or a subset
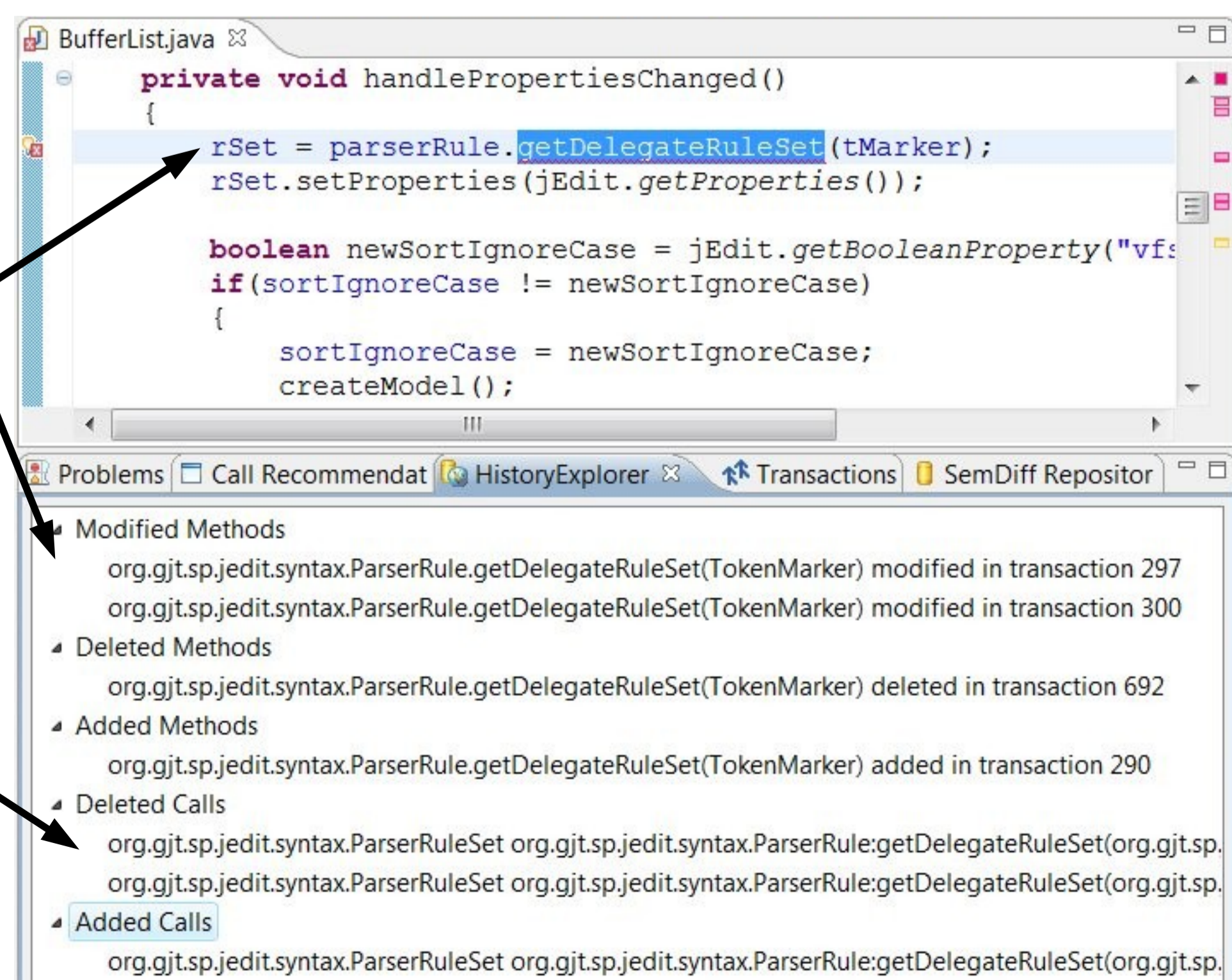• Extensible: create your own analysis!

4 You can use SemDiff to analyze a framework's history and browse though its change sets.
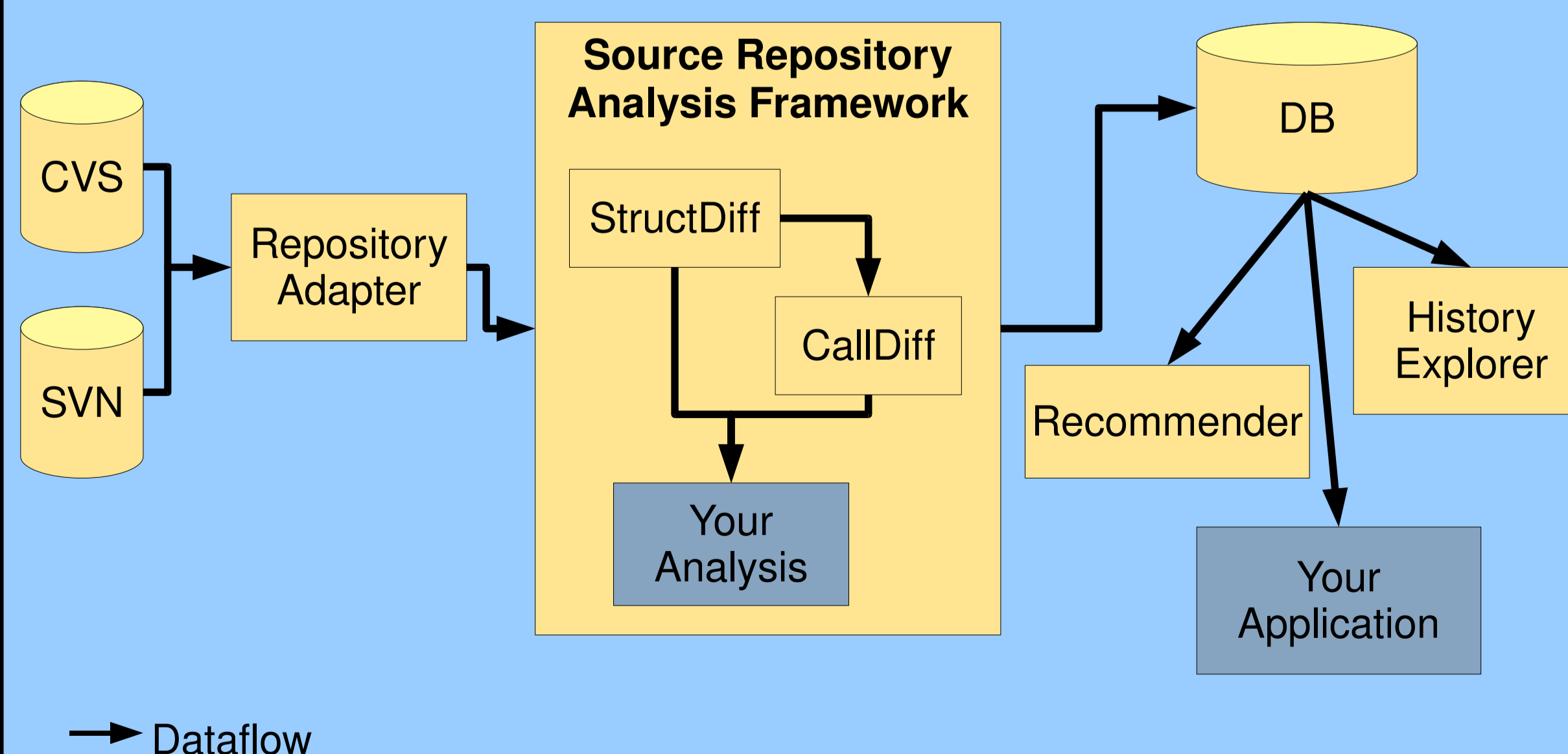
Navigating Source Code History with History Explorer

Powerful Search:
Search for all references of a method in the framework's history.

Access the context:
Double-click to open the transaction and see what led to this change.

5

An Extensible Research Platform

6 Leverage SemDiff fine-grained analyses to mine software repositories and develop your own analyses or history-based tool.