Critics: An Interactive Code Review Tool for Searching and Inspecting Systematic Changes

Tianyi Zhang,[1]   Myoungkyu Song,[2]   Miryung Kim[1]
[1] University of California, Los Angeles   [2] University of Texas at Austin

Problem Statement

Peer code reviews are widely-used quality assurance activities in software development. When using existing code review tools, it is difficult for developers to inspect *systematic changes* — similar but not identical edits scattered across different files. Developers may find it difficult to ensure that all locations are updated correctly and that there are no missing updates.

```
int keyDownEvent (int w) {
-    ExpandItem item = items [index];
     switch (w) {
         case OS.SPACE:
             Event event = new Event ();
-            event.item = item;
-            sendEvent(true, event);
+            event.item = focusItem;
+            sendEvent(event);
+            refreshItem(focusItem);
```
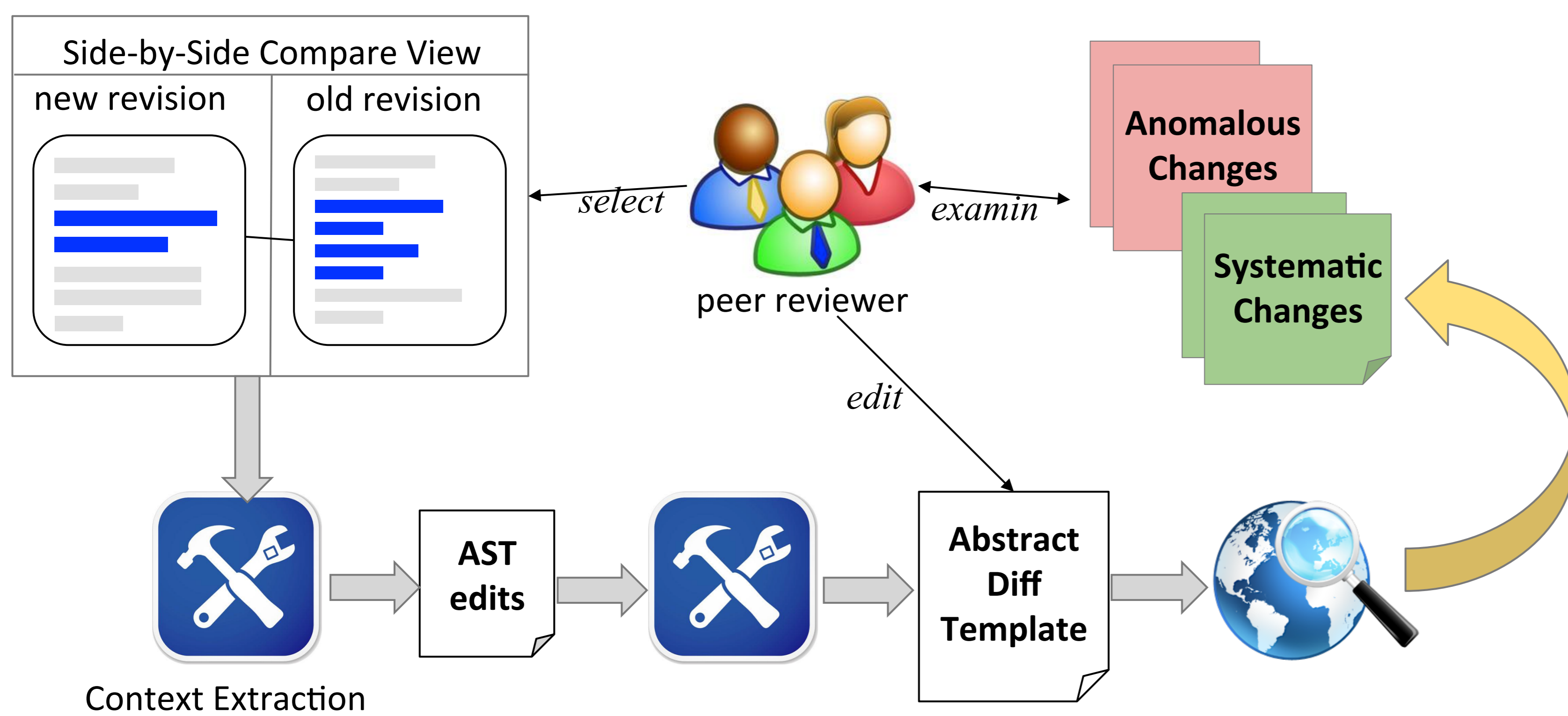
```
int keyReleaseEvent (int wParam) {
-    ExpandItem item = items [index];
     switch (wParam) {
         case OS.SPACE:
             Event ev = new Event ();
-            ev.item = item;
-            sendEvent(true, ev);
+            ev.item = focusItem;
+            sendEvent(ev);
+            refreshItem(focusItem);
```

```
int ButtonUpEvent (int wParam) {
-    ExpandItem item = items [index];
     if (wParam == HOVER){
         Event bEvent = new Event ();
-        bEvent.item = item;
-        sendEvent(true, bEvent);
+        sendEvent(bEvent);
+        refreshItem(focusItem);
```
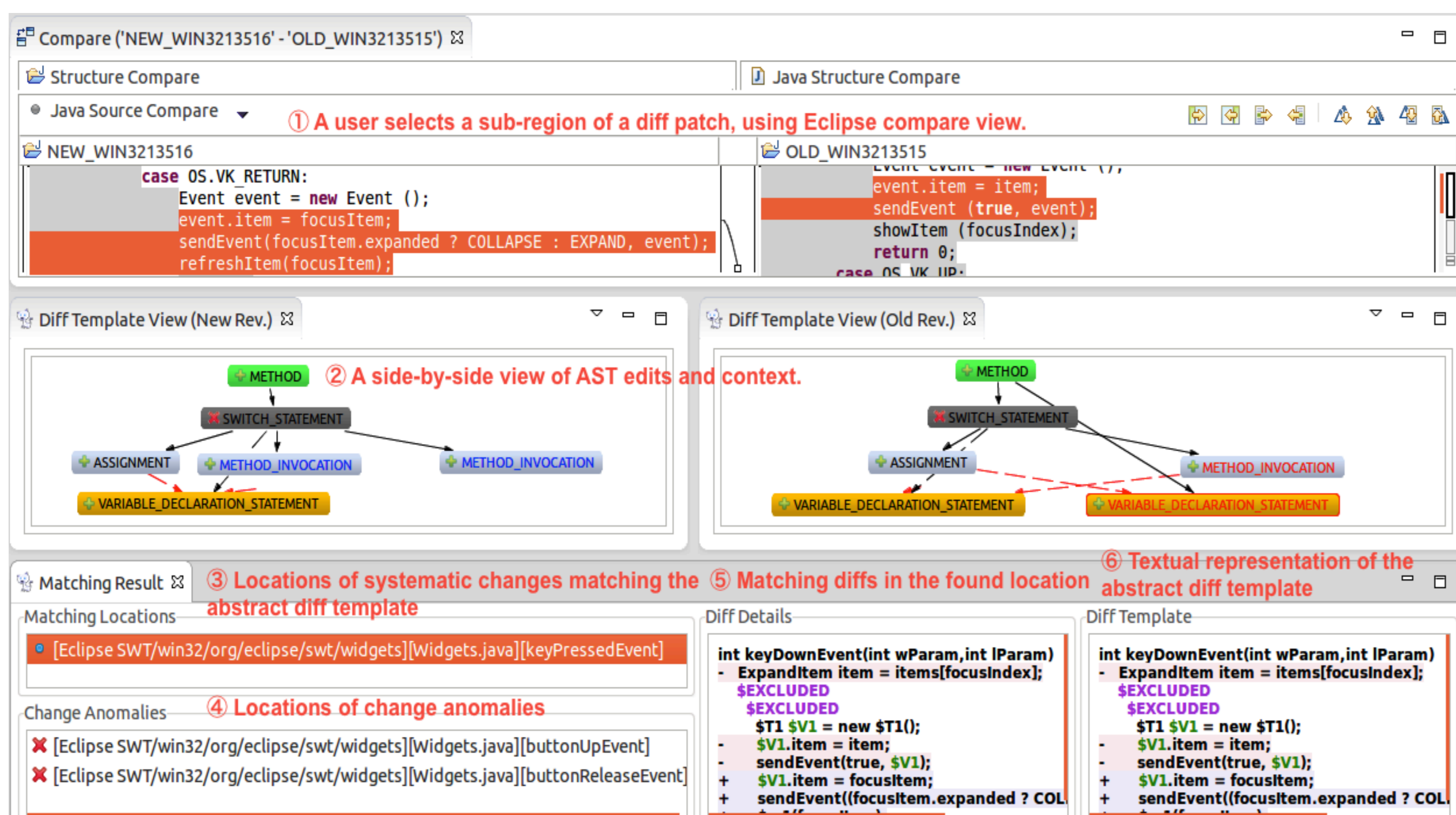
(a) change example   (b) a similar but not identical change   (c) an inconsistent change

Solution: Critics, An Interactive Code Review Tool

- *"What other code locations changed similar to this change?"*
- *"Are there any other locations that are similar to this code but are not updated?"*

- Combine **program differencing** with **pattern-based interactive code search**.
- Allow a reviewer to customize an abstract diff template by selecting a sub region of a *diff* patch and by generalizing the AST edits and surrounding context.
- Template matching and search using a Robust Tree Edit Distance algorithm

Critics Eclipse Plug-in Features

- A user can select code in an **Eclipse Compare View** and provide it as an input.
- A user can visualize and customize a template in a side-by-side **Diff Template View**.
- A user can examine similar changes and locates anomalies in the **Matching Result View**.
- A user can see a corresponding patch content using **Diff Details View**.
- A user can generalize type, method, and identifier names or exclude certain nodes for **Template Refinement**.