

Verbesserung und Erweiterung der Core-Bestandteile von Saros

Abschlussvortrag

Das Saros-Projekt

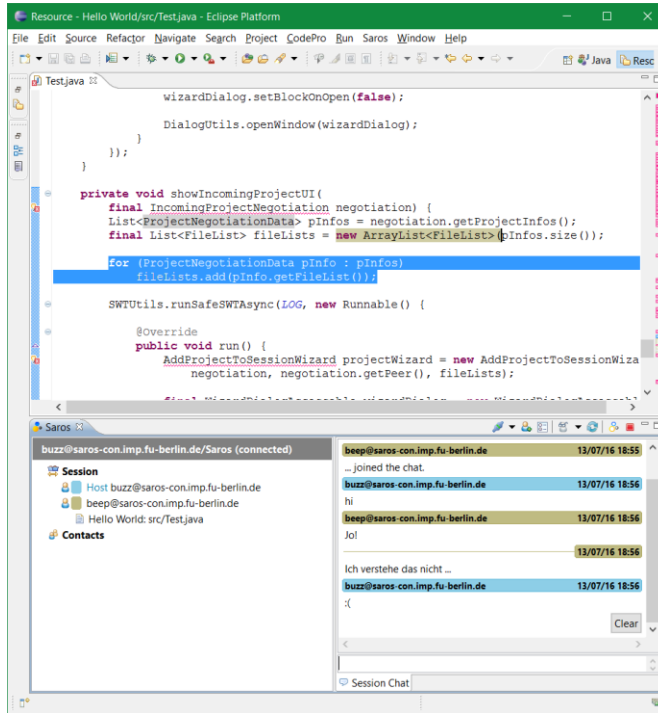


Saros

Distributed Party Programming

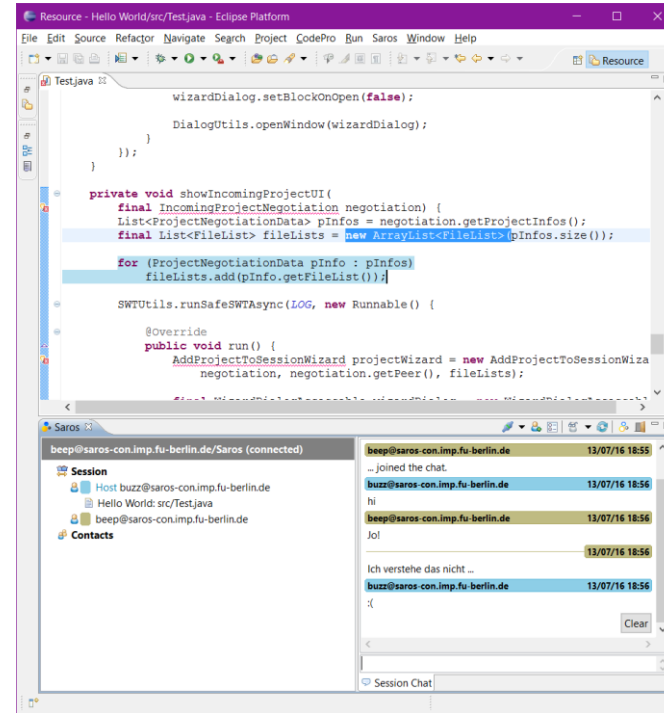
Saros-Logo, Quelle: <http://www.saros-project.org/>

- quelloffenes Projekt der AG Software Engineering
- verteilte Zusammenarbeit an Quelltext in Echtzeit
- Plug-In für Eclipse



The screenshot shows the Eclipse IDE with a Java file named 'Testjava'. The code includes a method 'showIncomingProjectUI' and a 'run' method. The chat window at the bottom shows a session with 'buzz@saros-con.imp.fu-berlin.de/Saros (connected)'. The chat history includes:

- ...joined the chat.
- hi
- Jol
- Ich verstehe das nicht ...
- :(



The screenshot shows the Eclipse IDE with the same Java file 'Testjava'. The chat window at the bottom shows a session with 'beep@saros-con.imp.fu-berlin.de/Saros (connected)'. The chat history includes:

- ...joined the chat.
- hi
- Hello World: src/Testjava
- Jol
- Ich verstehe das nicht ...
- :(

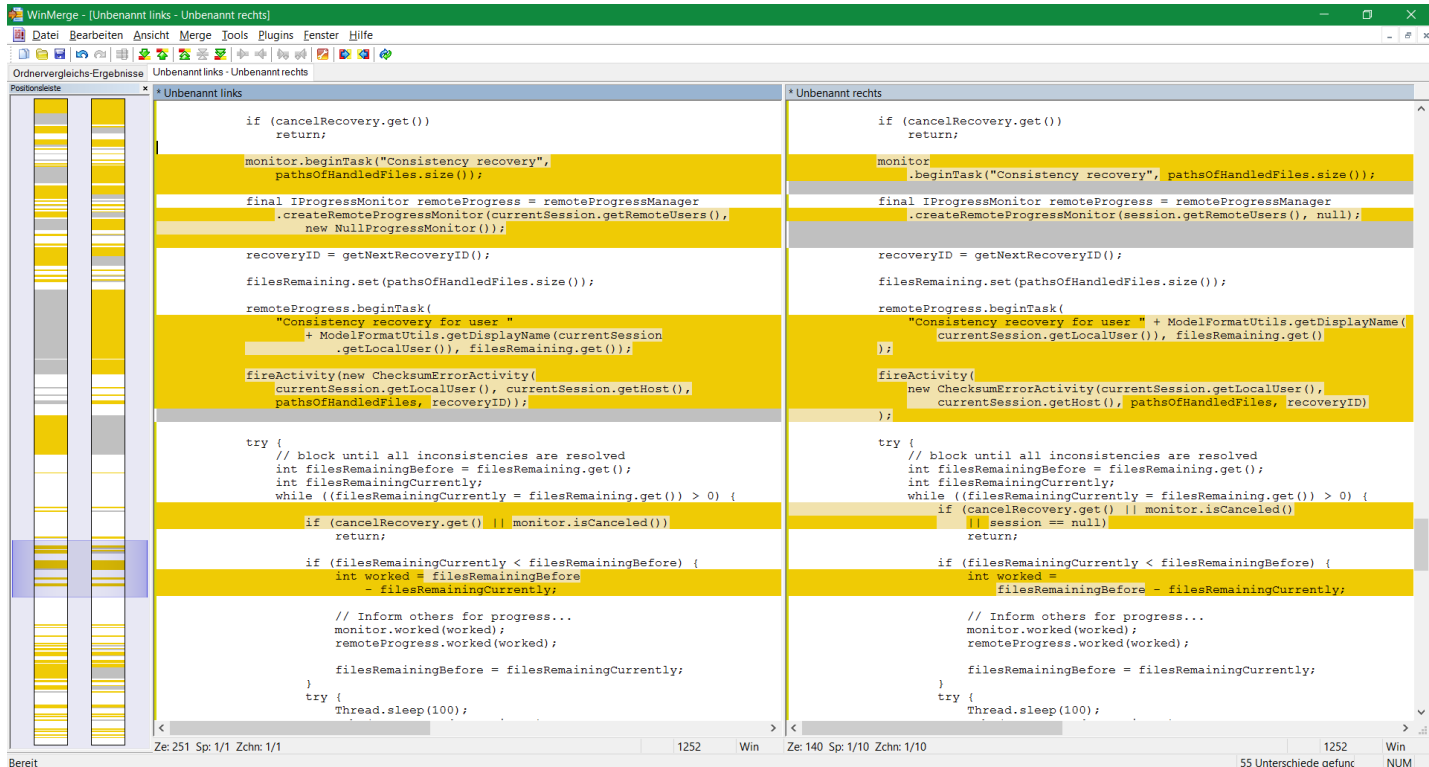
Saros für IntelliJ IDEA



Eclipse Logo, Quelle: <https://eclipse.org/artwork/>
IntelliJ IDEA Logo, Quelle: <https://www.jetbrains.com/company/press/>

- Plug-In für IntelliJ IDEA
- Core noch nicht umfangreich genug
- Codeduplikate für schnelle Implementierung angelegt

Codeduplikate



```

if (cancelRecovery.get())
    return;

monitor.beginTask("Consistency recovery",
    pathsOfHandledFiles.size());

final IProgressMonitor remoteProgress = remoteProgressManager
    .createRemoteProgressMonitor(currentSession.getRemoteUsers(),
        new NullProgressMonitor());

recoveryID = getNextRecoveryID();
filesRemaining.set(pathsOfHandledFiles.size());

remoteProgress.beginTask(
    "Consistency recovery for user "
    + ModelFormatUtils.getDisplayName(currentSession
        .getLocalUser()), filesRemaining.get());

fireActivity(new ChecksumErrorActivity(
    currentSession.getLocalUser(), currentSession.getHost(),
    pathsOfHandledFiles, recoveryID));

try {
    // block until all inconsistencies are resolved
    int filesRemainingBefore = filesRemaining.get();
    int filesRemainingCurrently;
    while ((filesRemainingCurrently = filesRemaining.get()) > 0) {

        if (cancelRecovery.get() || monitor.isCanceled())
            return;

        if (filesRemainingCurrently < filesRemainingBefore) {
            int worked = filesRemainingBefore
                - filesRemainingCurrently;

            // Inform others for progress...
            monitor.worked(worked);
            remoteProgress.worked(worked);

            filesRemainingBefore = filesRemainingCurrently;
        }
        try {
            Thread.sleep(100);
        }
    }
}
    
```

Positionenliste x Unbenannt links Unbenannt rechts

Ze: 251 Sp: 1/1 Zchn: 1/1 1252 Win

Ze: 140 Sp: 1/10 Zchn: 1/10 1252 Win

55 Unterschiede gefunc NUM

Anforderungen

- Entfernung von Codeduplikaten
- Abstraktion von geteilter Logik
- möglichst viel; bevorzugt komplexe Codeduplikate



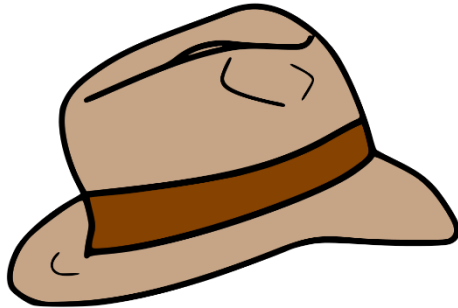
Sungaila16-CoreVerbesserung.pdf

Arbeitspakete

- „möglichst viel“ statt „fertige Implementierung“
- Arbeitspakete statt Meilensteine
- Entfernen eines komplexen Codeduplikates
→ Arbeitspaket

Die „zwei Hüte“-Metapher

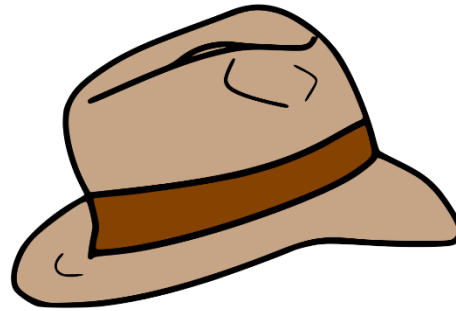
Refaktorisierung



neue Funktionen



Workflows of Refactoring, Martin Fowler,
Quelle: <http://martinfowler.com/articles/workflowsOfRefactoring/#2hats>



Verbesserung und Erweiterung der Core- Bestandteile von Saros


Arbeitspaket „RemoteEditorManager“

- ca. 2 Wochen
- 3 commits
- +176, -565



- verwaltet Editoren zwischen Teilnehmern
- **Problem:**
Eclipse-spezifische Datenstrukturen
- **Lösung:**
plattformunabhängige Datenstrukturen

Arbeitspaket „ConsistencyWatchdogs“

- ca. 4 Wochen
- 3 commits
- **+60**, **-1.214**

- hält Projektdateien konsistent
- **Problem:**
plattformspezifisches Speichern von Dateien
- **Lösung:**
Analyse ergibt, dass Speichern überflüssig

Arbeitspaket „Lifecycle“

- ca. 8 Wochen
- 4 commits
- +470, -517



- Initialisierung des Plug-Ins
- **Problem:**
individuelle
Implementierungen stark
ähnlicher Logik
- **Lösung:**
gemeinsame Logik in Core,
spezifische Logik in Plug-In

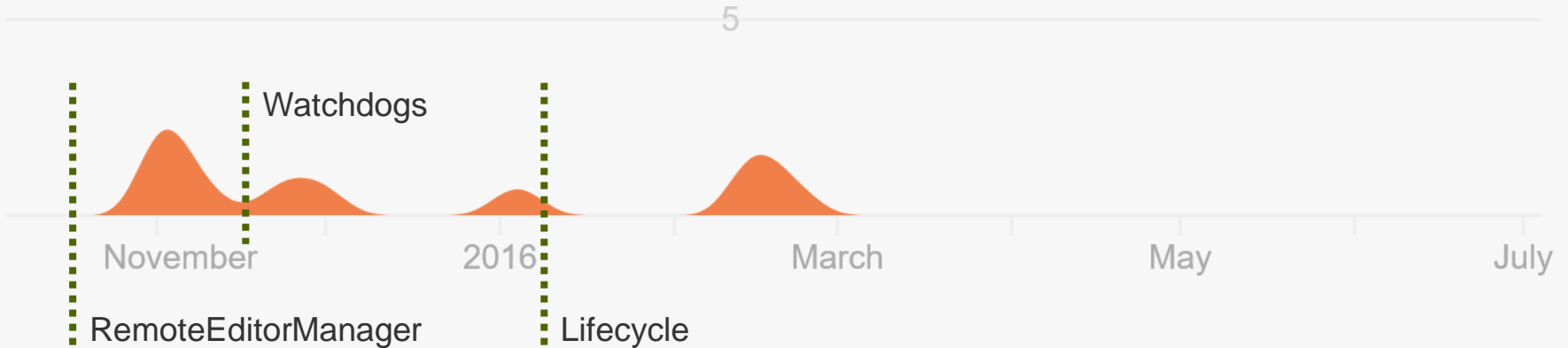
Zeitliche Verteilung



sungaila

#4

10 commits / 2,231 ++ / 3,821 --




saros-project/saros, github.com, Stand 10.07.2016



Verbesserung und Erweiterung der Core- Bestandteile von Saros

Erweiterung von HTML UI backend

- seit Ende Juni
 - 1 commit
 - +913, -29
- 
- Annahme von Sitzungseinladungen (noch) nicht möglich
 - Implementierung im Backend
 - Kooperation mit anderer Bachelorarbeit für Frontend

Ergebnisse

- komplexesten Codeduplikate entfernt
 - weitere Duplikate existieren; geringerer Aufwand
- Entwicklung an Plug-Ins erleichtert
- Erkenntnisse und Wissen dokumentiert
- Das HTML UI-Projekt unterstützt (laufend)

Vielen Dank!