

Einführung in die Softwaretechnik


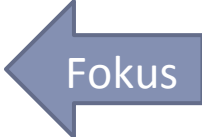
Konfigurationsmanagement

Klaus Ostermann
(mit Folien von Christian Kästner)

Agenda

- ▶ Verteiltes Arbeiten
- ▶ Versionskontrolle
 - ▶ Konzepte
 - ▶ CVS / SVN
 - ▶ Git
- ▶ Fehlerverwaltung mit Ticket-Systemen

Softwarekonfigurationsmanagement

- ▶ Übergreifende Disziplin
- ▶ Definitionen und Prozesse
- ▶ Versionierung und Konfliktbehandlung 
- ▶ Planung von Varianten
- ▶ Dokumentieren von Fehlern und deren Behebung 
- ▶ Änderungsmanagement
- ▶ Releaseplanung
- ▶ Automatisiertes Kompilieren/Ausliefern/Testen
- ▶ Zugriffskontrolle

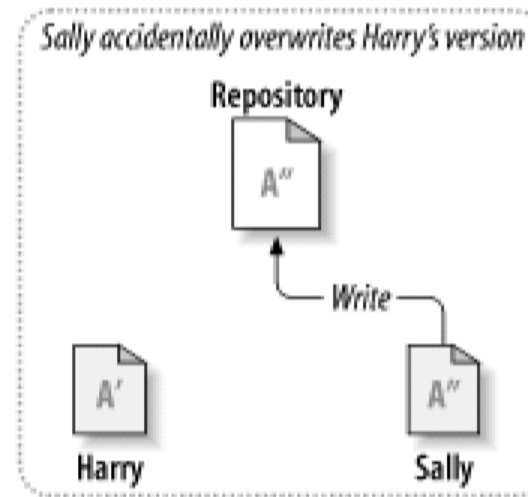
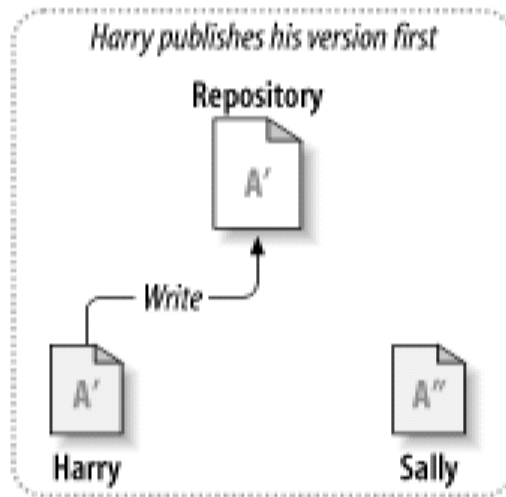
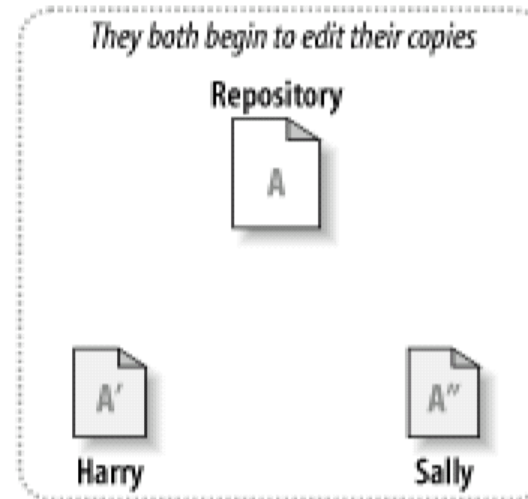
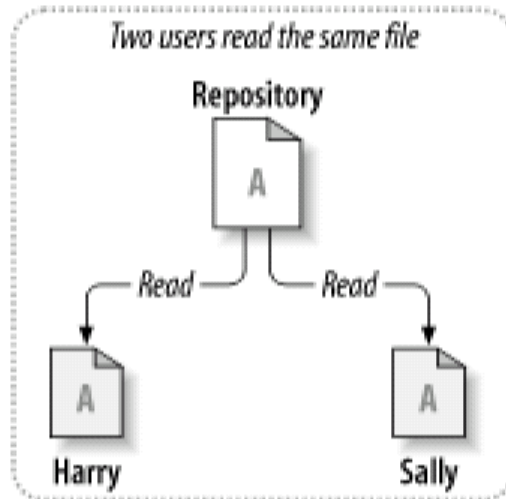
Kooperation auf gleicher Datei

Technische Zusammenarbeit

- ▶ **Wo liegen Dateien?**
 - ▶ Geänderte Dateien per Email zuschicken
 - ▶ Manuelles Synchronisieren bei Projekttreffen
 - ▶ Alle Dateien auf gemeinsamen Netzlaufwerk

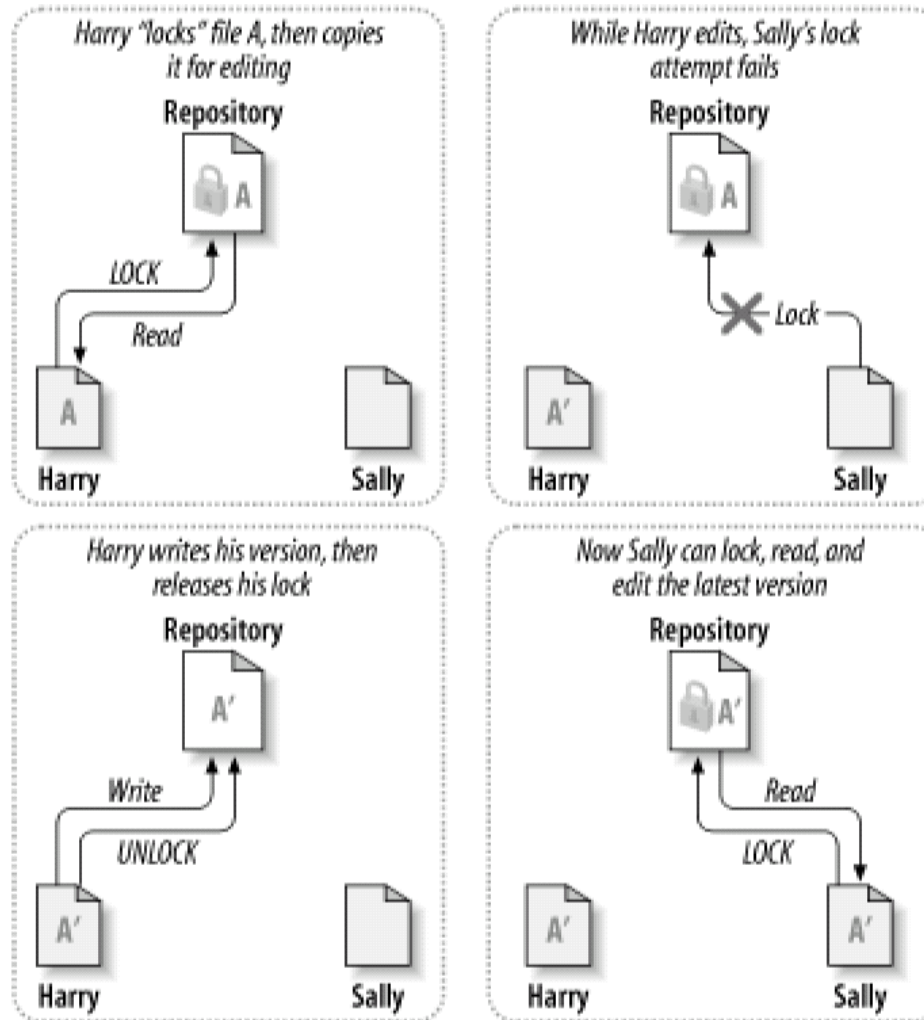
- ▶ **Wer darf was?**
 - ▶ Pro Datei ist ein Entwickler verantwortlich, nur er darf ändern
 - ▶ Jeder darf alles ändern

Änderungskonflikte



aus „Version Control with Subversion“

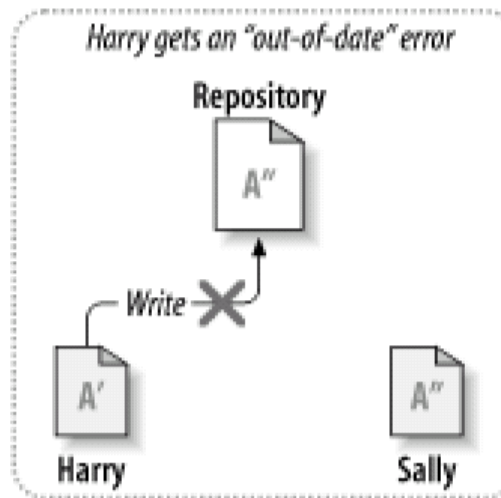
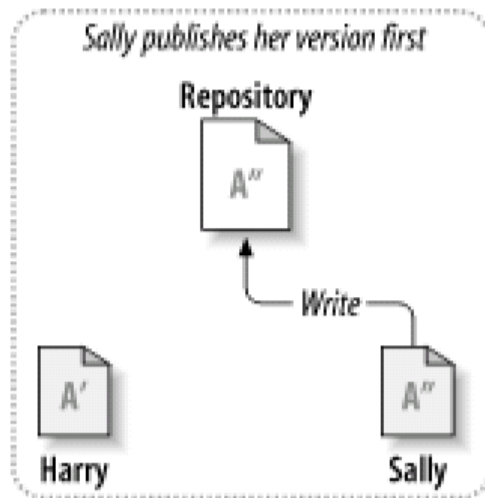
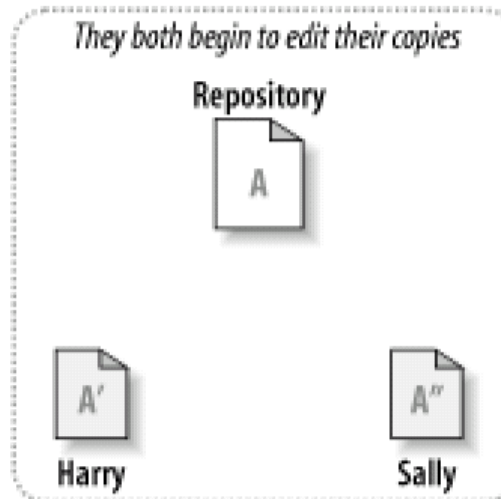
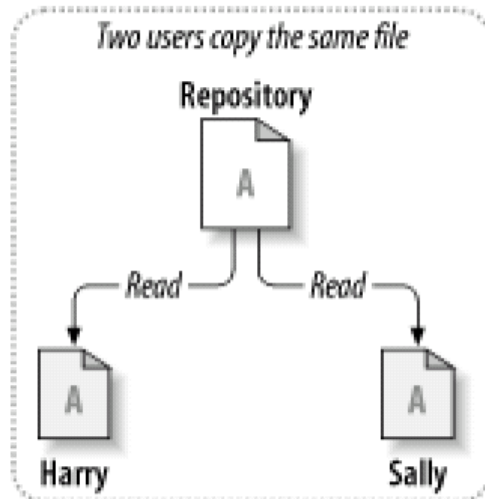
Konfliktvermeidung durch Sperren



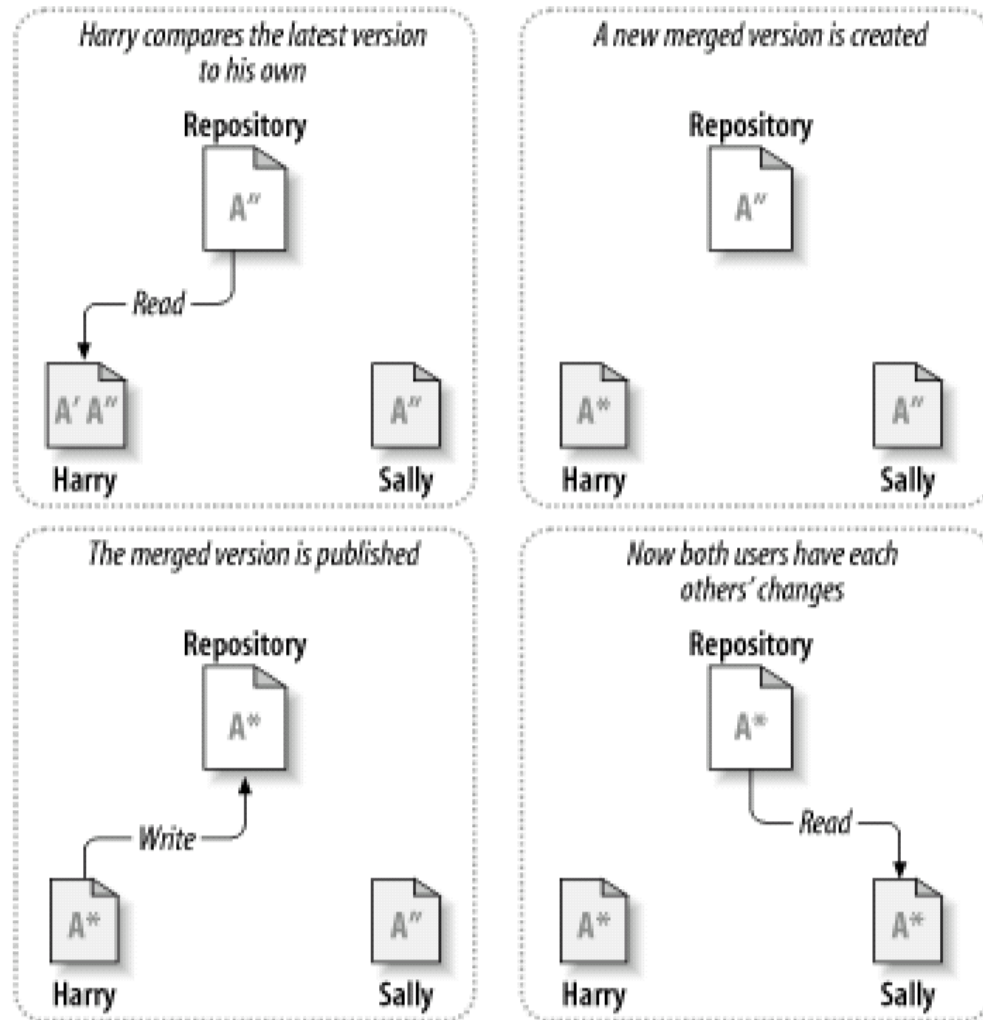
Probleme beim Sperren

- ▶ Technische Probleme:
 - ▶ Technische Sperren vs. Ankündigung auf Mailingliste
 - ▶ Vergessen zu entsperren typisch
- ▶ Unnötige Sequentialisierung der Arbeit:
 - ▶ Gleichzeitige Änderungen an unterschiedlichen Stellen nicht möglich
- ▶ Falsches Gefühl von Sicherheit:
 - ▶ Zwei Nutzer arbeiten getrennt auf den Dokumenten A und B. Was passiert, wenn A von B abhängig ist? A und B passen nicht mehr zusammen. Die Nutzer müssen dieses Problem diskutieren.

Konfliktlösung durch Mischen (Teil 1)



Konfliktlösung durch Mischen (Teil 2)



Beispiel

```
import java.util.LinkedList;
public class Stack<T> implements Cloneable {
    private LinkedList<T> items = new LinkedList<T>();
    public void push(T item) {
        items.addFirst(item);
    }
    public T pop() {
        if(items.size() > 0) return items.removeFirst();
        else return null;
    }
}
```

Beispiel

```
import java.util.LinkedList;
public class Stack<T> implements Cloneable {
    private LinkedList<T> items = new LinkedList<T>();
    public void push(T item) {
        items.addFirst(item);
    }
}
```

Edit 1

```
import java.util.LinkedList;
public class Stack<T>
    implements Cloneable {
    private LinkedList<T> items =
        new LinkedList<T>();
    public void push(T item) {
        items.addFirst(item);
    }
    public int size() {
        return items.size();
    }
    public T pop() {
        if(items.size() > 0) return
            items.removeFirst();
        else return null;
    }
}
```

Edit 2

```
import java.util.LinkedList;
public class Stack<T>
    implements Cloneable {
    private LinkedList<T> items =
        new LinkedList<T>();
    public void push(T item) {
        items.addFirst(item);
    }
    public T top() {
        return items.getFirst();
    }
    public T pop() {
        if(items.size() > 0) return
            items.removeFirst();
        else return null;
    }
}
```

Beispiel

```
import java.util.LinkedList;
public class Stack<T>
    implements Cloneable {
    private LinkedList<T> items =
        new LinkedList<T>();
    public void push(T item) {
        items.addFirst(item);
    }
    public int size() {
        return items.size();
    }
    public T pop() {
        if(items.size() > 0)
            return items.removeFirst();
        else return null;
    }
}
```



```
import java.util.LinkedList;
public class Stack<T> implements Cloneable {
    private LinkedList<T> items = new LinkedList<T>();
    public void push(T item) {
        items.addFirst(item);
    }
    <<<<<<< Top/Stack.java
    public T top() {
        return items.getFirst();
    }
    =====
    public int size() {
        return items.size();
    }
    >>>>>>> Size/Stack.java
    public T pop() {
        if(items.size() > 0) return items.removeFirst();
        else return null;
    }
}
```



```
import java.util.LinkedList;
public class Stack<T>
    implements Cloneable {
    private LinkedList<T> items =
        new LinkedList<T>();
    public void push(T item) {
        items.addFirst(item);
    }
    public T top() {
        return items.getFirst();
    }
    public T pop() {
        if(items.size() > 0)
            return items.removeFirst();
        else return null;
    }
}
```

System kann nicht automatisch die Reihenfolge entscheiden

Eigenschaften des Mischens

- ▶ Ein Dokument liegt in zwei Versionen vor.
 - ▶ Überlappende Änderungen: **Konflikt**
- ▶ Mischen nicht immer automatisierbar
 - ▶ Werkzeug (diff) zeigt Unterschiede an
 - ▶ Ein Nutzer integriert beide Versionen manuell (ggf. in Absprache)
- ▶ In der Praxis: die meisten Änderungen sind konfliktfrei

Verwalten von Projekten

Revisionen und Varianten

- ▶ Revisionen ersetzen frühere Revisionen (Zeitlich)
- ▶ Varianten koexistieren mit anderen Varianten (Inhaltlich)
- ▶ Version als Oberbegriff

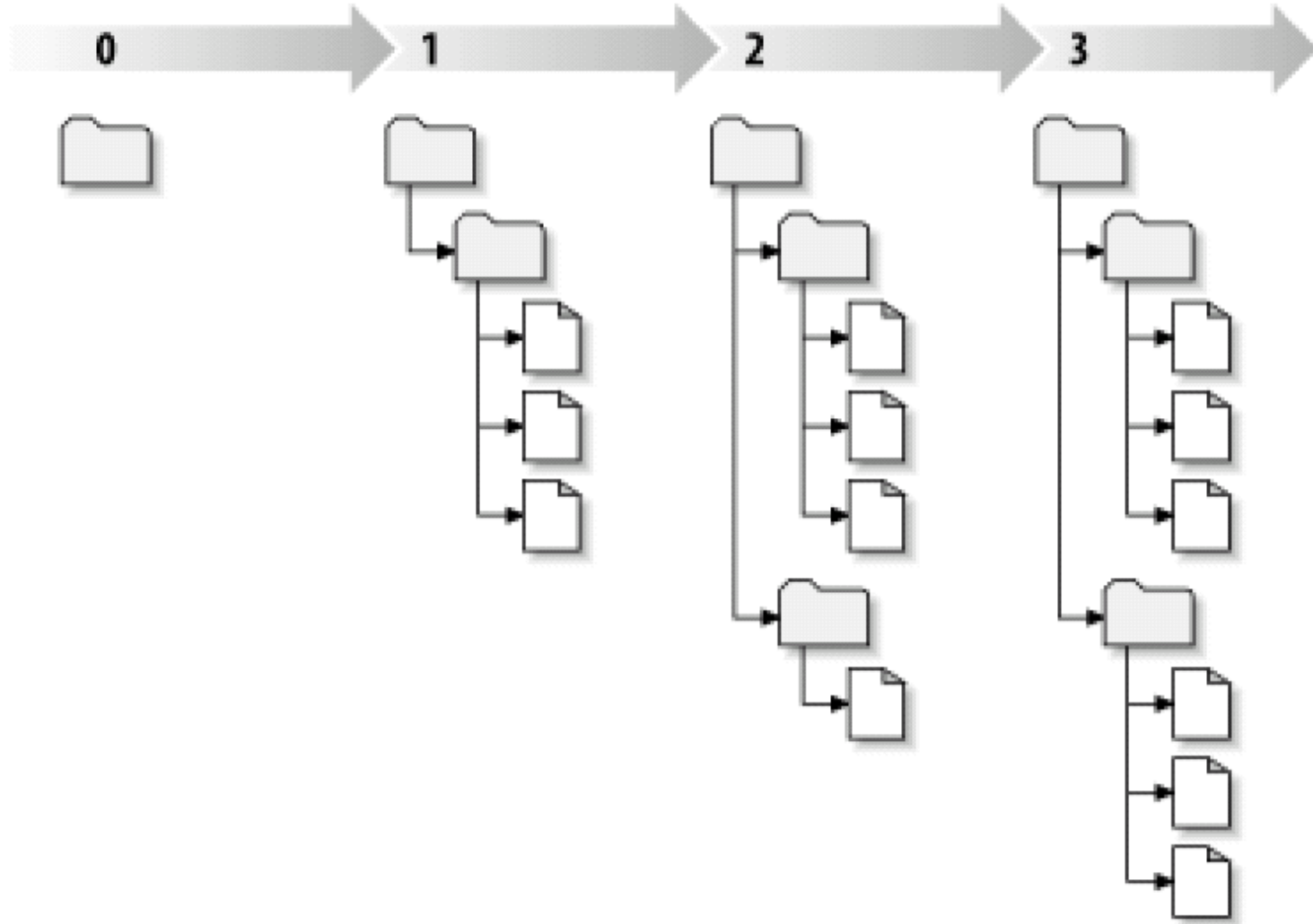
	V1.0	V1.1	V2.0	V3.0
Basissystem (Windows)	X	X	X	X
Linux-Variante		X	X	
Server-Variante			X	X
Erweiterung für Kunde A		X	X	X
Erweiterung für Kunde B				X



Code und Nicht-Code Dateien

- ▶ Java Code
 - ▶ Dokumentation
 - ▶ Modelle
 - ▶ Build-Scripte: Ant/Makefile
 - ▶ Lizenzen
 - ▶ Grammatiken
 - ▶ Kompilierte Dateien
 - ▶ HTML, JavaScript, CSS
-
- ▶ Bei Binärdateien ist Konfliktbehandlung schwieriger

Revisionen von Projekten

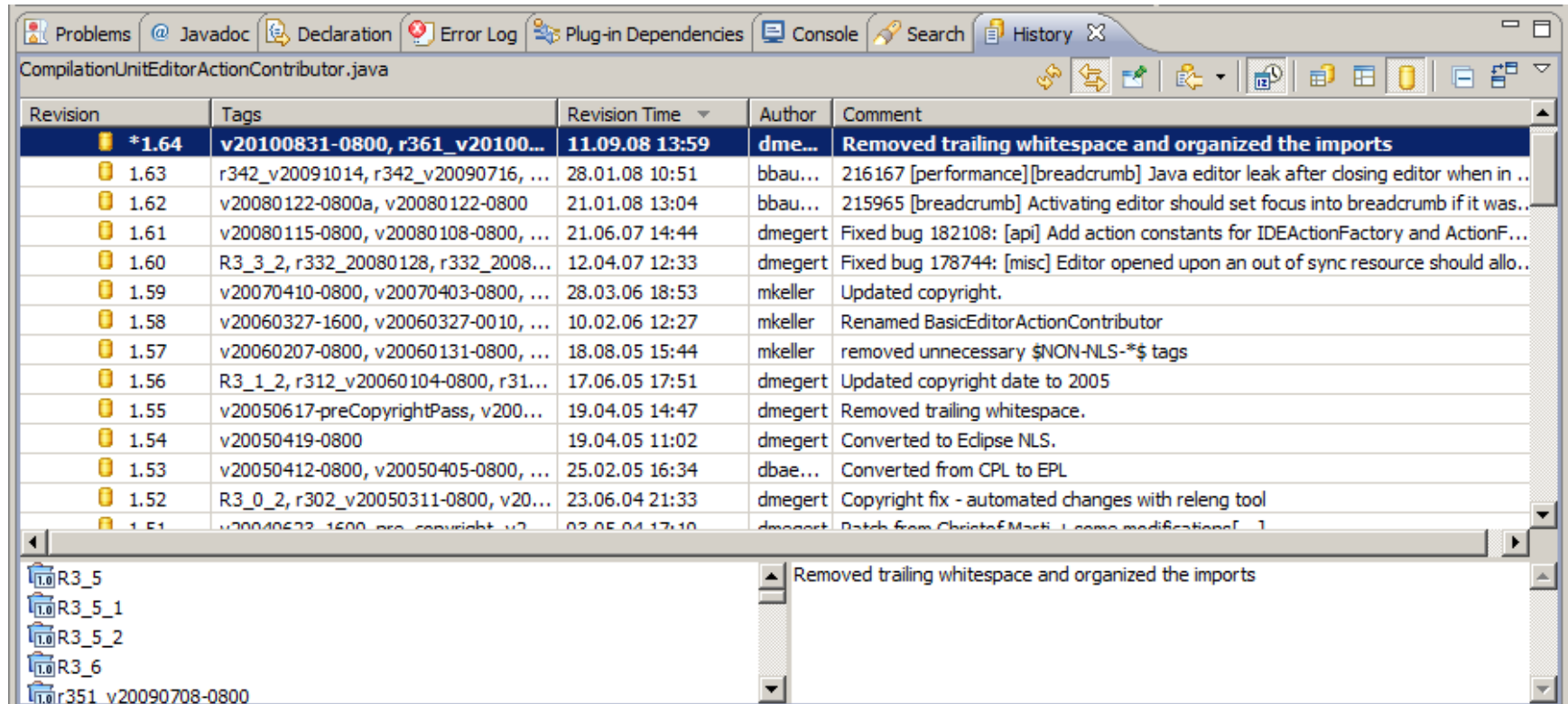


Versionsverwaltung

- ▶ Versionierung von Quelltextdateien
- ▶ Repository: Archiv alter Quelltextversionen
 - ▶ Zeitstempel und Benutzerkennung
 - ▶ Tags: Benannte Revisionen z.B. Release_1_0
 - ▶ Änderungen als Delta
 - ▶ Typisch: Kommentar beschreibt Änderung
- ▶ Jederzeit Änderungen nachvollziehen
- ▶ Alte Versionen wieder herstellbar



Revision History



Revision	Tags	Revision Time	Author	Comment
*1.64	v20100831-0800, r361_v20100...	11.09.08 13:59	dme...	Removed trailing whitespace and organized the imports
1.63	r342_v20091014, r342_v20090716, ...	28.01.08 10:51	bbau...	216167 [performance][breadcrumb] Java editor leak after closing editor when in ..
1.62	v20080122-0800a, v20080122-0800	21.01.08 13:04	bbau...	215965 [breadcrumb] Activating editor should set focus into breadcrumb if it was...
1.61	v20080115-0800, v20080108-0800, ...	21.06.07 14:44	dmegert	Fixed bug 182108: [api] Add action constants for IDEActionFactory and ActionF...
1.60	R3_3_2, r332_20080128, r332_2008...	12.04.07 12:33	dmegert	Fixed bug 178744: [misc] Editor opened upon an out of sync resource should allo...
1.59	v20070410-0800, v20070403-0800, ...	28.03.06 18:53	mkeller	Updated copyright.
1.58	v20060327-1600, v20060327-0010, ...	10.02.06 12:27	mkeller	Renamed BasicEditorActionContributor
1.57	v20060207-0800, v20060131-0800, ...	18.08.05 15:44	mkeller	removed unnecessary \$NON-NLS-*\$ tags
1.56	R3_1_2, r312_v20060104-0800, r31...	17.06.05 17:51	dmegert	Updated copyright date to 2005
1.55	v20050617-preCopyrightPass, v200...	19.04.05 14:47	dmegert	Removed trailing whitespace.
1.54	v20050419-0800	19.04.05 11:02	dmegert	Converted to Eclipse NLS.
1.53	v20050412-0800, v20050405-0800, ...	25.02.05 16:34	dbae...	Converted from CPL to EPL
1.52	R3_0_2, r302_v20050311-0800, v20...	23.06.04 21:33	dmegert	Copyright fix - automated changes with releng tool
1.51	v20040622-1600, pre-copyright, v2...	02.05.04 17:10	dmegert	Patch from Christof Marti: some modifications...

Aus Eclipse Quelltext: org.eclipse.jdt.ui

Release

- ▶ Release identifiziert veröffentlichte Produktversion

Java 1.6.0_15

Eclipse Platform SDT 3.5.2.M20100211-1343

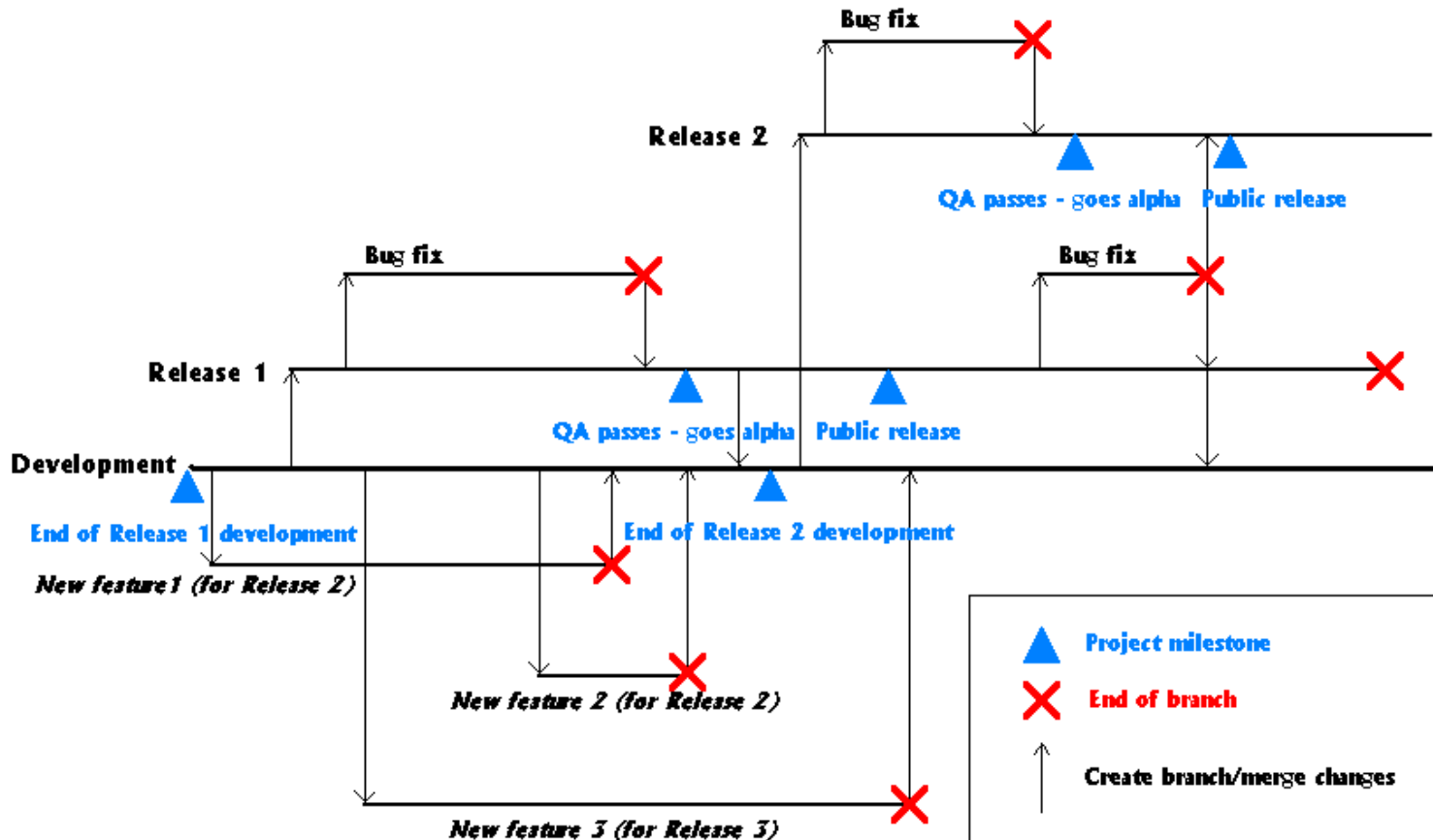
- ▶ Typisches Muster:
 - ▶ Hauptreleasenummer: Signifikante Änderungen
 - ▶ Nebenreleasenummer: Funktionserweiterungen
 - ▶ Revisionsnummer: Fehlerbehebungen
 - ▶ Buildnummer: weitere Details
- ▶ Release 0.x für Beta-Releases (vorab)
- ▶ Release-Versionen oft unabhängig von Revisionsnummern
- ▶ Tags markieren Releases

Branches (Verzweigen)

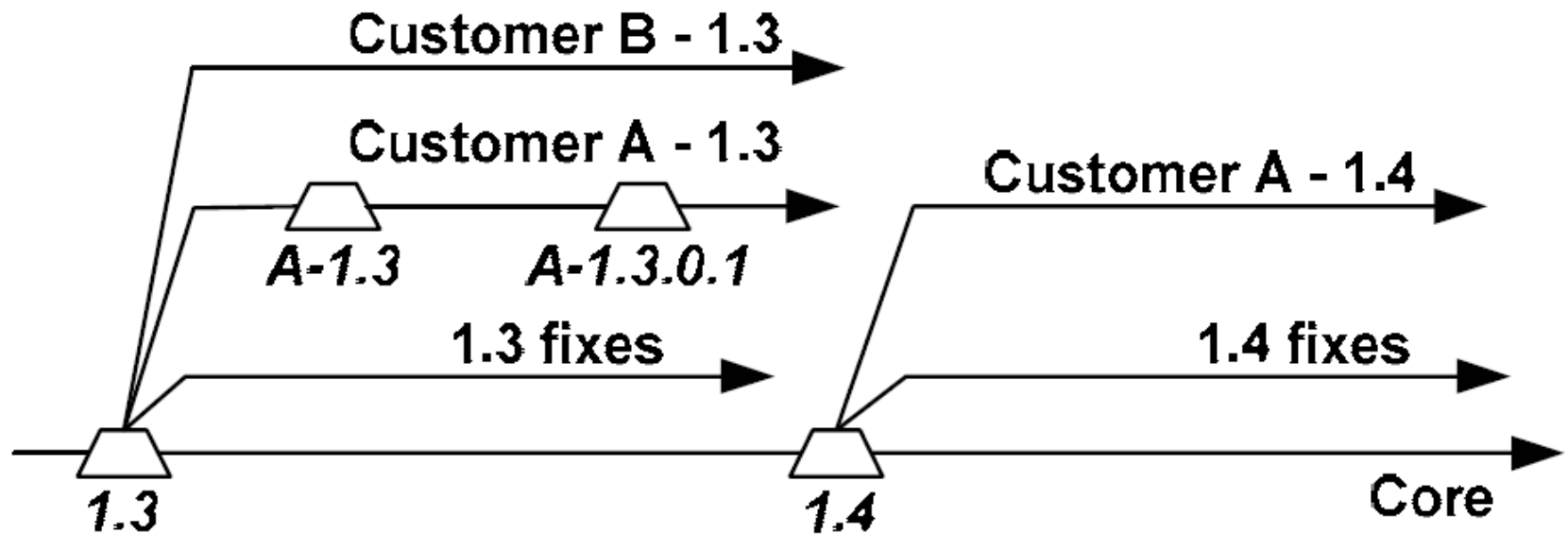
- ▶ Kopie des Quelltext
- ▶ Wird getrennt versioniert
- ▶ Kann wieder zusammengefügt werden (merge)

- ▶ Typisches Vorgehen
 - ▶ Hauptbranch für Wartung **oder** Hauptentwicklung
 - ▶ Neuer Branch für experimentelle Funktionalität, wird zusammengefügt wenn erfolgreich
 - ▶ Neuer Branch für Wartungsaufgaben
 - ▶ Teils neuer Branch für Varianten

Branches – Beispiel



Varianten und Revisionen



Variantenmanagement

- ▶ Variantenerstellung durch Branches skaliert nicht
- ▶ Spezielle Mechanismen oder Werkzeuge benötigt
- ▶ Komplexität muss geplant werden

- ▶ Viele Lösungen

- ▶ Konfigurationsdateien
- ▶ Präprozessor
- ▶ Build-System
- ▶ Spezielle Sprachen
- ▶ ...

- ▶ -> Softwareproduktlinien

```
/* common parts */  
...  
/* dependent on operating system */  
#if (OS == Unix)  
...  
#elif (OS == VMS)  
...  
#else  
...  
#endif  
...
```

Analog Abhängigkeiten von
Merkmalen oder Kunden

Versionsverwaltungssysteme

- ▶ Systeme für Sperren und Mischen verfügbar
- ▶ Lokale Versionsverwaltung
 - ▶ Lokale Archivierung (meist einzelner) Dateien
 - ▶ Beispielsysteme: SCCS und RCS
- ▶ Zentrale Versionsverwaltung
 - ▶ Revisionen liegen auf zentralem Server
 - ▶ Clients erfragen Updates, senden Änderungen
 - ▶ Beispielsysteme: CVS, SVN, Perforce, Visual SourceSafe
- ▶ Verteilte Systeme
 - ▶ Verteilte Repositories (mit allen bekannten Revisionen) die synchronisiert werden können
 - ▶ Beispielsysteme: Git, Mercurial, ClearCase

CVS / SVN

CVS und SVN

- ▶ Zentrale Versionsverwaltungssysteme
- ▶ CVS seit 1990, SVN als inoffizieller Nachfolger seit 2004
- ▶ Ein zentrales Repository
- ▶ Benutzer erstellen lokale Kopie
- ▶ Änderung der lokalen Kopie, Abgleich mit Repository
- ▶ Update – Commit Zyklen
- ▶ Unterstützt Branches und Tags
- ▶ Zentrale Rechteverwaltung

Typischer Arbeitszyklus

- ▶ Einmalig: Projekt lokal anlegen

- ▶ svn checkout

- ▶ Arbeitskopie auf den neuesten Stand bringen:

- ▶ svn update

- ▶ Änderungen an der Ordnerstruktur durchführen:

- ▶ svn add
- ▶ svn delete
- ▶ svn copy
- ▶ svn move

- ▶ Änderungen prüfen:

- ▶ svn status
- ▶ svn diff

- ▶ Änderungen zurücknehmen (optional):

- ▶ svn revert

- ▶ Konflikte auflösen:

- ▶ svn update
- ▶ svn resolved

- ▶ Änderungen in das Repository einlesen:

- ▶ svn commit

CVS vs. SVN

CVS

- ▶ Revisionsnummer pro Datei
- ▶ Textdateien (Binärdateien mögl.)
- ▶ Weiter verbreitet

SVN

- ▶ Revisionsnummer für ganzes Projekt
- ▶ Atomare Commits: alle Dateien oder keine
- ▶ Dateien und Verzeichnisse
- ▶ Umbenennungen
- ▶ Metadaten möglich und versioniert



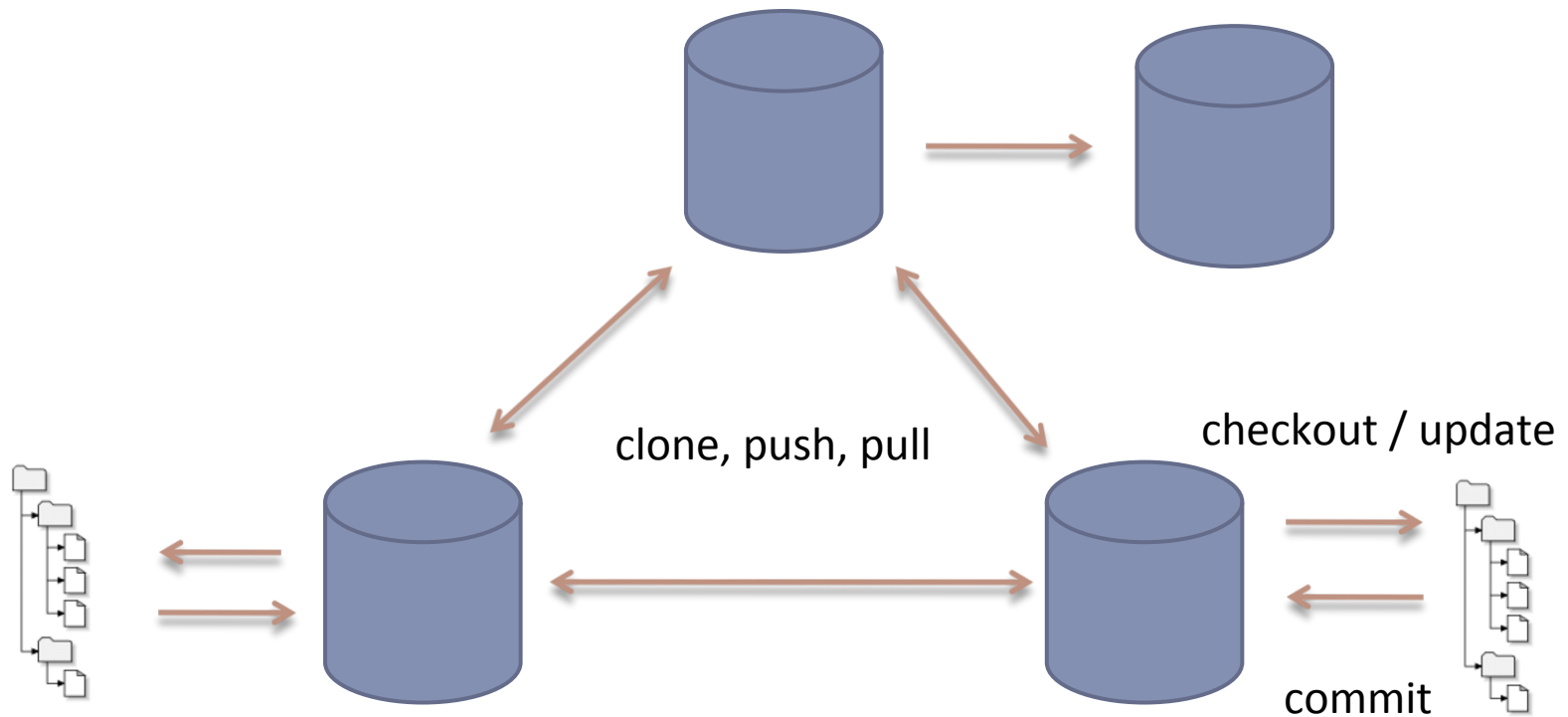
Git

Git

- ▶ Verteilte Versionsverwaltung
- ▶ Kein zentraler Server nötig
- ▶ Kopie des gesamten Repository lokal
 - ▶ Lokale Funktionalität ähnlich SVN (update, commit, branches, diff)
- ▶ Nicht-lineare Entwicklung: Spezieller Fokus auf verteiltes Verzweigen und Mischen
- ▶ Datenabgleich zwischen Repositories möglich

- ▶ Hohe Geschwindigkeit bei commit/diff/revert/..., da alle Daten lokal

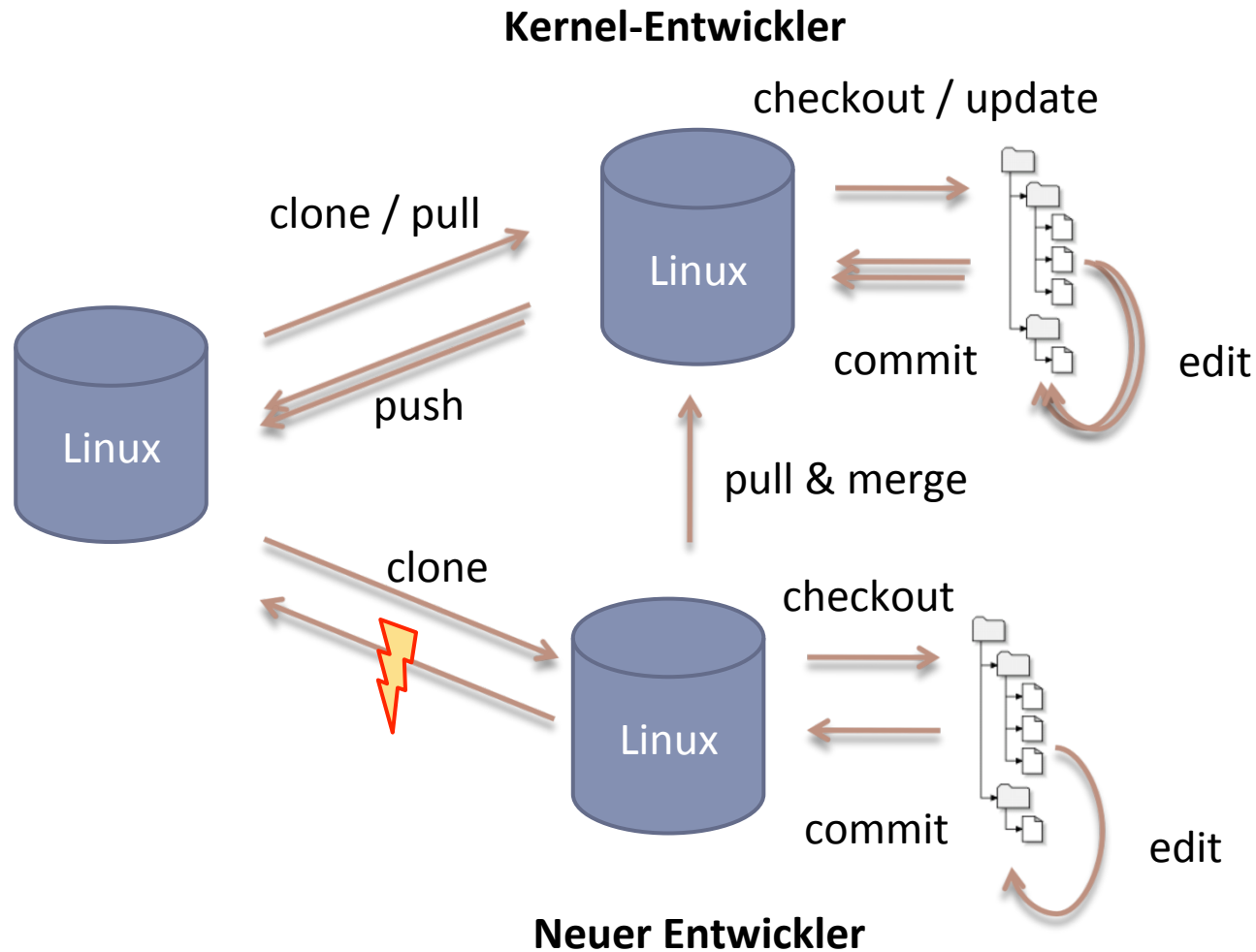
Übersicht



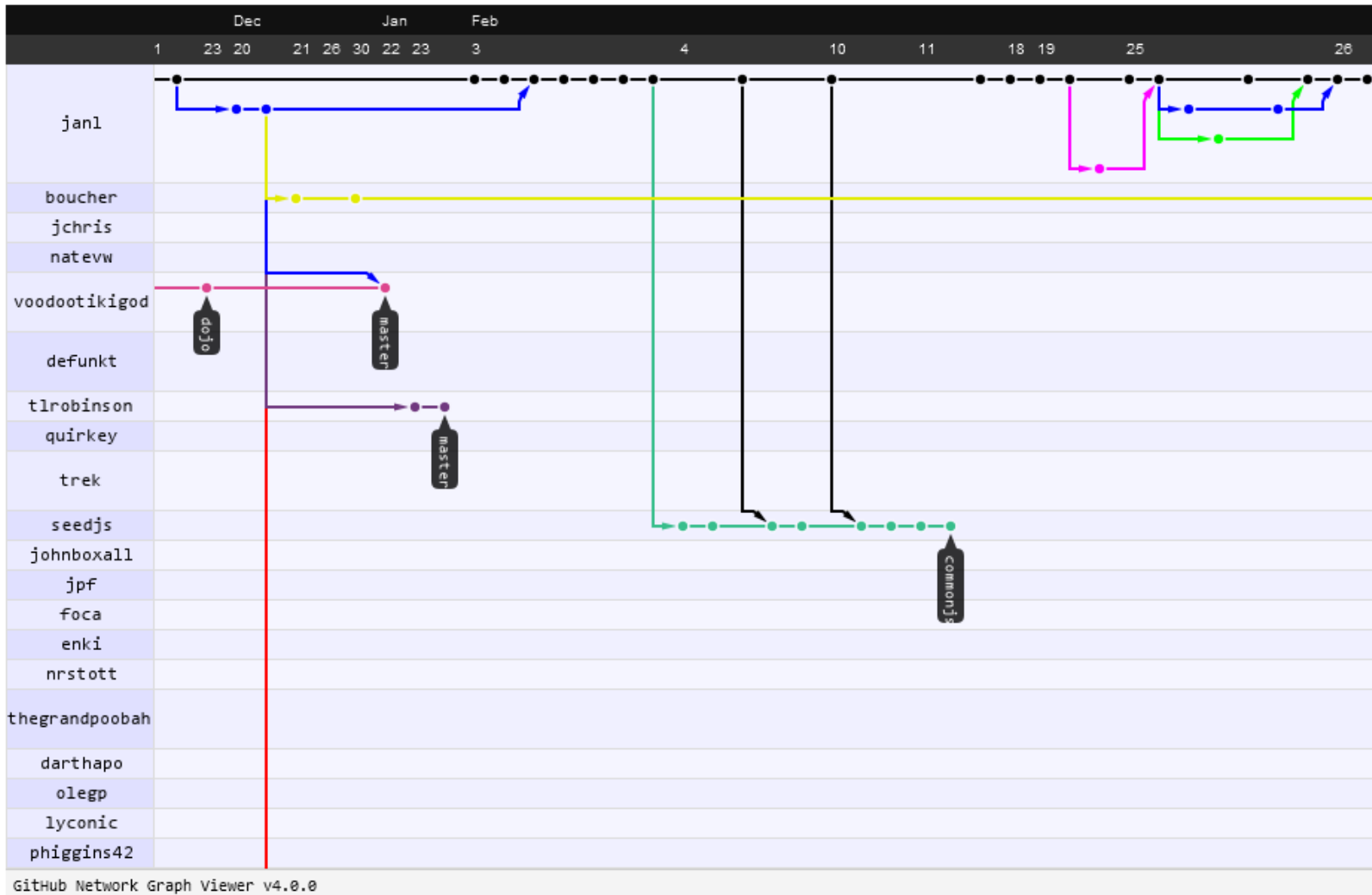
Verteilte Revisionen

- ▶ Revisionen nicht global koordiniert/sortiert
- ▶ Revisionen und Branches global eindeutig durch HashIDs
 - ▶ z.B. 52a0ff44aba8599f43a5d821c421af316cb7305
- ▶ Clonen eines Repositories jederzeit möglich
 - ▶ Merkt sich Ursprung (wichtig für Updates und Merging)
 - ▶ Normale Checkout/Commit Operationen
 - ▶ Commits möglich ohne ursprüngliches Repository zu ändern
- ▶ Fetch und Pull-Operationen holen Updates aus entfernten Repositories (auch mehreren!)
- ▶ Push-Operation kopiert lokale Änderung zu entferntem Repository (wenn Rechte vorhanden)

Beispiel



Repositories in mustache.js

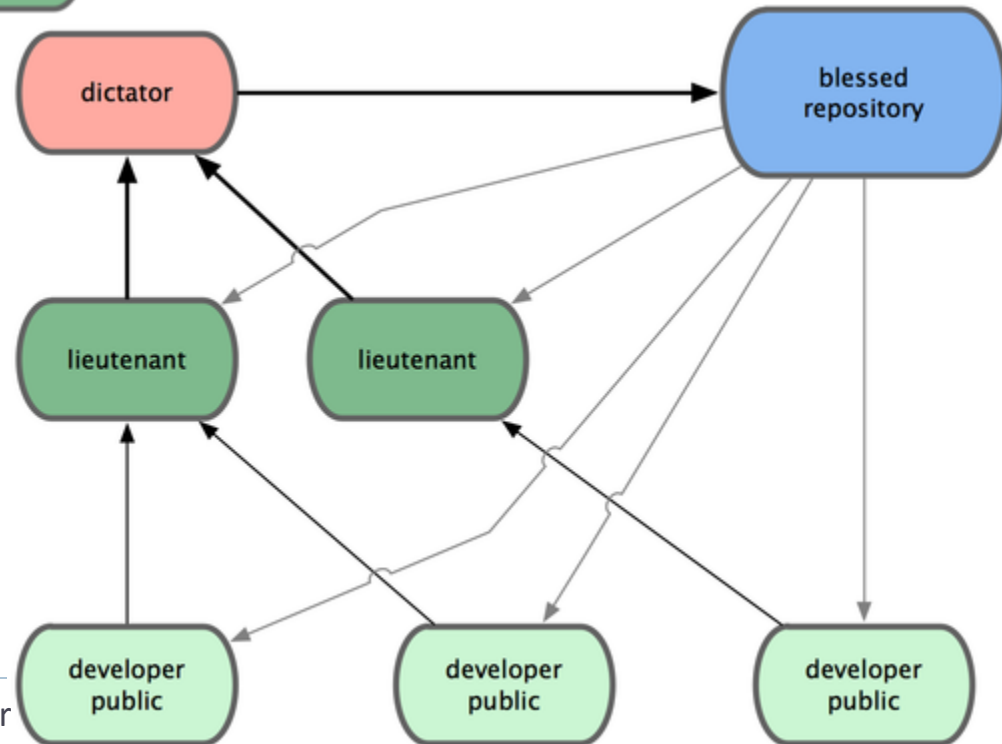
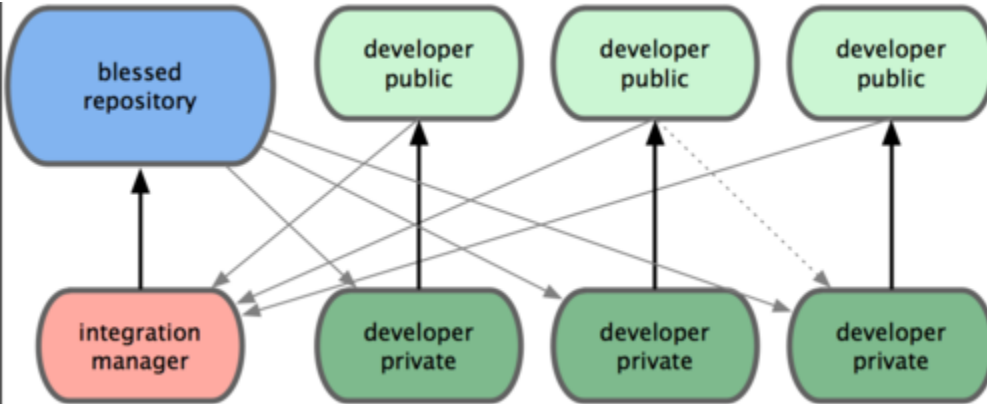


mustache.js Projekt

Beispiel Revisionshistorie

```
Terminal — less — 156x42
sh  bash  less
* 2009-12-08 575c2d8 bchesneau@gmail.com (bchesneau@gmail.com) Merge remote branch 'thomo/master' into mthomo
* 2009-12-07 a315955 Thomas.Mohaupt@gmail.com (Thomas.Mohaupt@gmail.com) Merge branch 'master' of git://github.com/couchapp/couchapp
* 2009-12-01 cbd2bc8 Thomas.Mohaupt@gmail.com (Thomas.Mohaupt@gmail.com) Bug: Trailing carriage returns are not stripped.
* 2009-11-28 c649863 Thomas.Mohaupt@gmail.com (Thomas.Mohaupt@gmail.com) Merge branch 'master' of git://github.com/couchapp/couchapp
* 2009-11-27 1418fa1 Thomas.Mohaupt@gmail.com (Thomas.Mohaupt@gmail.com) Fix wrong couchapp path
* 2009-11-27 1944cc0 Thomas.Mohaupt@gmail.com (Thomas.Mohaupt@gmail.com) Merge branch 'master' of git://github.com/couchapp/couchapp
* 2009-11-27 ae97905 Thomas.Mohaupt@gmail.com (Thomas.Mohaupt@gmail.com) Fix UTF-8 issue on WinXP
* 2009-11-26 97b5026 Thomas.Mohaupt@gmail.com (Thomas.Mohaupt@gmail.com) Merge branch 'master' of git://github.com/couchapp/couchapp
* 2009-11-19 00ed4d9 Thomas.Mohaupt@gmail.com (Thomas.Mohaupt@gmail.com) Fix WinXP problem: Attachment name is not UTF-8 encoded
* 2009-11-17 6a52137 mot@sq.s.int (mot@sq.s.int) Fix: USERPROFILE handling
* 2009-12-08 34828d0 bchesneau@gmail.com (bchesneau@gmail.com) fix hooks and compress hook
* 2009-12-08 60dec82 bchesneau@gmail.com (bchesneau@gmail.com) fix compress options search path
* 2009-12-07 3c89e38 jasondavies@apache.org (jasondavies@apache.org) Generate attachment signatures before processing macros.
* 2009-12-07 54c3e83 jasondavies@apache.org (jasondavies@apache.org) Merge branch 'master' of git://github.com/couchapp/couchapp
* 2009-12-04 9b97679 bchesneau@gmail.com (bchesneau@gmail.com) fix __iter__
* 2009-12-04 4b8c9e7 bchesneau@gmail.com (bchesneau@gmail.com) fix typo. thanks to markh.
* 2009-12-04 b37230e bchesneau@gmail.com (bchesneau@gmail.com) patch from @rmg. thanks!
* 2009-11-29 233238a bchesneau@gmail.com (bchesneau@gmail.com) bump version number to 0.5.1.
* 2009-11-29 d8c9709 bchesneau@gmail.com (bchesneau@gmail.com) add update template to generators
* 2009-11-28 0964b63 bchesneau@gmail.com (bchesneau@gmail.com) couchapp standalone for macosx via py2app
* 2009-11-28 4def0a6 bchesneau@gmail.com (bchesneau@gmail.com) make sure ocontent type is defined
* 2009-11-28 10b53e3 benoitc@.(none) (benoitc@.(none)) new fixes while testing
* 2009-11-28 051e778 bchesneau@gmail.com (bchesneau@gmail.com) rethink hook system. final version. Now like extensions each hooktype is a list of
* 2009-11-28 c47718f bchesneau@gmail.com (bchesneau@gmail.com) rethink extensions. So now extensions are a list of key=value pair. where key is
* 2009-11-27 53a2040 bchesneau@gmail.com (bchesneau@gmail.com) fix compress extension & load extensions like we load hooks
* 2009-11-27 343862d bchesneau@gmail.com (bchesneau@gmail.com) fix template paths with windows
* 2009-11-27 bfa2f8c bchesneau@gmail.com (bchesneau@gmail.com) backport import_module from python 2.7
* 2009-11-27 3380a4c benoitc@.(none) (benoitc@.(none)) more windows fix
* 2009-11-27 a049b41 bchesneau@gmail.com (bchesneau@gmail.com) more fixes
```

Zentrale Repositories weiterhin möglich



© Scott Chacon Buch "Pro Git"

“Social Coding” mit Github u.ae.

The screenshot shows a web browser window titled "jan's Profile - GitHub - Opera". The address bar shows "http://github.com/janl". The profile page for Jan Lehnardt is displayed, featuring a profile picture, name, and contact information. The page is divided into several sections: a profile summary, public repositories, and public activity.

Profile Summary:

- Name: Jan Lehnardt
- Email: jan@apache.org
- Website/Blog: <http://jan.prima.de/plok/>
- Location: Berlin, Germany
- Member Since: May 23, 2008

Public Repositories (92):

- mustache.js** (JavaScript, 619 stars, 65 forks): Minimal templating with `{{mustaches}}` in JavaScript. Last updated 2 days ago. 52 week participation.
- talks** (3 stars, 1 fork): my talks. Last updated 3 days ago. 52 week participation.
- konamibacon.js** (JavaScript, 2 stars, 1 fork): what it says on the tin. Last updated 5 days ago.

Public Activity:

- janl pushed to gh-pages at oreilly/couchdb-guide** about 13 hours ago. 5e3505e fix listing, thanks to damjan.
- janl started following daleharvey** 1 day ago. daleharvey has 12 public repos and 29 followers.
- janl closed issue 294 on oreilly/couchdb-guide** 2 days ago. Confusion in tour.html.
- janl pushed to gh-pages at oreilly/couchdb-guide** 2 days ago. d209771 Clarify note, fix markup. Closes #294.
- janl pushed to master at janl/mustache.js** 2 days ago. e4a7966 More FAQurate. c6524cc spelling.

Ticket-Systeme

Wie verwaltet man Fehlermeldungen?

- ▶ Sofort bearbeiten
- ▶ Email-/Text-/Zettelsammlung
- ▶ Kommentare im Quelltext
- ▶ Wiki

- ▶ Typische Fragen:
 - ▶ Organisation?
 - ▶ Wer ist verantwortlich?
 - ▶ Hat sich jemand darum gekümmert?
 - ▶ Ist es noch aktuell?
 - ▶ Welche Module sind besonders fehleranfällig?

Ticket-Systeme

- ▶ Speichern Fehlermeldung in zentraler Datenbank
- ▶ Meta-Informationen
- ▶ In Open-Source-Projekten typischerweise öffentlich

- ▶ Fehler können Gruppen/Listen/Personen zugeordnet werden
- ▶ Prioritäten setzen
- ▶ Fortschritt wird protokolliert
- ▶ Alle Änderungen nachvollziehbar

Bugzilla für Eclipse

200 bugs found.

ID	Sev	Pri	OS	Assignee	Status	Resolution	Summary
28556	nor	P3	Windows XP	daniel_megert	RESO	INVA	Creating a java project favors the Java perspective above a current Java Browsing
33005	nor	P3	Windows XP	jdt-text-inbox	RESO	DUPL	Java Editor: shows problem annotations of java files in non-java projects
216042	nor	P3	Windows XP	jdt-ui-inbox	RESO	WORK	change "Java, Java Browsing, Java Type Hierarchy" perspectives links
63225	maj	P3	Windows 2000	jdt-ui-inbox	RESO	DUPL	Mark occurrences breaks Java editor on Java file when file not part of Java project
37554	nor	P3	Linux-Motif	jdt-ui-inbox	CLOS	INVA	Eclipse2.1 Linux Motif: Double click on a .java file in Java Package Explorer will not open the default Java editor (OK on Windows and on Linux GTK)
67648	nor	P3	Windows XP	jdt-ui-inbox	RESO	INVA	"Java Project" Wizard is not grouped under the "Java"
116935	nor	P3	All	jdt-ui-inbox	RESO	INVA	New Class wizard: Java Model Exception: Java Model Status [Invalid name specified: Examble.java]
120563	min	P3	Windows XP	jerome_lanneluc	VERI	FIXE	Javadoc has many references to .java where all java source files are meant
75894	nor	P3	Windows 2000	jdt-text-inbox	ASSI		[implementation] 'Internal Error' if delete all Java Classes from Java Project
14194	nor	P3	Linux-Motif	philippe_mulet	RESO	WORK	Java source files shouldn't show errors when in src dir, but not java resource
36940	enh	P4	All	jdt-ui-inbox	RESO	DUPL	[plan item] Support Java references outside Java code
89266	nor	P3	Windows 2000	jdt-text-inbox	RESO	DUPL	Java Editor UI is broken with non-java extension support (bug 71460)
254766	nor	P3	Windows XP	daniel_megert	VERI	FIXE	Eclipse doc - Java incl./excl. pattern page unclear re Java source vs. resource files
33229	nor	P3	Windows XP	jdt-text-inbox	ASSI		[misc] Edit->Show Tooltip Desc. is enabled for Java file from CVS / non-Java project
72582	nor	P3	Windows XP	jdt-text-inbox	RESO	FIXE	Periodically I get "Java AST creation" internal error when the Java editor gets focus
134411	nor	P3	Windows XP	jdt-core-inbox	RESO	WORK	Error opening .java files in non-java projects.
21055	nor	P3	Windows NT	Darin_Wright	RESO	WONT	java/lang/NoClassDefFoundError: java/lang/Object on Run Local Java Application with jdk118
60246	nor	P3	Windows XP	martin_aeschlimann	RESO	FIXE	Wizards] Java Model Exception when create Java Elements (Package/Source Folder) with same name, different case
133609	nor	P3	Windows XP	tobias_widmer	RESO	FIXE	Java Model Exception: A.java does not exist when Extract Supertype
41099	nor	P3	All	jdt-text-inbox	RESO	FIXE	[misc] Opening what seems to be a non-java file with the Java Editor causes

Beispiel

3. Refactor - Rename project 'foo' to 'foob'
==> Dialog pops up. Stack trace above is in error log.
==> Breakpoint is removed and subsequent breakpoint refactorings seem to be disabled.

Scott Kellicker 2009-11-17 18:08:45 EST [Comment 1](#)

[Created an attachment \(id=152440\) \[details\]](#)
Zip file of the project to be imported.

Project to import in step 1.

Scott Kellicker 2009-11-17 18:09:42 EST [Comment 2](#)

seems different than but maybe related to 186719 and 162987

Olivier Thomann 2009-11-17 21:46:04 EST [Comment 3](#)

Move to JDT/Debug

Scott Kellicker 2009-11-18 08:57:27 EST [Comment 4](#)

[Zip file](#)
zip)
2009-11-
[Add an](#)

Note
You ne

1. root is a PackageFragmentRoot ("src/main/java [in foo] ...")
2. rootResource is incorrectly set via this call:
rootResource = project.getFolder(root.getElementName());
to Folder "F/foob/java". Seems it should be "F/foob/src/main/java"
4. destRoot is set based on rootResource and used later in the refactoring.

Hopefully this additional info will help someone with more knowledge of this code.

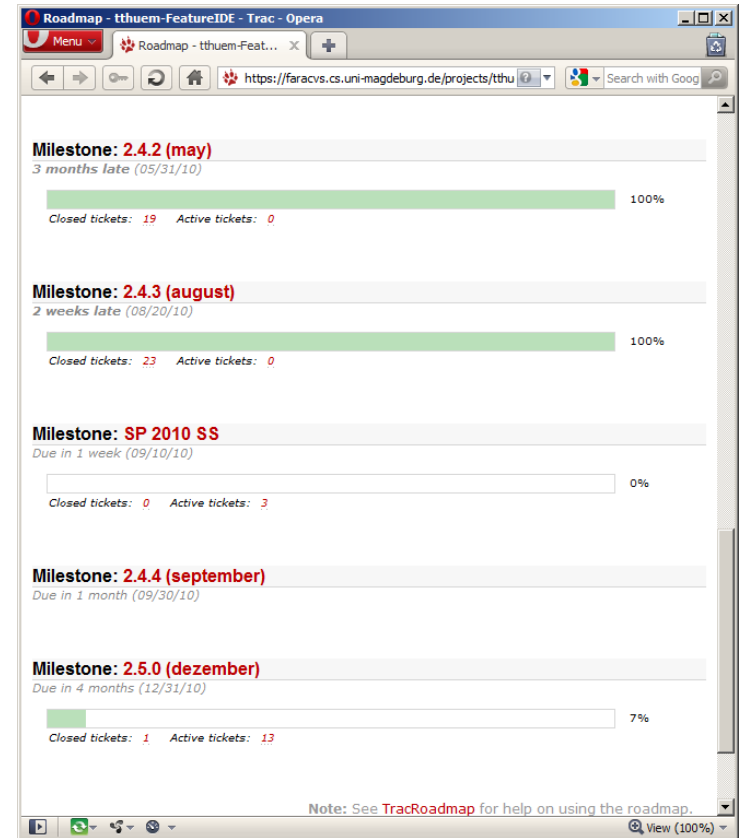
View (100%)

Vorgehen (grob)

- ▶ Fehlermeldung kommt an
- ▶ Aufnehmen als Ticket (Status: new, Priorität setzen)
- ▶ Prüfen ob der Fehler wirklich auftritt (Status: confirmed)
- ▶ Projektmanager weist Ticket passendem Entwickler zu (Status: assigned)
- ▶ Entwickler stellt ggf. Rückfragen oder gibt das Ticket weiter
- ▶ Entwickler entfernt Fehler und schließt das Ticket (Status: closed, Resolution: Fixed / Duplicate / Invalid / Won't Fix)
- ▶ Ggf. release neue Version, Information an Kunden

Nicht nur Fehlerverwaltung (Issue Tracking)

- ▶ Offene Aufgaben
- ▶ Ideen
- ▶ Kundenwünsche
- ▶ Help-Desk-Anrufe
- ▶ Automatische Ticketgenerierung von Alarmsystemen
- ▶ Jeweils mit Möglichkeit zur
 - ▶ Diskussion
 - ▶ Priorisieren
 - ▶ Protokollierung von Zuständigkeiten und Fortschritt

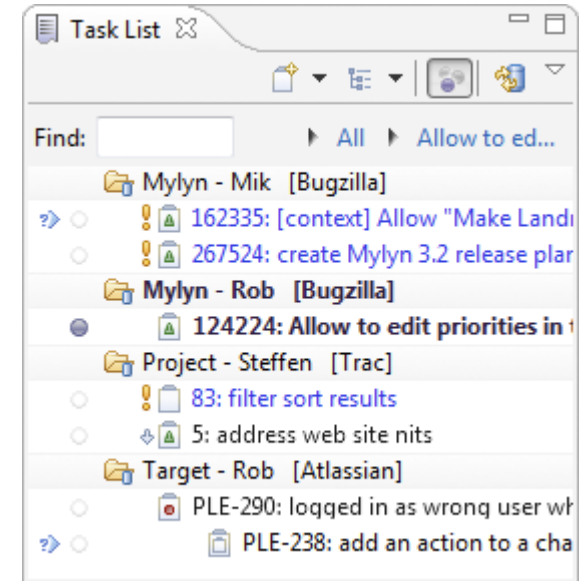
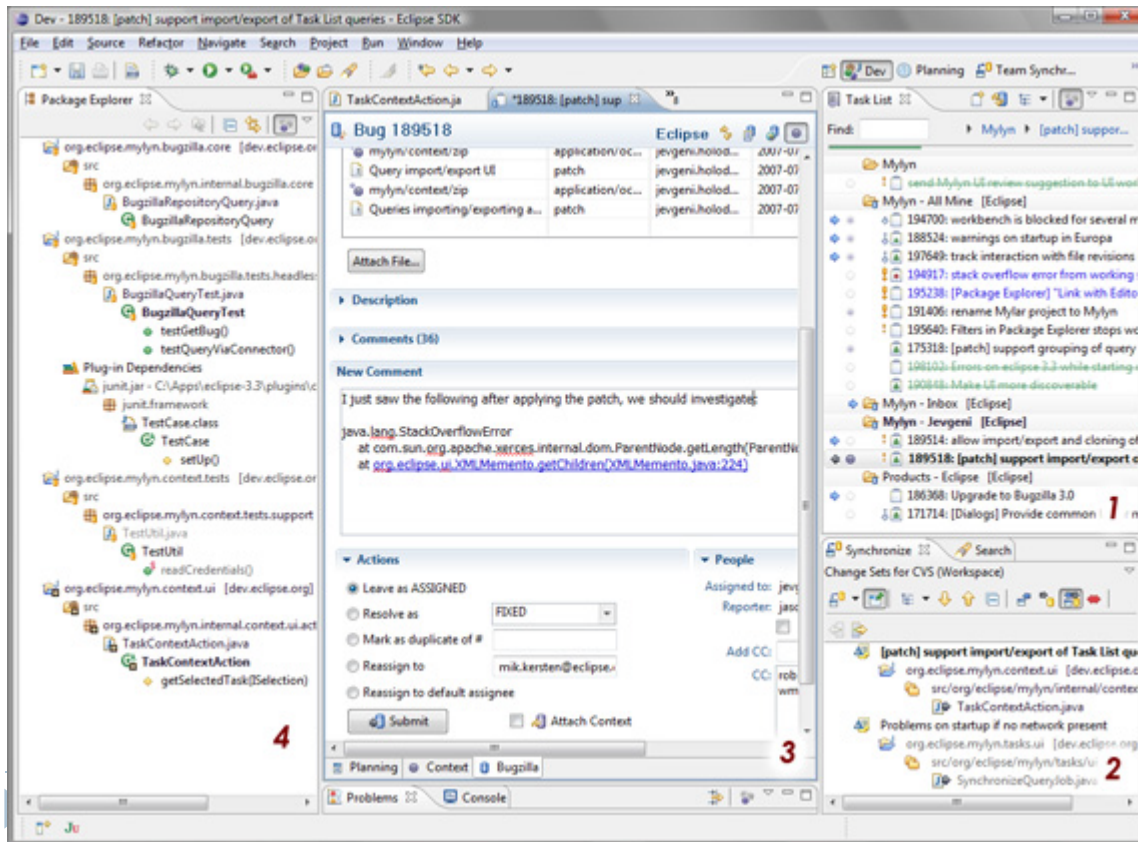


Ticket-Systeme als Kontrollwerkzeug

- ▶ Erzwing vordefinierte Vorgehensprozesse (Workflows)
- ▶ Alle Änderungen werden protokolliert
 - ▶ z.B. wer hat wann die Priorität geändert
- ▶ Erlaubt diverse Statistiken
- ▶ Bearbeitungsdauer und –Qualität
- ▶ Welche Module sind besonders fehleranfällig
 - ▶ ggf. hilfreich bei Ursachenforschung
- ▶ Sammeln von häufigen Fragen (FAQs)
- ▶ Nachvollziehbarkeit für Kunden

IDE Integration: Eclipse Mylyn

- ▶ Zeigt Tickets direkt in Eclipse an
- ▶ Integration mit Versionsverwaltung
- ▶ Automatische Kontextverwaltung



Software

Server?

- ▶ Viele kostenlose Angebote für (Open-Source) Projekte
 - ▶ sourceforge.net
 - ▶ assembla.com
 - ▶ github.com
 - ▶ code.google.com
 - ▶ ...
- ▶ Web-Oberfläche
- ▶ Oft mit Bug-Tracker
- ▶ Aufsetzen eines eigenen Servers gut dokumentiert

siehe auch http://en.wikipedia.org/wiki/Comparison_of_open_source_software_hosting_facilities

Clients?

- ▶ Kommandozeilenwerkzeuge
- ▶ Graphische Benutzeroberflächen, z.B.
 - ▶ TortoiseCVS/SVN/Git
 - ▶ gitk, giggle
 - ▶ rapidsvn
- ▶ Integration in IDEs, z.B.
 - ▶ Native CVS Unterstützung in Eclipse
 - ▶ Subversion Plugin für SVN
 - ▶ EGit Plugin für Git
- ▶ Webfrontends für Lesezugriff

Ticket-Systeme

- ▶ Bugzilla
- ▶ Trac
- ▶ SAP CRM

- ▶ Lassen sich mit Versionsverwaltungssystemen kombinieren
- ▶ In SourceForce, Assembla, Github, etc. mit angeboten

Zusammenfassung

- ▶ Revisionen und Varianten
- ▶ Verzweigung und Behandeln von Konflikten
- ▶ Verteilte Versionsverwaltung

- ▶ Fehlerverwaltung mit Ticket-Systemen