

Zentrale und dezentrale Versionsverwaltungswerkzeuge in Open Source Projekten

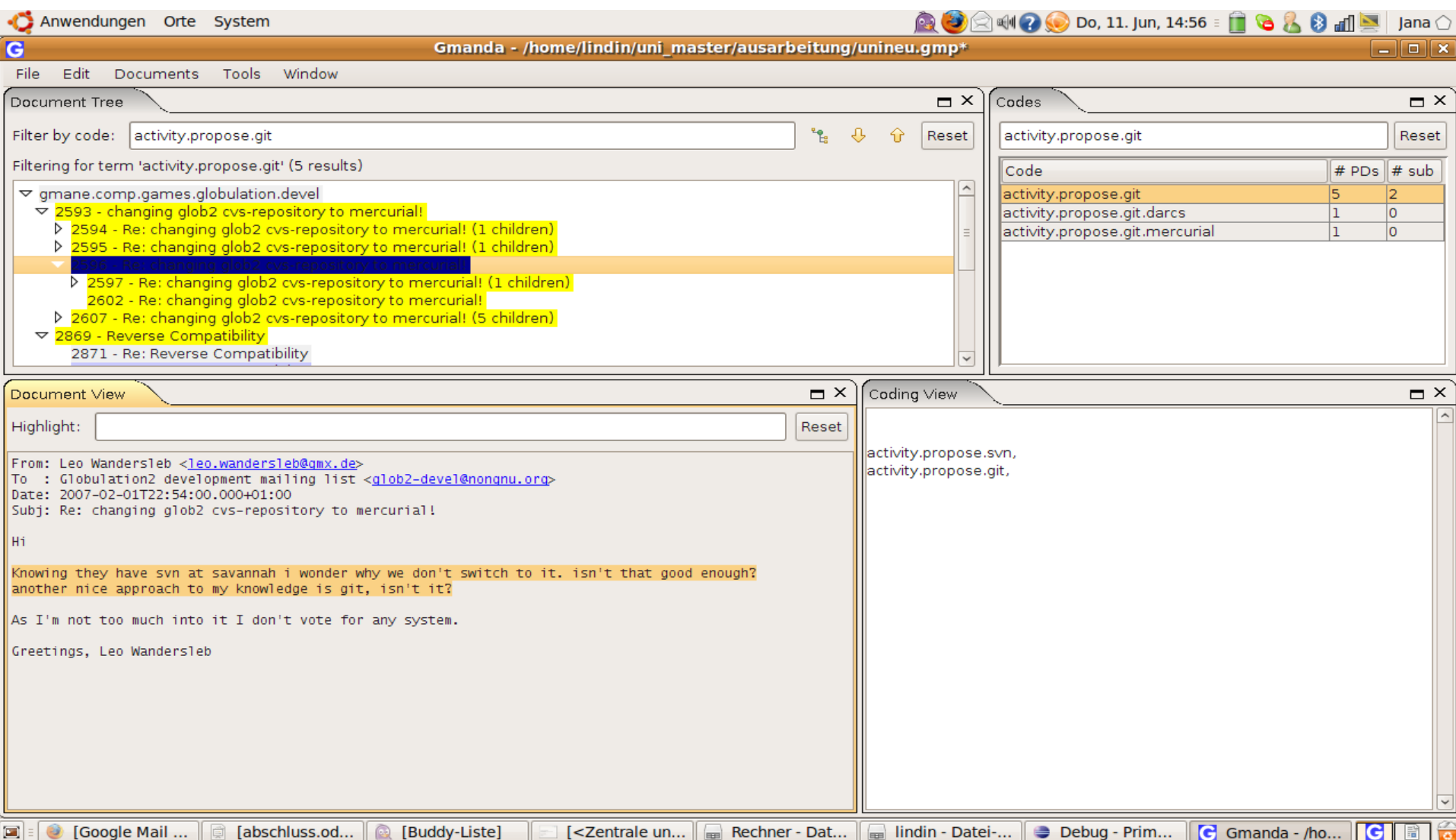
Jana von dem Berge
Institut für Informatik
FU Berlin
12.06.09

- Einbettung in die Arbeitsgruppe
 - Christopher Oezbek - Dissertation
 - „Einführung von Innovationen in Open Source Projekten“
 - 2007 auffallend viele Migrationen der SCM
- Untersucht werden sollten
 - Unterschiede in den Arbeitsprozessen
 - Einführungsprozesse
 - Lernprozesse
 - Adoption
 - Einfluss von Machtverhältnissen
 - ...

Grounded Theory

- Gegenstandsverankerte Methode
- Schrittweise Theoriegenerierung
- Bietet 3 aufeinander aufbauende Vorgehensweisen
 - Offenes Kodieren
 - Axiales Kodieren
 - Selektives Kodieren
 - Werden abwechselnd angewendet
- Weitere Datenerhebung erfolgt auf Grundlage bereits entstandener Theorieansätze

- Gmane Data Analyzer
- Unterstützt die Analyse von Textdokumenten nach den Arbeitsweisen der Grounded Theory
- Ursprünglich Mailinglisten von gmane.org
- Kodierung einzelner Emails
- Filterung der Emails nach
 - Suchbegriffen
 - Codes
- Visualisierung
 - Zeitliches Auftreten bestimmter Codes



The screenshot shows the GmanDA application interface. At the top, there is a system tray with various icons and the date 'Do, 11. Jun, 14:56'. The main window title is 'Gmanda - /home/lindin/uni_master/ausarbeitung/unineu.gmp*'. The interface is divided into several panes:

- Document Tree:** Shows a tree view of files and folders. The filter is set to 'activity.propose.git'. The tree shows a folder 'gmane.comp.games.globulation.devel' containing several sub-items, including '2593 - changing glob2 cvs-repository to mercurial!', '2594 - Re: changing glob2 cvs-repository to mercurial! (1 children)', '2595 - Re: changing glob2 cvs-repository to mercurial! (1 children)', '2597 - Re: changing glob2 cvs-repository to mercurial! (1 children)', '2602 - Re: changing glob2 cvs-repository to mercurial!', '2607 - Re: changing glob2 cvs-repository to mercurial! (5 children)', '2869 - Reverse Compatibility', and '2871 - Re: Reverse Compatibility'.
- Codes:** A table showing the results of the search for 'activity.propose.git'.

Code	# PDs	# sub
activity.propose.git	5	2
activity.propose.git.darcs	1	0
activity.propose.git.mercurial	1	0
- Document View:** Shows the content of the selected document. The text is an email from Leo Wandersleb to the Globulation2 development mailing list. The subject is 'Re: changing glob2 cvs-repository to mercurial!'. The body text includes: 'Hi', 'Knowing they have svn at savannah i wonder why we don't switch to it. isn't that good enough? another nice approach to my knowledge is git, isn't it?', 'As I'm not too much into it I don't vote for any system.', and 'Greetings, Leo Wandersleb'.
- Coding View:** Shows the code snippets extracted from the document, including 'activity.propose.svn,' and 'activity.propose.git,'.



ROX Desktop



Globulation



Flyspray



GNU Grub

Das U-Boot



- Vorkodierte Daten
 - Verständnis für das Kodieren gewinnen
 - Filtern von bereits als wichtig erkannten Emails
- Suche und Kodierung von geeigneten neuen Emaillisten
- Filterung über Stichwortsuche in den Emails
 - Ständig erweitert
 - z.B.: SVN, Subversion, repository, merge...
- Memo-Vergabe bei interessant erscheinenden Emails
 - z.B. „Der Maintainer weist auf Probleme hin, die eine Migration des SCMs mit sich führen kann“
- Bestimmung von zentralen Phänomenen
- In Beziehung setzen von Emails, die das gleiche Phänomen behandeln

Vorgehen - Probleme

- Kodierung
 - Versuche, Christophers Schema zu übernehmen
 - Konnte nicht funktionieren
 - Schema auf Innovationen bezogen
 - Kein formalisiertes Schema
 - Schwierigkeiten eigenes Schema zu entwickeln → Memos
- Bestimmung von Beziehungen zwischen den Emails sehr mühsam
- Projekte waren sehr unterschiedlicher Natur
 - Größe
 - Rollenverteilung
 - → allgemein gültige Theorie schwer zu finden

- Migration
- Repository-Hosting
- Betriebsmodi
- Adaptersysteme
- Living on the Bleeding Edge

- 3 Phasen: Vorschlag, Diskussion, Einführung
- Gründe für einen Vorschlag
 - Zunehmende Verbreitung von dezentralen Systemen
 - Häufig Erfahrung mit verschiedenen SCMs aus anderen Projekten
 - Probleme mit verwendetem SCM
 - Branching (z.B. in Globulation)
 - ...
- Wichtige Faktoren
 - Wer ist der Innovator?
 - Maintainer
 - Aktives Mitglied
 - Sonstige
 - Migrationsvorschlag inkl. Lösungsvorschlag Ja/Nein?

Migration – Vorschlag - Beispiel

- ROX
- Innovator = Maintainer
- Email sehr strukturiert und durchdacht
 - Beschreibung der Probleme des existierenden SCMs
 - Vorteile für Maintainer
 - Vorteile für Mitglieder
 - Lösung: Git
- Praktisch keine Diskussion
 - Ausschließlich positive Reaktionen

- Wichtige Faktoren
 - Anzahl der vorgeschlagenen SCMs groß
 - Diskussion kann ausarten
 - Erfahrung der Mitglieder mit gleichen SCMs
 - Diskussion bleibt kompakt
 - Machtverhältnisse unter den teilnehmenden Mitglieder
 - Beispiel KVM: von Maintainer verhindert
 - Mitgliedergewinnung
 - Populäres SCM wählen
 - Verhindern eine baldigen erneuten Migration
 - Wahl fällt auf SCM, welches aktiv weiterentwickelt wird

- Problem
 - In den Emails wird normalerweise nur über Probleme berichtet, das Mitverfolgen einer erfolgreichen Umsetzung ist daher leider nur selten möglich!

- Öffentliche Hosters für viele populäre SCMs
- Teilweise existieren inaktive Repositories
 - Für eine mögliche zukünftige Migration des SCM
 - Nicht mehr verwendete Repositories nach Migration
- Dezentrale SCMs
 - Hosters im Pull-Betrieb nicht unbedingt notwendig
 - Einrichtung von Accounts für Commits entfällt
 - Nicht allen ist es möglich ihr Repository öffentlich zu machen
 - Patch
 - Private Repositories auf öffentlichem Hosters (z.B. für Git: `repo.or.cz`)

- Patch-Betrieb
 - Maintainer stark belastet
 - Gefahr des Verlustes
 - Führte bei ROX zur Migration
- Commit-Betrieb
 - Accountverwaltung
 - Accountvergabe: Vertrauen vorhanden?
- Pull-Betrieb
 - Aufmerksamkeit per Email auf sich lenken

- Nutzung des gewünschten SCM über Adapter
- Aufgefallen in den Migrations-Diskussionen
 - Nicht als Innovator, nur als Befürworter
 - Drängen nicht zur Migration, verwenden bereits bevorzugtes SCM
- Verwendung eines einheitlichen SCM bei Teilnahme an unterschiedlichen Projekten
- Fast alle modernen SCM bieten den Import von Repositorys populärer SCMs
 - Einsatz bei Migration

- Verwendung einer Version, an der aktiv entwickelt wird
 - Meist Head
 - Regelmäßige Updates
 - Kein Verlass auf Stabilität
- Gründe
 - Gewünschtes Feature / Fehlerkorrektur nicht in Release vorhanden
 - Flyspray: Seltene Releases zwingen zur Verwendung von BE → Tester werden erreicht

- Trotz Verwendung eines dezentralen SCM ist oft ein Hauptrepository vorhanden
- Migration oft nicht wegen ermöglichter dezentraler Nutzung sondern wegen besserer Effizienz der modernen meist dezentralen SCMs
 - Volle lokale Versionskontrolle (Serverunabhängigkeit)
 - Wesentlich schneller
- Administrative Tätigkeiten müssen auf andere Art und Weise erforscht werden

Vielen Dank!