



DOKUMENTATION UND KOMMUNIKATION IN IT- FORSCHUNGSPROJEKTEN

Ringvorlesung „Themen der Softwareentwicklung“
WS 2011/12

Dipl.-Inf. Jens-Martin Loebel
Dipl.-Inf. Heinz-Günter Kuper

4.11.2011

ZIEL

- Die Vorlesungen werden eher **tutorial-** als akademischen Charakter haben - es geht um eine **Übersicht** über ein praktisches Thema, **Best Practices**, Hinweise auf typische **Anfängerfehler**, Angaben zu guter (und zu schlechter) **Literatur** oder auch **Tools** etc.
- Die Vorlesung richtet sich in erster Linie an die **(Bachelor-)Studenten**, die in diesem Semester ein Semesterprojekt machen ...

DIMENSIONEN DER SOFTWAREENTWICKLUNG



PROJECT TRIANGLE

Ihr Projekt kann

- schnell (Entwicklungszeit)
- gut (Zuverlässigkeit, Effizienz, ...)
- billig (Entwicklungs-, Personalkosten, ...)

sein.



Fast, Good or Cheap. Pick two.

Wählen Sie zwei Eigenschaften!

* Quelle: http://en.wikipedia.org/wiki/Project_triangle

Bild aus urheberrechtlichen Gründen entfernt
Quelle: Dilbert Comic 14. April 2001

online unter:

<http://dilbert.com/strips/comic/2001-04-14/>

Bild aus urheberrechtlichen Gründen entfernt
Quelle: „User Interface = Magic“, The Next Web, 27.10.2009

online unter:

<http://thenextweb.com/shareables/2009/10/27/user-interface-magic/>

ARTEN VON DOKUMENTATION

- Quellcode-Kommentare
- Logs / Protokolle
- technische Spezifikationen, Pflichten- / Lastenheft
- Nutzerhandbuch

VERSIONIERUNG: EINFÜHRUNG

- Revisionsgeschichte eines Projektdokumentes erhalten / nachvollziehen
- Ziele
 - Kollaboration
 - Konfliktvermeidung / -handling
 - „Backup“ / Fehler rückgängig machen
 - Alternativen gefahrlos ausprobieren

VERSIONIERUNG: GRUNDBEGRIFFE

- Commit / Check-In
 - neue Revision zum Repository hinzufügen, Changelog erzeugen
- Checkout
 - Revision einer Datei / eines Projektes aus dem Repository laden
- Change / Diff
 - Änderungen zwischen zwei Revisionen / Versionsunterschied

VERSIONIERUNGSSYSTEME (OPEN SOURCE)

- Concurrent Versions System (CVS)
- Subversion (SVN)
- git
- Mercurial (hg)



VERSIONSSYSTEM VERZEICHNISSTRUKTUR

- /trunk
 - Sandbox, Hauptentwicklungsstrang, aktuelle Entwicklungsversion
- /branches
 - Isolierte Nebenentwicklungen, Architekturänderungen
- /tags
 - Stabile Versionen, durchnummerierte Releases der Software

SUBVERSION



SUBVERSION

- GRAPHISCHE CLIENTS
 - LINUX
 - RapidSVN (<http://rapidsvn.tigris.org/>)
 - MAC
 - Versions (<http://versionsapp.com/>)
 - WIN
 - TortoiseSVN (<http://tortoisesvn.net/>)

SUBVERSION

- HILFE
 - `svn help command`
- REPOSITORY ANLEGEN (auf dem Server)
 - `svnadmin create /Users/Shared/svnrepos`
 - `chown -R www:admin /Users/Shared/svnrepos`

SUBVERSION

- CHECKOUT WORKING COPY
 - `svn --username user co http://www.example.com/repos`
 - `svn co http://www.example.com/repos`

SUBVERSION

- Versions.app Demo



SUBVERSION

- BEARBEITUNGSSTAND ÜBERPRÜFEN (WORKING COPY)
 - `svn status`
- DATEIEN HINZUFÜGEN
 - `svn add file.name`
- DATEIEN LÖSCHEN
 - `svn remove file.name`

SUBVERSION

- COMMIT WORKING COPY
 - `svn ci -m "log message" file/dir/...`
 - `~.bash_profile` oder `~.subversion/config`
 - `export SVN_EDITOR="vi"`
 - `svn ci`

SUBVERSION

- REVERT OHNE PROTOKOLL ZU VERLIEREN
 - `svn merge --dry-run -r NEWER:OLDER .`
 - `svn merge -r NEWER:OLDER .`

SUBVERSION

- `svn propset svn:keywords "Id" file.name`
 - In Datei *file.name*:
 - `//Id` wird ersetzt mit
 - `//$Id: KinderGuideServer.m 119 2011-10-20 17:10:10Z kuper $`
- `svn propset svn:ignore 'build' .`
 - Ignore build dir in cwd

SUBVERSION

- ANDERE SINNVOLLE BEFEHLE
 - svn up
 - svn mkdir
 - svn remove
 - svn move
 - svn diff
 - svn log *file.name*

TRAC + SUBVERSION

- DEMO:
 - <http://trac.edgewall.org/demo-0.12/browser>

SUBVERSION

LITERATUR UND TUTORIALS

- Version Control with Subversion
 - <http://svnbook.red-bean.com/>
 - <http://svnbook.red-bean.com/index.de.html> (deutsche Fassung)
- Subversion: A Quick Tutorial
 - <http://aymanh.com/subversion-a-quick-tutorial>

BEST PRACTICES (SUBVERSION)

- VOR Beginn der Arbeit und VOR dem Einchecken
 - SVN UPDATE !
- Binärdateien locken (Mediendateien, etc.)
- Sprechende Changelogs beim Einchecken
 - +/-/*-Notation benutzen
- Je mehr Dokumentation, desto einfacher die Zusammenarbeit
- Nur lauffähige Versionen einchecken!

EIN WORT ZU VERSIONSNUMMERN VON DOKUMENTEN

- nicht nur Code kann mit SVN etc. verwaltet werden, sondern auch Binärdateien und andere Projektdateien
- unabhängig davon sollte jedes Projektdokument eine Versionsnummer erhalten
- Best Practice
 - Datum oder
 - SVN Build ID oder
 - major/minor/patch Label als Versionsnummer

GIT

- Verteiltes Versionskontrollsystem
 - alle Entwickler besitzen lokale Kopie der gesamten Revisionsgeschichte
- ursprünglich von Linus Torvalds entwickelt
- Dateisystem-Metapher mit Versionierungsansätzen

GIT-OBJEKTE

- blob-Objekt
 - Bitstrom einer Datei, ohne Metadaten (Datum, Name, ...)
- tree-Objekt
 - repräsentiert ein Verzeichnis, enthält die Metadaten der Blob-Objekte, referenziert diese und/oder symbolische Links auf weitere trees
- commit-Objekt
 - verbindet tree-Objekte zu einer gemeinsamen Revisionsgeschichte
- tag-Objekt
 - virtueller Container für ein anderes Objekt + Metadaten, oft für Signaturen von Commit-Objekten genutzt

GIT

WICHTIGE BEFEHLE

- git init
- git add .
- git commit -m "<CHANGELOG>"
 - Flag: -a
- git log
- git diff

BUG-TRACKING



BUG-TRACKING

- VORTEILE

- Gestaltet Kommunikation zwischen Entwicklern und Anwendern
- Kann als Wissensbasis (*Knowledge Base*) verwendet werden.
- Man kann auf bereits gemeldete Bugs explizit hinweisen

BUG-TRACKING

- NACHTEILE
 - Verwaltungsaufwand
 - Änderungen im System müssen im Bugtracker widerspiegelt werden (z.B. neue Versionen, Komponente)

BUG-TRACKING

- TICKETING-SYSTEM
 - Synonym für Bug-Tracking
 - Bug-Tracking Untermenge von Issue-Tracking
- SYSTEME
 - http://en.wikipedia.org/wiki/Comparison_of_issue_tracking_systems
 - <http://www.bugzilla.org/>

BUG-TRACKING

ID	Summary	Status	Opened	Assignee	Submitter	Resolution	Priority
Assignee: <input type="text" value="Any"/> Status: <input type="text" value="Any"/> Category: <input type="text" value="Any"/> Group: <input type="text" value="Any"/> Submitter: <input type="text"/> Keyword: <input type="text"/> Artifact ID: <input type="text"/> <input type="button" value="Filter"/> <input type="button" value="Reset"/> <input type="button" value="Permalink"/>							
<input type="checkbox"/> 3091465	Error while trying to access group via HIEditor	Open	2010-10-20	nobody	joerosario	None	5
<input type="checkbox"/> 2952125	import fehler	Open	2010-02-15	nobody	sculpture	None	5
<input type="checkbox"/> 2943748	Editor startet nicht (MfN-Berlin)	Closed	2010-02-01	nobody	mpingel	Invalid	5
<input type="checkbox"/> 2911963	Schwarze Ränder bei Bildern mit Kantenlänge < 128px	Closed	2009-12-10	hgkuper	ericdecker	Wont Fix	2
<input type="checkbox"/> 2906985	Nachtrag zu "diverse Fehlermeldungen"	Closed	2009-12-01	jmlobel	carmenwedemeyer	Accepted	5
<input type="checkbox"/> 2906978	Diverse Fehlermeldungen	Closed	2009-12-01	jmlobel	carmenwedemeyer	Duplicate	5
<input type="checkbox"/> 2883152	Farbeinstellungen aus der Bibliothek	Closed	2009-10-21	jmlobel	carmenwedemeyer	Wont Fix	5
<input type="checkbox"/> 2872213	Upload fehlerhaft	Closed	2009-10-03	hgkuper	carmenwedemeyer	Rejected	1
<input type="checkbox"/> 2872206	Reimport	Closed	2009-10-03	jmlobel	carmenwedemeyer	Fixed	5
<input type="checkbox"/> 2872202	Titelanzeige Objekte/Ansichten	Closed	2009-10-03	jmlobel	carmenwedemeyer	Later	2
<input type="checkbox"/> 2840943	Unicode Diakritika auf Lichttischen	Closed	2009-08-20	hgkuper	ericdecker	Invalid	5



BUG-TRACKING

Add Artifact

Please enter all bug reports relating to the software using this Bug Tracker. There are separate trackers for Feature Requests and Support Requests. You are welcome to enter your bugs in English or German.

Project: HyperImage

Category: (admin)

Private:

Group: (admin)

Assigned To: (admin)

Priority:

Summary:

Description:

Upload a file attachment

File:

BUG-TRACKING

- BEST PRACTICES
 - keine anonyme Nutzer (wegen Rückfragen)
 - nicht aufregen und nicht drängeln lassen

BUG-TRACKING

- BUG REPORT
 - Wichtigkeit einstufen:
 - blocker, critical, major, trivial, usw.
 - Klare Konfigurationszuweisung
 - welche Umgebung (WLAN, Betriebssystem)
 - welche Softwareversion

BUG-TRACKING

- BUG REPORT
 - Steps to Repeat / Reproducibility
 - Der Entwickler muss genug Information haben um das System(fehl)verhalten zu reproduzieren
 - Logdateien und Screenshots anhängen

BUG-TRACKING

- BUG REPORT
 - Rückmeldung/Resolution
 - fixed
 - duplicate
 - works for me
 - invalid
 - won't fix

BUG-TRACKING

- SITES
 - <http://sourceforge.net/>
 - <https://github.com/>
- TRAC DEMO
 - <http://trac.edgewall.org/demo-0.12/newticket>

WEITERFÜHRENDE LITERATUR

- Hunt, A; Thomas, D; Cunningham, W.: *The Pragmatic Programmer. From Journeyman to Master*. Addison-Wesley Longman, Amsterdam, 1999.
- Brooks, F. P.: *The Mythical Man-Month. Essays on Software Engineering*. Addison-Wesley Longman; Jubiläumsausgabe, 1995.

DISKUSSION

- Anregungen?
- Fragen?
- Sorgen?
- Nöte?