

XION IT SYSTEMS

AKTIENGESELLSCHAFT

Dresdnerstraße 81-85/8.Stock
A-1200 Wien

Tel: 0664-8242-600

E-mail: office@xion.at

Web: xion.at

Festnetz: +43/1/333 91 99-0

Fax: +43/1/333 91 99-199

x i o n . i t systems . a g



Software Wartung und Evolution

Dipl.-Ing. Dr. techn. Johannes Weidl-Rektenwald
Xion IT Systems AG

XION IT SYSTEMS

AKTIENGESELLSCHAFT

Dresdnerstraße 81-85/8.Stock
A-1200 Wien

Tel: 0664-8242-600

E-mail: office@xion.at

Web: xion.at

Festnetz: +43/1/333 91 99-0

Fax: +43/1/333 91 99-199

x i o n . i t systems . a g



Lecture 5

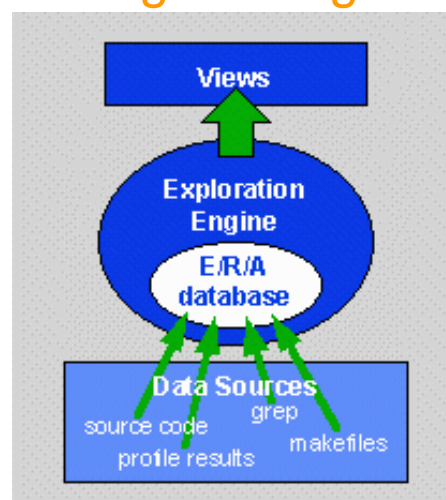
Lecture 5

- Inhalte
 - Tool Demo
 - Reverse Engineering mit Imagix4D
 - Refactoring mit IntelliJ IDEA
 - Architektur- und Evolutionsanalyse mit dem Sotograph
 - Eclipse mit dem CREOLE Plugin
 - Organisation der Wartung
 - Software Configuration Management
 - Defect Tracking und Change Tracking
 - Software Artifact Management
 - Produktivstellung

© J. Weidl-Rektenwald 02-07

142

Reverse Engineering: Imagix 4D



© J. Weidl-Rektenwald 02-07

143

Refactoring: IntelliJ IDEA

```

public class ListUsersAction extends Action {
    /**
     * @see ActionSupport(ActionMapping, ActionForm, HttpServletRequest, HttpServletResponse)
     */
    public ActionForward perform(ActionMapping mapping,
        ActionForm form,
        HttpServletRequest request,
        HttpServletResponse response)
        throws IOException, ServletException {
        // Validate the request parameters specified by the user
        ActionErrors errors = new ActionErrors();

        ListUsersForm myform = (ListUsersForm) form;

        try {
            if (StringUtil.isLoggedin(request)) {
                UserService user = UserServiceUtil.getUser();
                UserService us = user.create();

                Collection result = null;

                String id = request.getParameter("id");
                if (id == null) {
                    result = us.searchAllUsers();
                } else {
                    result = us.search(id);
                    result.add(us.searchById(id));
                }

                myform.setExistingUsers(result);
            }
        }
    }
}

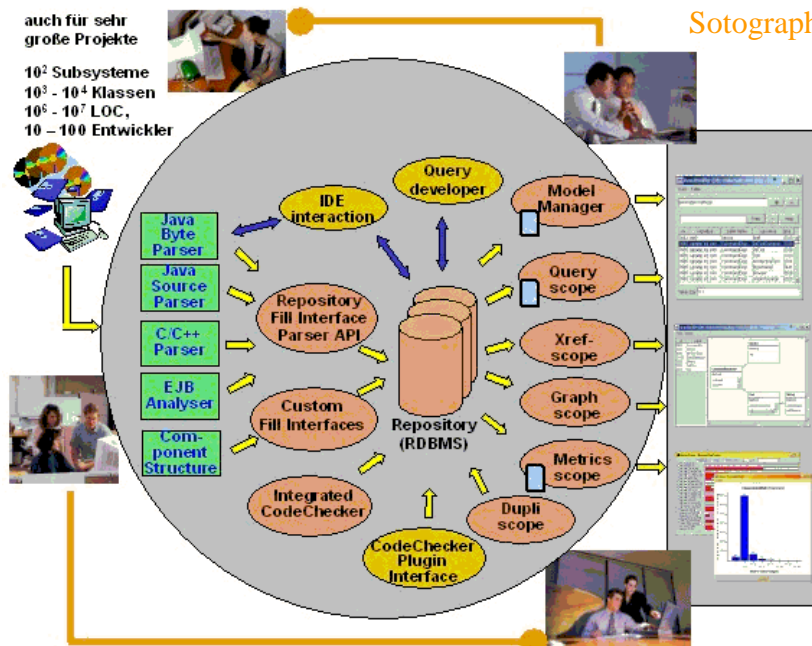
```

© J. Weidl-Rektenwald 02-07

144

auch für sehr große Projekte
 10² Subsysteme
 10³ - 10⁴ Klassen
 10⁶ - 10⁷ LOC,
 10 - 100 Entwickler

Sotograph



CREOLE

- Eclipse plugin by „The Chisel Group“
 - Computer Human Interaction & Software Engineering Lab
 - [<http://www.thechiselgroup.org/creole>]
- „Creole is the term used to describe our plug-in to the Eclipse platform which integrates SHriMP with the Eclipse platform's Java Development Tools (JDT).“
- SHriMP = **S**imple **H**ierarchical **M**ulti-**P**erspective

© J. Weidl-Rektenwald 02-07

146

Software Configuration Management (SCM)

Software Configuration Management: Begriffsdefinition

- Software Artifact Management (SAM)
 - Management von versionierten Software Komponenten
- Software Configuration Management (SCM)
 - SAM in Verbindung mit Defect und Change Tracking

SAM Features

- Version Control und Release Management
- Workspace Management
- Build Management
- Process Configurability
- Parallel Development
- Distributed Development

SCM Features

- Defect Tracking
 - Management von Software Bugs
 - Erfassung, Life Cycle Management, Reporting
- Change Tracking
 - Management von Change Requests
 - Erfassung, Life Cycle Management, Reporting
- Essentiell in der Software Wartung!

Defect Tracking und Change Tracking

Fehlermeldung/Änderungsantrag

- Erfassung über
 - Call Center
 - Bug Tracking Tool
 - Email
 - Telefon, Fax
 - ...
- Das professionelle Management von Fehlermeldungen wird als *Defect Tracking* bezeichnet
- Das professionelle Management von change requests wird als *Change Tracking* bezeichnet
- Defect Tracking und Change Tracking werden umfassend als *Change Management* bezeichnet

© J. Weidl-Rektenwald 02-07

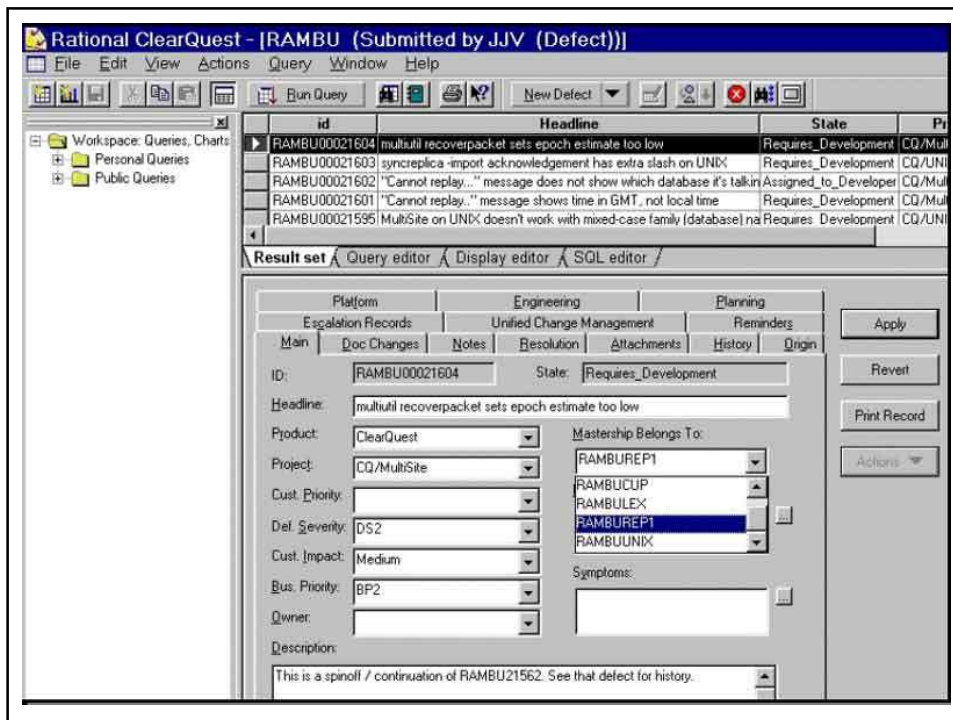
152

Spezifikation eines Defects

- Kurzbeschreibung
- Zeitstempel des Auftretens
- Submitter (Name, Organisation)
- Längere Beschreibung zum Nachvollziehen der Fehlersituation durch den Ingenieur
- Komponente, in der der Fehler aufgetreten ist (Server, Client, GUI, Connectivity, ...)
- Priorität

© J. Weidl-Rektenwald 02-07

153



Life Cycle eines Defects

- Submission (z.B. per mail, Web, etc.)
- Assignment
 - Der Verantwortliche Manager trifft eine Zuteilung des Defects an einen Wartungsingenieur (Asignee)
- Eventuell Reassignment
 - Der Asignee weist den Defect begründet zurück
- Resolution
 - Der Defect wird mit einer Fehlerbehebungsbeschreibung als gefixt (oder resolved) klassifiziert
- Verification
 - Die Behebung wird als erfolgreich klassifiziert
- Closing

Eigenschaften von Defect Tracking Tools

- Bilden den komplette Life Cycle von der Meldung bis zur Schließung ab
- Speichern Defect Reports in Datenbanken
- Liefern umfassende (auch grafische) Reports
 - Defects per Programmer (assigned, fixed)
 - Defects per Month
 - Defects per Module etc.
- Daraus können Meßgrößen für den Zustand der Software und die Effizienz der Organisation abgeleitet werden
- Trend: Defect Tracking Tools entwickeln sich zu Change Management Tools

© J. Weidl-Rektenwald 02-07

156

Probleme von Defect Tracking Tools

- Mangelnde Integration
 - in die Entwicklungsumgebung
 - Link von einer Fehlermeldung direkt in den Source Code des Moduls
 - Konfiguration des Debuggers nach der Fehlermeldung (Setzen von Breakpoints, Watches, etc.)
 - in Management Tools
 - in Tools, die andere Entwicklungsphasen betreffen
 - z.B. Aufnahme aller change requests in ein Anforderungsmanagement Tool

© J. Weidl-Rektenwald 02-07

157

Evaluierung von Defects

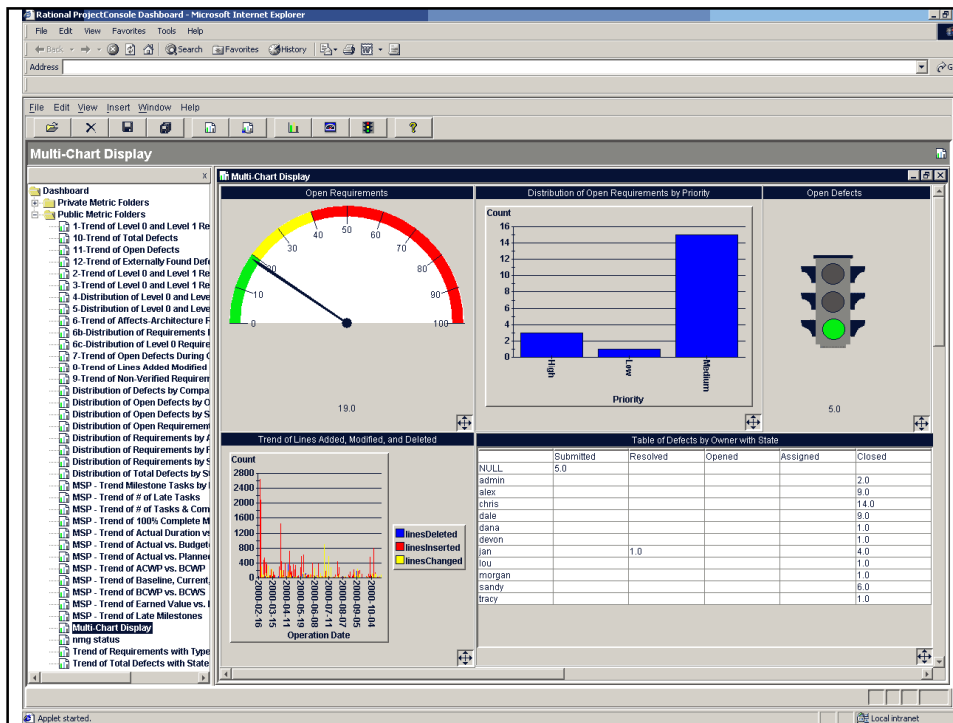
- Wird meist toolunterstützt
 - Low end
 - z.B. Excel
 - High End
 - z.B. Rational Produkte
- Gewinnung von Kenndaten als Input für das (Projekt-)controlling

© J. Weidl-Rektenwald 02-07

158

The screenshot shows the Rational Project Console interface. The main window displays a table of defect query results. The table has four columns: ID, Headline, State, and Priority. The data rows list various defects with their corresponding IDs and descriptions.

ID	Headline	State	Priority
CLSLIC00000037	spelling error in login screen	Resolved	3-Normal Queue
CLSLIC00000038	sales tax incorrect if item deleted from purchase	Resolved	2-Give High Attention
CLSLIC00000039	cancel sale doesn't correctly repaint screen	Resolved	3-Normal Queue
CLSLIC00000040	columns out of alignment	Resolved	2-Give High Attention
CLSLIC00000041	delete item not working correctly	Opened	2-Give High Attention
CLSLIC00000042	override price does not work	Resolved	2-Give High Attention
CLSLIC00000043	alt-C does not invoke cancel operation	Resolved	3-Normal Queue
CLSLIC00000044	clerk allowed to charge too much on credit card	Resolved	3-Normal Queue
CLSLIC00000045	end-of-shift report fails if after midnight	Resolved	3-Normal Queue
CLSLIC00000046	too many spaces in 'change due' field	Assigned	3-Normal Queue
CLSLIC00000047	delete operation leaves blank line in form	Resolved	3-Normal Queue
CLSLIC00000048	add item button not active after adding 3 other items	Resolved	1-Resolve Immediately
CLSLIC00000049	sales tax incorrect for NH	Assigned	1-Resolve Immediately
CLSLIC00000050	credit card refused message is unclear	Resolved	3-Normal Queue
CLSLIC00000051	inventory report is not running correctly	Assigned	1-Resolve Immediately
CLSLIC00000052	delete item button deletes two items	Resolved	2-Give High Attention
CLSLIC00000053	overriding price operation allows negative number	Assigned	2-Give High Attention
CLSLIC00000054	heading of application looks too crowded	Resolved	2-Give High Attention
CLSLIC00000055	part number column not wide enough	Opened	3-Normal Queue
CLSLIC00000056	add item button is out of line with the other buttons	Assigned	3-Normal Queue
CLSLIC00000057	context sensitive help fails from reorder window	Assigned	2-Give High Attention
CLSLIC00000058	formatting does not look right in inventory report	Opened	3-Normal Queue
CLSLIC00000059	add items fails for large quantities	Opened	2-Give High Attention
CLSLIC00000060	spelling error in cancel sale help	Resolved	2-Give High Attention
CLSLIC00000061	shortcut to logout does not work	Assigned	3-Normal Queue



Defect Tracking als statistisches Orakel

- Durch statistische Auswertungen der Defect Daten können „malicious modules“ entdeckt werden
- Das sind Module, die besonders oft und/oder von vielen verschiedenen Personen geändert werden
- Deutet hin auf mangelhaftes Design, zu komplexe Programmierung, bis hin zu organisatorischen Schwächen in den Geschäftsprozessen der Organisation

Software Artifact Management – Warum?

- Nachvollziehbarkeit / Traceability
 - Wer hat wann was verändert? (Historie, Versionsvergleich)
 - Warum? (Version Comments)
- Parallele Entwicklung
 - Mehrere Entwickler arbeiten auf demselben File auf verschiedenen *private branches (aka streams)*
 - Mehrere Versionen eines Files werden zu einem gemeinsamen Nachfolger *merged*
- Möglichkeit der Reproduktion aller ausgelieferten Versionen
 - verschiedene Kunden
 - verschiedene Zielplattformen
 - verschiedene Feature-Zusammenstellung

© J. Weidl-Rektenwald 02-07

162

Versionsmanagement

- Versionen entstehen durch
 - Bug fixes
 - Anpassung an neue Hardware, Betriebssysteme, Libraries
 - Erweiterungen
 - ...
- Eben durch Einflüsse von Wartung und Evolution

© J. Weidl-Rektenwald 02-07

163

Versionsmanagement

- Typische Funktionen eines SCM Tools
 - Check in / check out
 - Labeling
 - Textual Difference
 - Branching
 - Merging
 - Views
 - Search
 - Reporting
 - History

© J. Weidl-Rektenwald 02-07

164

Versionsmanagement

- Unter Versionsverwaltung stehen
 - Source Code
 - Script files (batch files / shell scripts)
 - html, xml files
 - Javascript files
 - Makefiles, Projektfiles
 - Libraries
 - System Drivers
 - Dokumentation (!)

© J. Weidl-Rektenwald 02-07

165

Versionsmanagement

- Was wird **nicht** versionsverwaltet
 - Alle generierten Artefakte
 - Ausführbare Programme (exe, class files, binaries)
 - Generierte Dokumentation (z.B. javadoc)
 - Temporäre Files

Versionsmanagement Tipps

- SCM sollte vor Beginn eines Projektes aufgesetzt und stabil sein
- Es sollten keine ableitbaren Artefakte eingecheckt werden (Executables, JavaDoc)
- Jede Version, die an den Kunden geht, muss vollständig gelabelt sein (Reproduktion)!
- SCM Repository muss ins Backup!

Build Management

- „Build“
 - Aus den versionierten Komponenten wird eine Produktionsversion erzeugt
 - Probleme
 - Welche Komponenten müssen aufgrund von Abhängigkeiten neu erzeugt werden?
 - Welche Komponenten können wieder verwendet werden?
 - Welche Komponenten in welchen Versionen fanden in welchen Builds Verwendung?
 - „Stückliste“

© J. Weidl-Rektenwald 02-07

168

Release Management

- Ein „Release“ ist ein auslieferbares Produkt definierten Umfangs
 - Sollte jederzeit reproduziert werden können
 - Was befand sich in dieser Release?
 - Dokumentiert durch Release Notes
 - Kann durch Tools (semi-)automatisch geschehen
 - Einpflegen der Release in Dokumentation
 - Release für wen? Wann ausgeliefert? Durch wen? Wann abgenommen? Warum ausgeliefert? Welche Release wurde ersetzt?

© J. Weidl-Rektenwald 02-07

169

Tools

- SCCS (Source Code Control System)
- RCS (Revision Control System)
- CVS (Concurrent Versions System)
- Microsoft Visual Source Safe (VSS)
- Merant PVCS
- Rational ClearCase
- uvm.

© J. Weidl-Rektenwald 02-07

170

Produktivstellung

Wartungsfenster

- Ein Wartungsfenster ist ein **begrenzt**es Zeitintervall, in dem ein Produktionssystem für Wartungsarbeiten außer Betrieb geht
 - Das System muss **definiert** außer Betrieb genommen werden (Ankündigung und Wartungsmeldung)
 - Die Wartungsarbeiten müssen genau **geplant** werden (Projektmanagement)
 - Definition des „Point of no return“
 - Ab diesem Zeitpunkt kann nicht mehr auf das alte System zurückgestellt werden (außer durch Einspielen eines Backups)
 - Definition des Wiederanlaufes

© J. Weidl-Rektenwald 02-07

172

Checkliste Wartungsfenster

- **Voraussetzung:** Freigabe der neuen Produktions-Release
- Projektplan erstellen
 - **Ressourcen/Rollen** festlegen
 - Detaillierter **Ablaufplan** (Work Breakdown Structure), Definition von Go/No-Go bzw. **Rollback** Punkten
 - **Dauer** definieren
- Projektplan durch **Tests** verifizieren
- Termin mit dem Systemverantwortlichen bzw. den Benutzern (intern, eventuell extern) vereinbaren
- Verteilen des **Projektplans** an die Akteure (Betriebsführungspersonal, Wartungspersonal, etc.)

© J. Weidl-Rektenwald 02-07

173