

**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 . it systems ag



# 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 . it systems ag

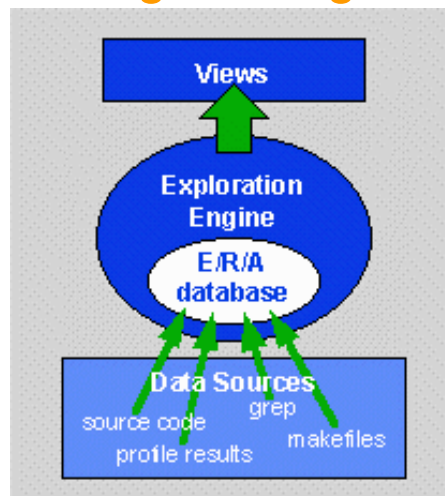


# Lecture 4

# Lecture 4

- Inhalte
  - Tool Demo
    - Reverse Engineering mit Imagix4D
    - Refactoring mit IntelliJ IDEA
    - Architektur- und Evolutionsanalyse mit dem Sotograph
  - Organisation der Wartung
    - Nachtrag aus Lecture 3
      - Management des Wartungsfalles
      - Life Cycle Modelle der Software Wartung
    - Software Configuration Management
      - Defect Tracking und Change Tracking
      - Software Artifact Management
    - Produktivstellung

## Reverse Engineering: Imagix 4D



# Refactoring: IntelliJ IDEA

```

public class ListItemsAction extends Action {
    // ...
    public ActionForward perform(ActionMapping mapping,
        ActionForm form,
        HttpServletRequest request,
        HttpServletResponse response)
        throws ServletException, ServletException {
        // Validate the input parameters specified by the user
        ActionErrors errors = new ActionErrors();

        ListItemsForm myForm = (ListItemsForm) form;

        try {
            if (!myForm.isLoggedOn(request)) {
                HttpServletResponse web = HttpServletResponse.getWriter();
                HttpServletResponse web.create();

                Collection result = null;

                String id = request.getParameter("id");
                if (id != null) {
                    result = myForm.getAllItems();
                } else {
                    result = new Vector();
                    result.add(myForm.getAllItems());
                }

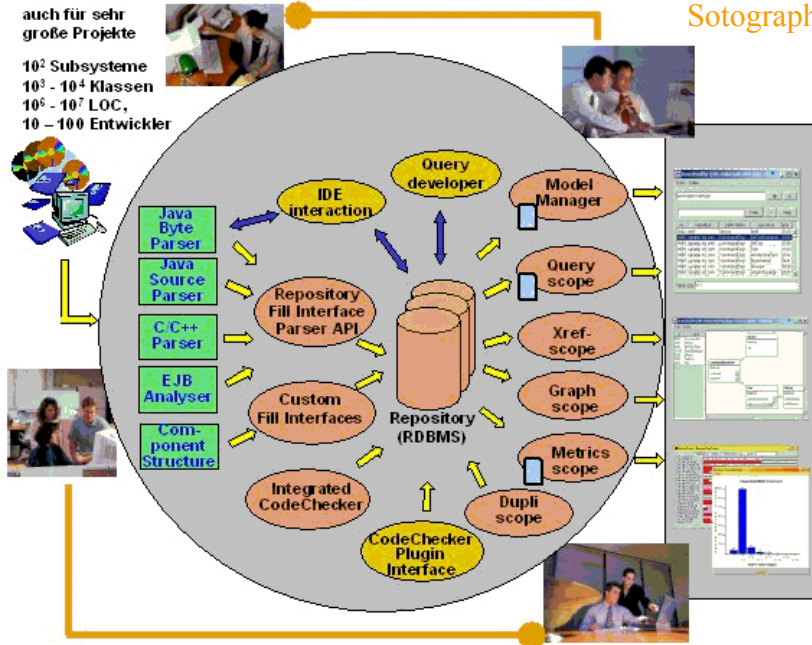
                myForm.populateForm(result);
            }
        }
    }
}
    
```

© J. Weidl-Rektenwald 02-06

142

auch für sehr große Projekte  
 10<sup>2</sup> Subsysteme  
 10<sup>3</sup> - 10<sup>4</sup> Klassen  
 10<sup>6</sup> - 10<sup>7</sup> LOC,  
 10 - 100 Entwickler

## Sotograph



# 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

## 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

The screenshot displays the Rational ClearQuest interface for a defect record. The window title is "Rational ClearQuest - [RAMBU (Submitted by JJV (Defect))]". The main area shows a table of defect records with columns for ID, Headline, State, and Priority. The selected record is RAMBU00021604 with the headline "multitil recoverpack.et sets epoch estimate too low" and state "Requires\_Development".

id	Headline	State	Pr
RAMBU00021604	multitil recoverpack.et sets epoch estimate too low	Requires_Development	CQ/Mult
RAMBU00021603	syncreplica -import acknowledgement has extra slash on UNIX	Requires_Development	CQ/UNI
RAMBU00021602	"Cannot replay..." message does not show which database it's talking	Assigned_to_Developer	CQ/Mult
RAMBU00021601	"Cannot replay..." message shows time in GMT, not local time	Requires_Development	CQ/Mult
RAMBU00021595	MultiSite on UNIX doesn't work with mixed-case family (database) na	Requires_Development	CQ/UNI

Below the table, the "Result set" section shows the "Query editor" tab. The defect details form includes the following fields:

- ID: RAMBU00021604
- State: Requires\_Development
- Headline: multitil recoverpack.et sets epoch estimate too low
- Product: ClearQuest
- Project: CQ/MultiSite
- Cust. Priority: [Empty]
- Def. Severity: DS2
- Cust. Impact: Medium
- Bus. Priority: BP2
- Owner: [Empty]
- Description: This is a spinoff / continuation of RAMBU21562. See that defect for history.

On the right side of the form, there are buttons for "Apply", "Revert", "Print Record", and "Actions".

## 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
- Verficiation
  - Die Behebung wird als erfolgreich klassifiziert
- Closing

## Eigenschaften von Defect Tracking Tools

- Bilden den komplette Life Cylce 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



# 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

# Evaluierung von Defects

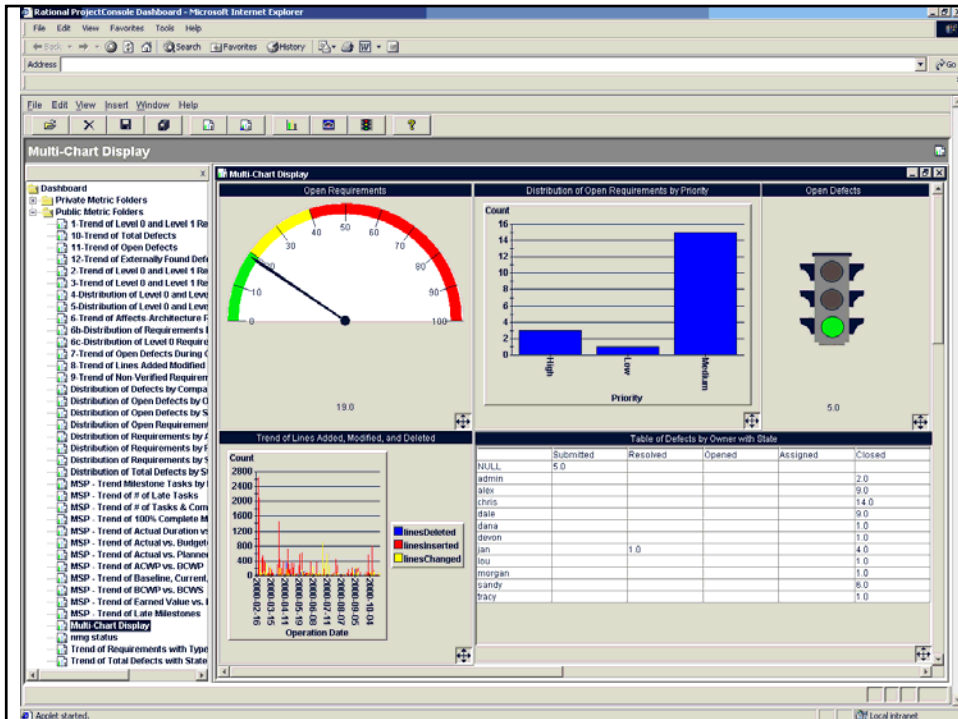
- Wird meist toolunterstützt
  - Low end
    - z.B. Excel
  - High End
    - z.B. Rational Produkte
- Gewinnung von Kenndaten als Input für das (Projekt-)controlling

Rational Project Console - Microsoft Internet Explorer

Address: Rational.com Home About Update

### Query Results: Public Queries/All Defects

ID	Headline	State	Priority
CL.SIC00000037	spelling error in login screen	Resolved	3-Normal Queue
CL.SIC00000038	sales tax incorrect if item deleted from purchase	Resolved	2-Give High Attention
CL.SIC00000039	cancel sale doesn't correctly reprint screen	Resolved	3-Normal Queue
CL.SIC00000040	columns out of alignment	Resolved	2-Give High Attention
CL.SIC00000041	delete item not working correctly	Open-d	2-Give High Attention
CL.SIC00000042	override price does not work	Resolved	2-Give High Attention
CL.SIC00000043	ab-C does not invoke cancel operation	Resolved	3-Normal Queue
CL.SIC00000044	clerk allowed to charge too much on credit card	Resolved	3-Normal Queue
CL.SIC00000045	end-of-shift report fails if after midnight	Resolved	3-Normal Queue
CL.SIC00000046	too many spaces in 'change due' field	Assigned	3-Normal Queue
CL.SIC00000047	delete operation leaves blank line in form	Resolved	3-Normal Queue
CL.SIC00000048	add item button not active after adding 3 other items	Resolved	1-Resolve Immediately
CL.SIC00000049	sales tax incorrect for NIH	Assigned	1-Resolve Immediately
CL.SIC00000050	credit card refund message is unclear	Resolved	3-Normal Queue
CL.SIC00000051	inventory report is not running correctly	Assigned	1-Resolve Immediately
CL.SIC00000052	delete item button deletes two items	Resolved	2-Give High Attention
CL.SIC00000053	overriding price operation allows negative number	Assigned	2-Give High Attention
CL.SIC00000054	heading of application looks too crowded	Resolved	2-Give High Attention
CL.SIC00000055	part number column not wide enough	Open-d	3-Normal Queue
CL.SIC00000056	add item button is out of line with the other buttons	Assigned	3-Normal Queue
CL.SIC00000057	context sensitive help fails from reorder window	Assigned	2-Give High Attention
CL.SIC00000058	formatting does not look right in inventory report	Open-d	3-Normal Queue
CL.SIC00000059	add items fails for large quantities	Open-d	2-Give High Attention
CL.SIC00000060	spelling error in cancel sale help	Resolved	2-Give High Attention
CL.SIC00000061	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

# Versionsmanagement

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

# Versionsmanagement

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

# Versionsmanagement

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

# 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“

## 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?

## Tools

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

# 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



## 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.)