



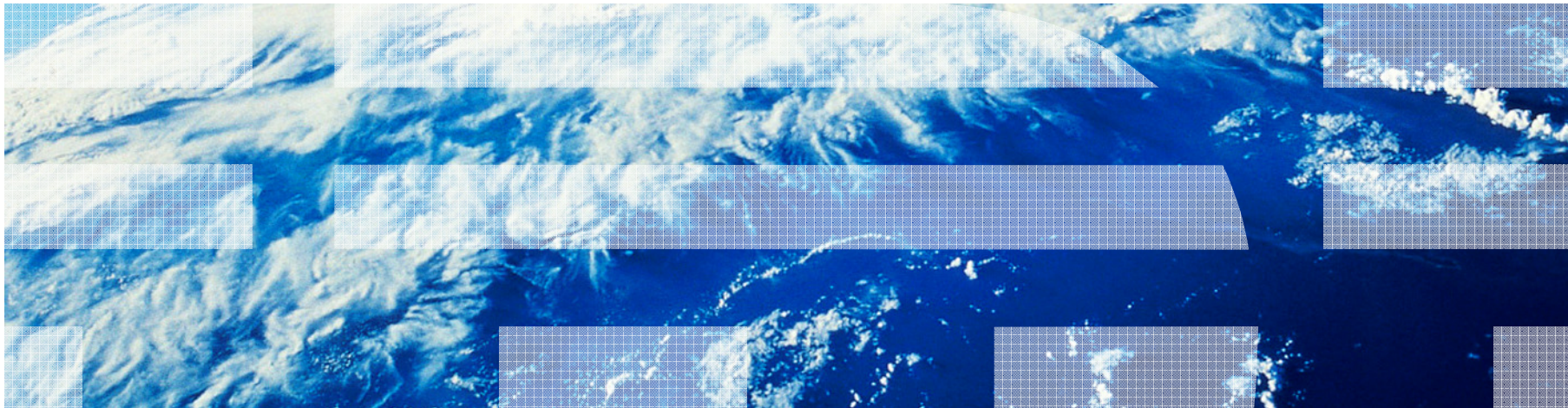
Werner Schoepe
IBM Rational
werner.schoepe@de.ibm.com

Rene Meyer
IBM Rational
Rene.Meyer@de.ibm.com

Lothar Kappen
IBM Rational
kappen@de.ibm.com

Agil zum Ziel mit Jazz

Jazz Roadshow



- **Agile Softwareentwicklung**
- **Werkzeuge für agile Projekte**
- **Jazz: eine Werkbank für C/ALM**



**over
processes and tools**



**over
comprehensive documentation**



**over
contract negotiation**



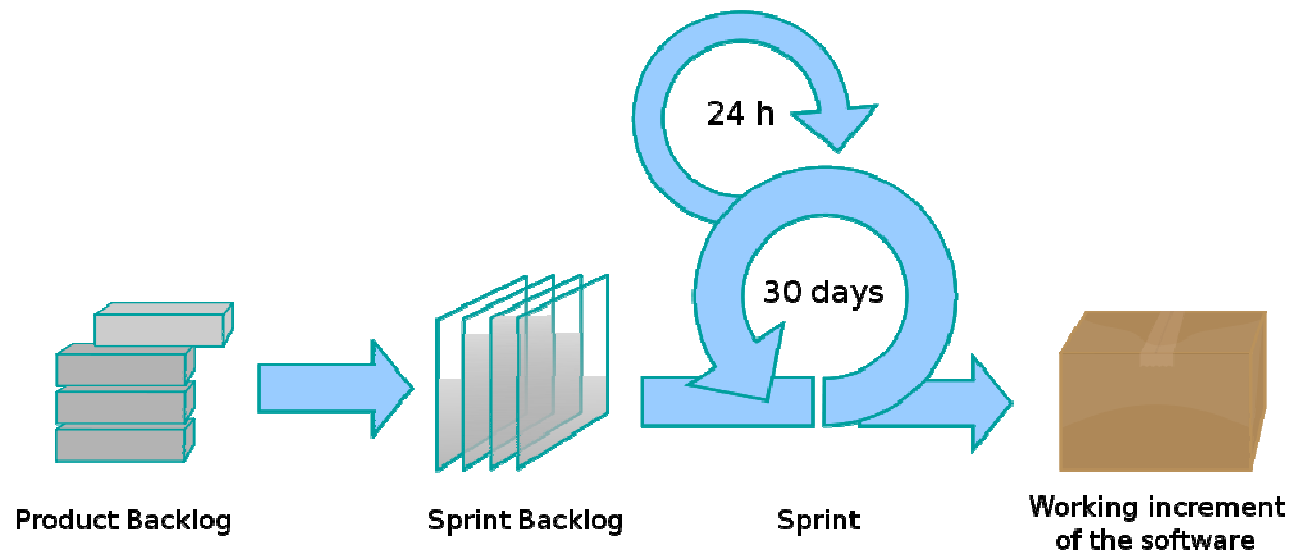
**over
following a plan**

Manifesto for Agile Software Development

- Iteratives und inkrementelles Entwickeln
- Zweistufiges Planen
- Häufige Releases
- Selbst organisierende Teams
- Andauernde Kundenbeteiligung
- Zeitnahe Anforderungsdefinition
- Ständiges Testen
- Laufende Integration
- Refactoring
- Unit testing & Test Driven Development (TDD)
- Retrospektiven



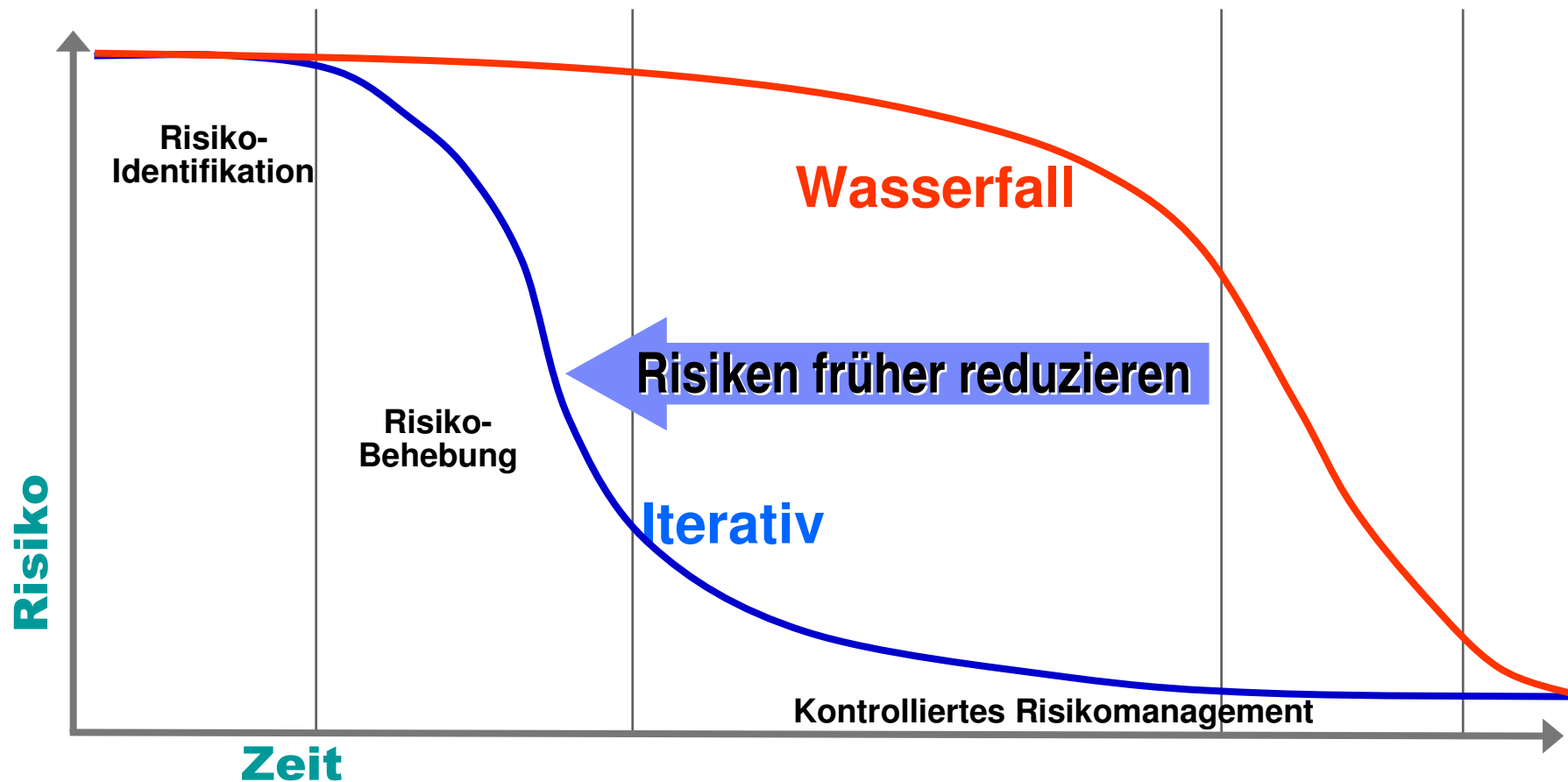
Scrum Lebenszyklus



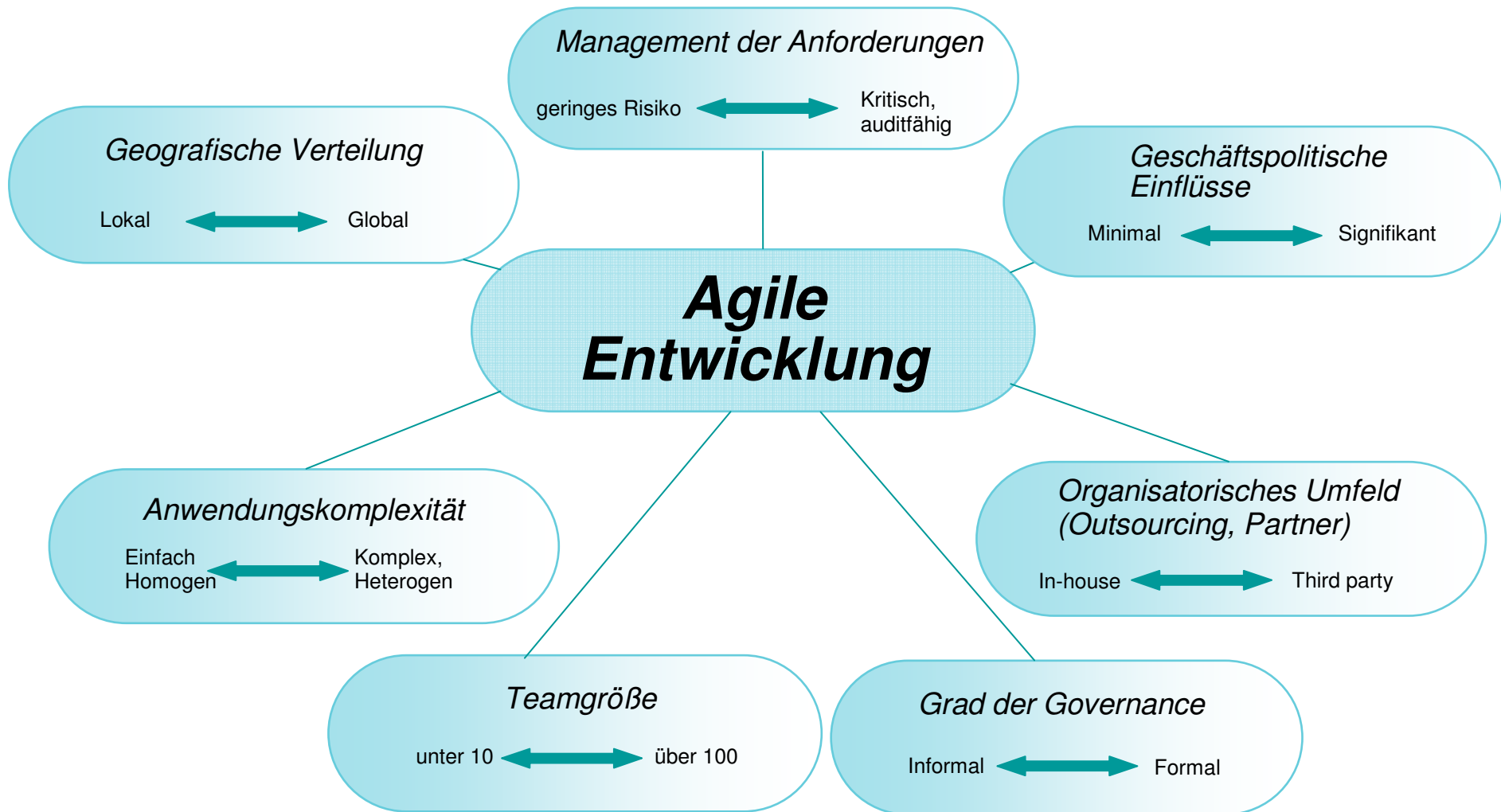
Verbesserung des Prozesses



- Kollaborativ, selbstgesteuert und motiviert
- Automatisiert: Agil und iterativ
- Transparent: zuständig, verantwortlich, verfolgbar



... aber Agilität muss skalieren

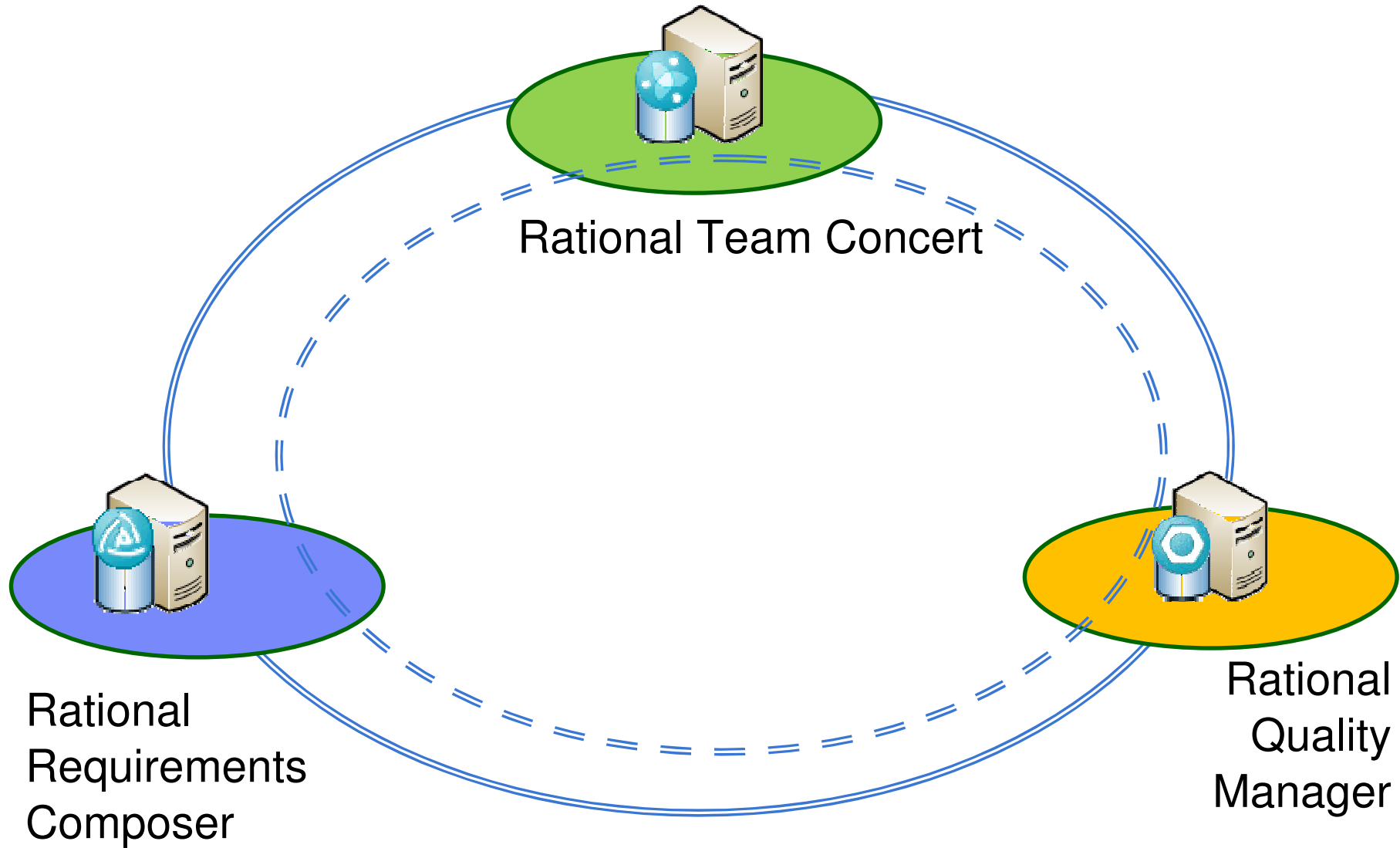


- **Agile Softwareentwicklung**

- **Werkzeuge für agile Projekte**

- **Jazz: eine Werkbank für C/ALM**

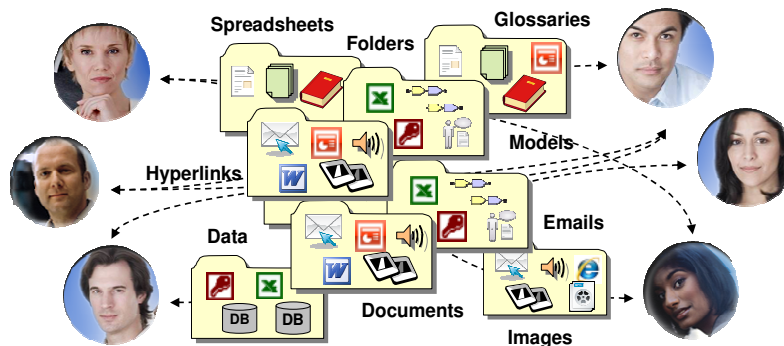
Die agile Werkbank



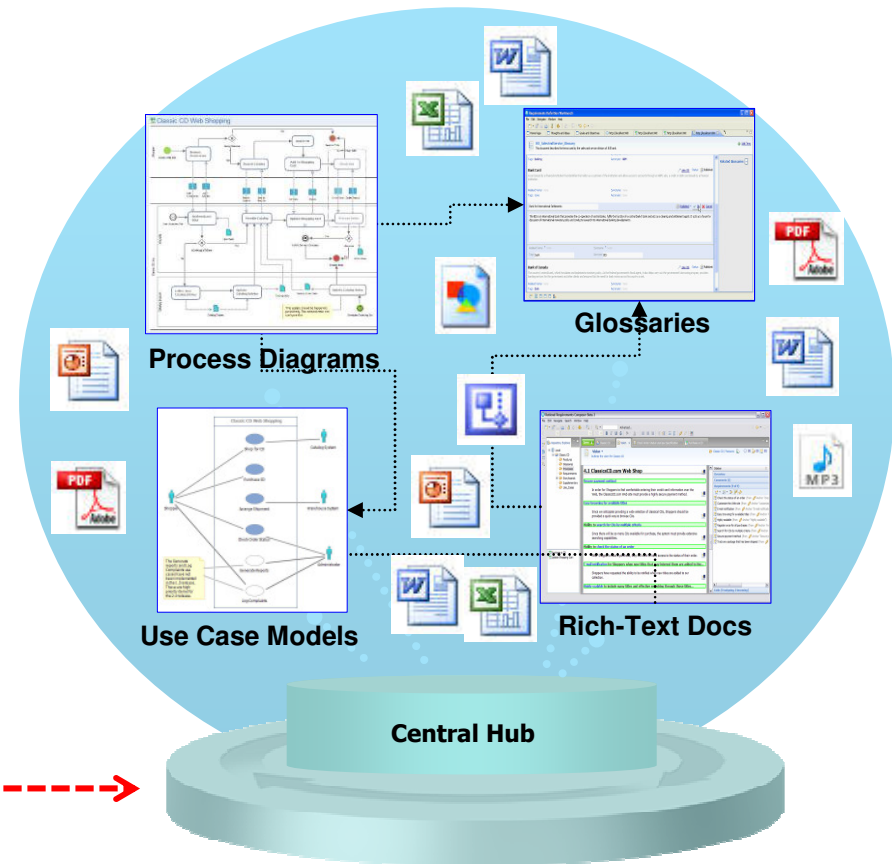
Was ist Rational Requirements Composer ?



Brücke zwischen den Informations-Inseln



- Es gibt die **unterschiedlichsten Artefakte**
- Viele Werkzeuge, Formate und Repositories bilden **Informations-Inseln**
- Eine großes **erweitertes Team** ist an der Anforderungsdefinition beteiligt



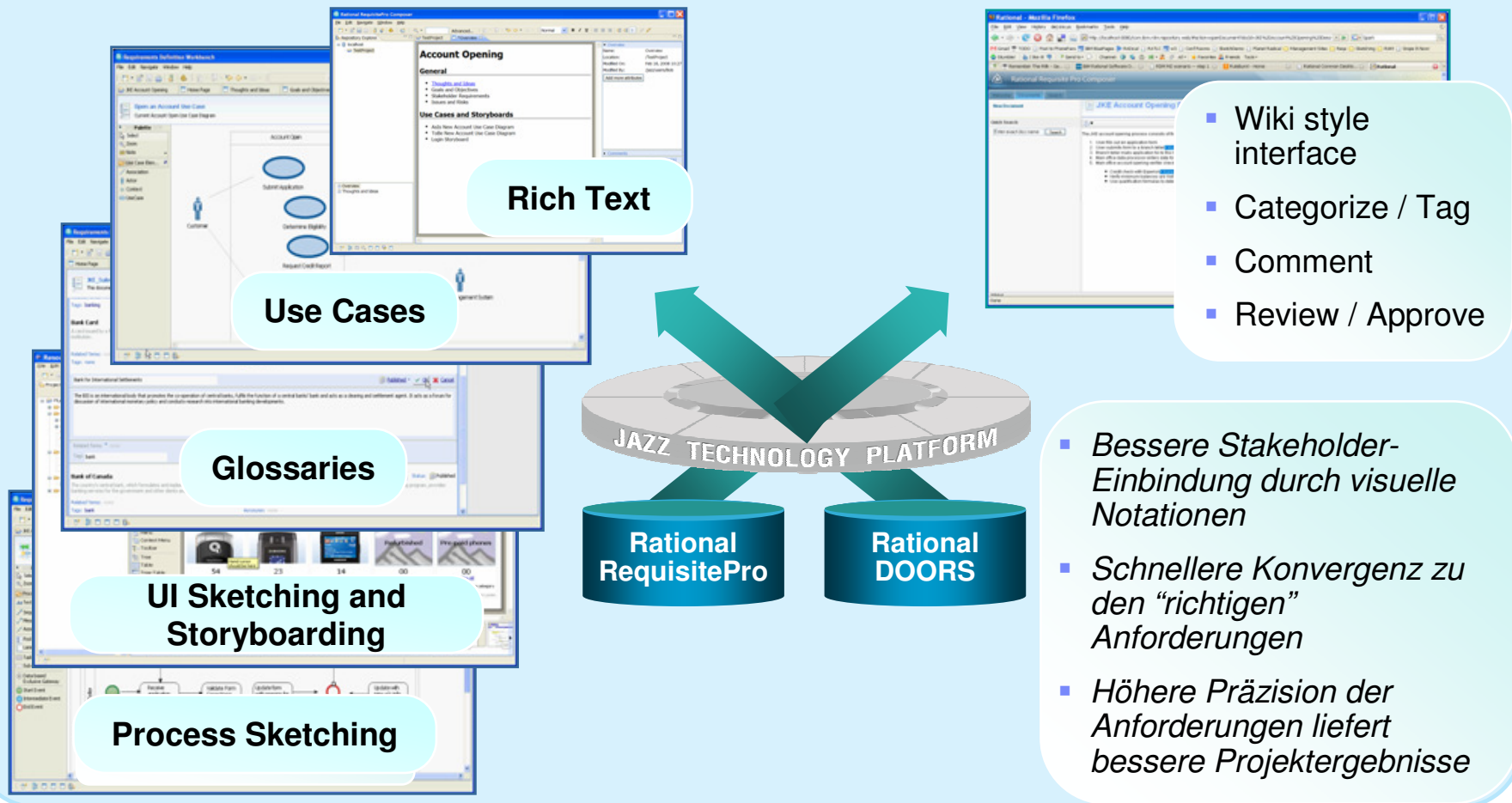
Was macht Rational Requirements Composer ?



Rational Requirements Composer

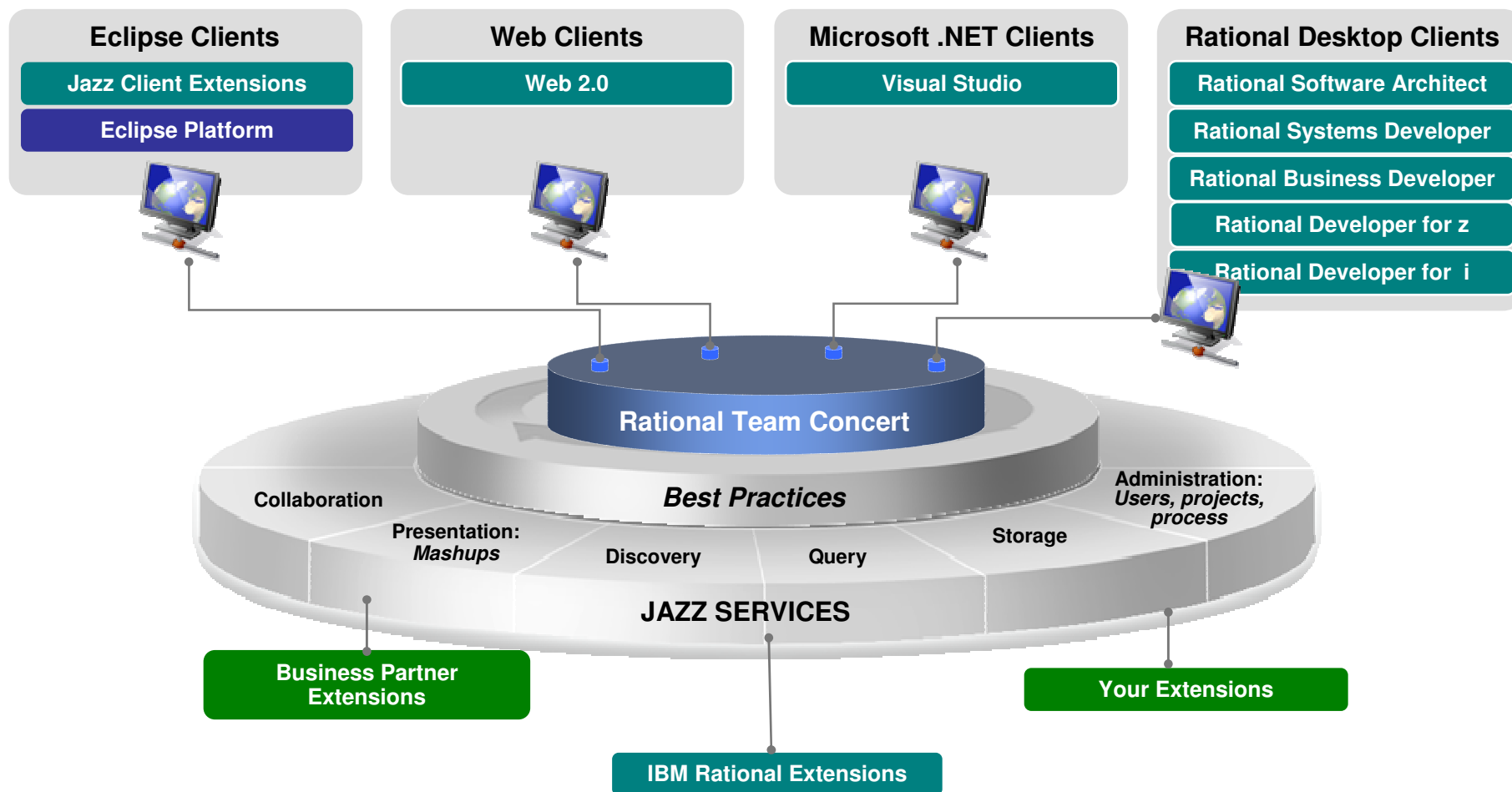
Rich Authoring Environment

Web Review and Approval



Was ist Rational Team Concert ?

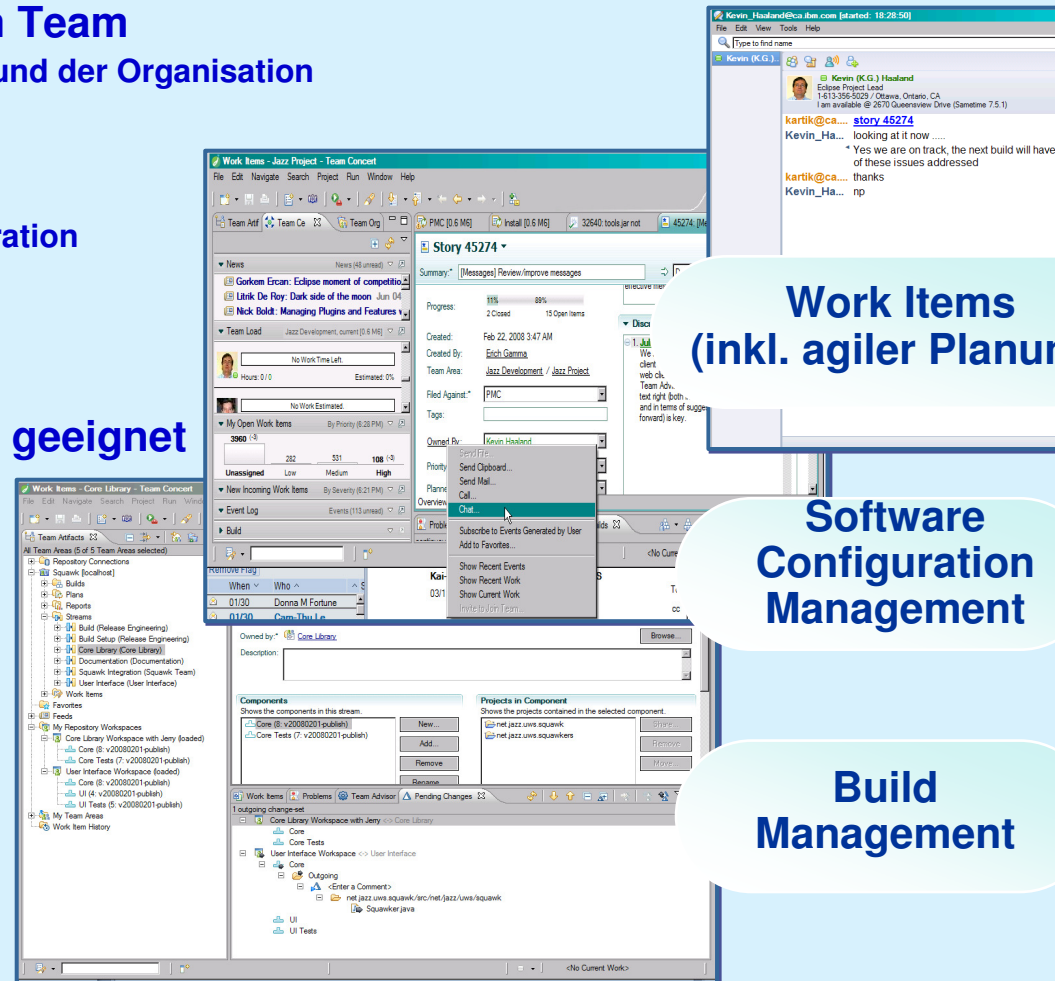
Erweiterung der IDEs



Warum Rational Team Concert ?



- **Unterstützt die Entwicklung im Team**
 - Effizienz des Individuums, des Team und der Organisation
- **Führt Projektteam zusammen:**
 - Aufgabenverwaltung und -planung
 - Versionierung und Software Konfiguration
 - Build Management
 - Reporting und Dashboards
 - Kommunikation
- **Für unterschiedliche Prozesse geeignet**
 - Agil und konventionell
 - Anpassbare Templates
- **Nachvollziehbarkeit und Transparenz**
 - Verlinkung der Artefakte des Entwicklungsprozesses
- **Plattformübergreifend:**
 - Sprachen: Java, J2EE, C, C++, C#, Cobol, PL1, EGL
 - Distributed: Linux, Unix, AIX
 - Zentral: Host (RDz Integration)
 - PC: .Net (MS Visual Studio Integration)



Work Items
(inkl. agiler Planung)

Software
Configuration
Management

Build
Management

Was kann Rational Team Concert ?



Iterationsplanung

- Integrierte Iterationsplanung
- Fortschrittskontrolle
- Vordefinierte Schablonen für agile Prozesse(SCRUM..)

Projekt-Transparenz

- Wikis
- Anpassbare Dashboards und Reports
- Aktueller Status
- Vordefinierte Schablonen und Muster

SCM

- Integriertes Konfigurationsmanagement
- Sofort einsatzbereit
- Anpassbar
- ClearCase Konnektor

Work Items

- Vordefinierte und frei definierbare Typen
- Freigabeverfahren (4-Augen-Prinzip)
- Angehängte Diskussionsbeiträge
- ClearQuest Konnektor

Build

- Nachverfolgbarkeit
- Flexible Build-Definitionen
- Lokale oder entfernte Build-Server
- Integration von Ant, Maven und Kommandozeilen-Tools
- Integration mit Build Forge

Jazz Team Server

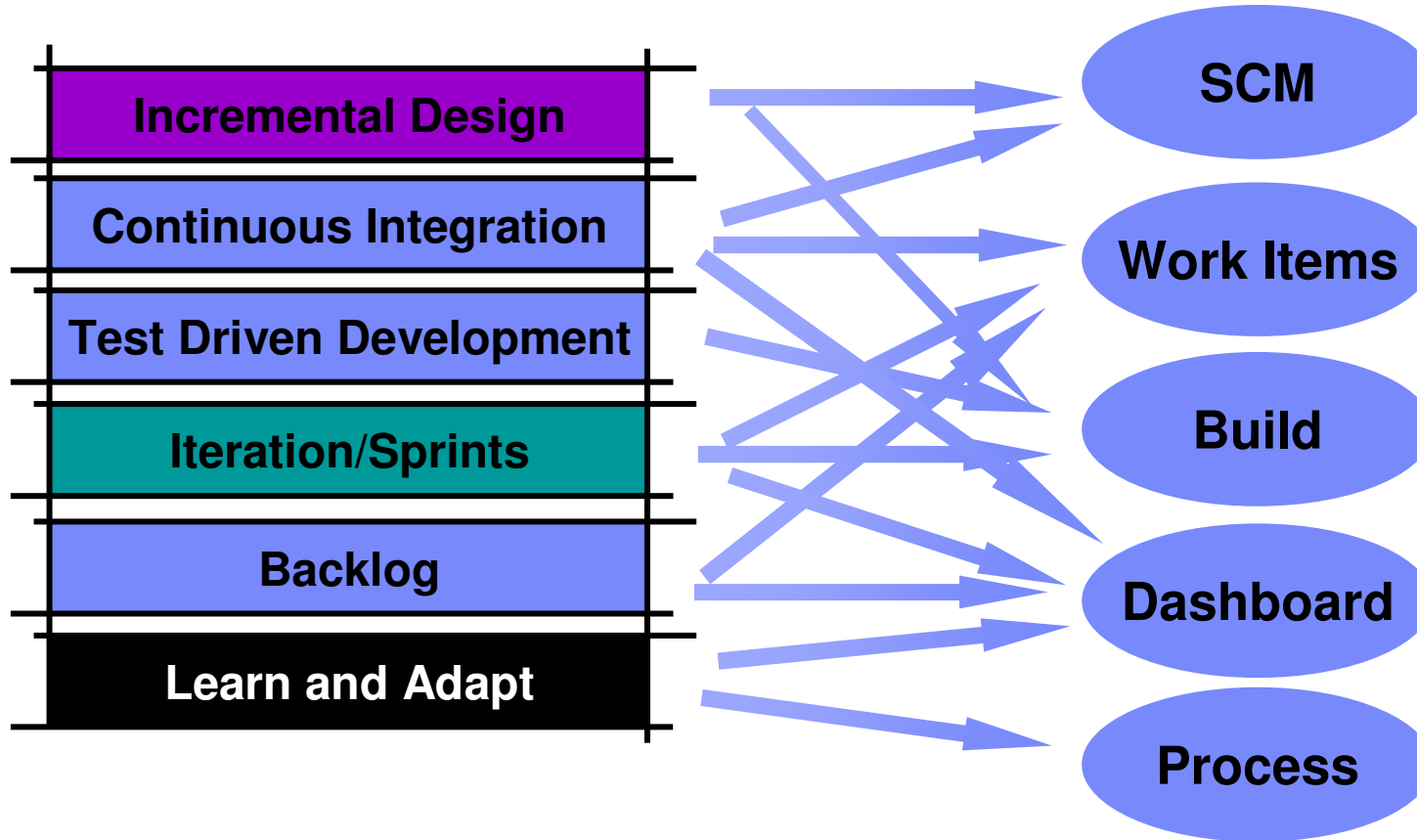
Repository für den gesamten Application Lifecycle

Teamfähigkeit

Prozessumsetzung

Zusammenarbeit im Kontext

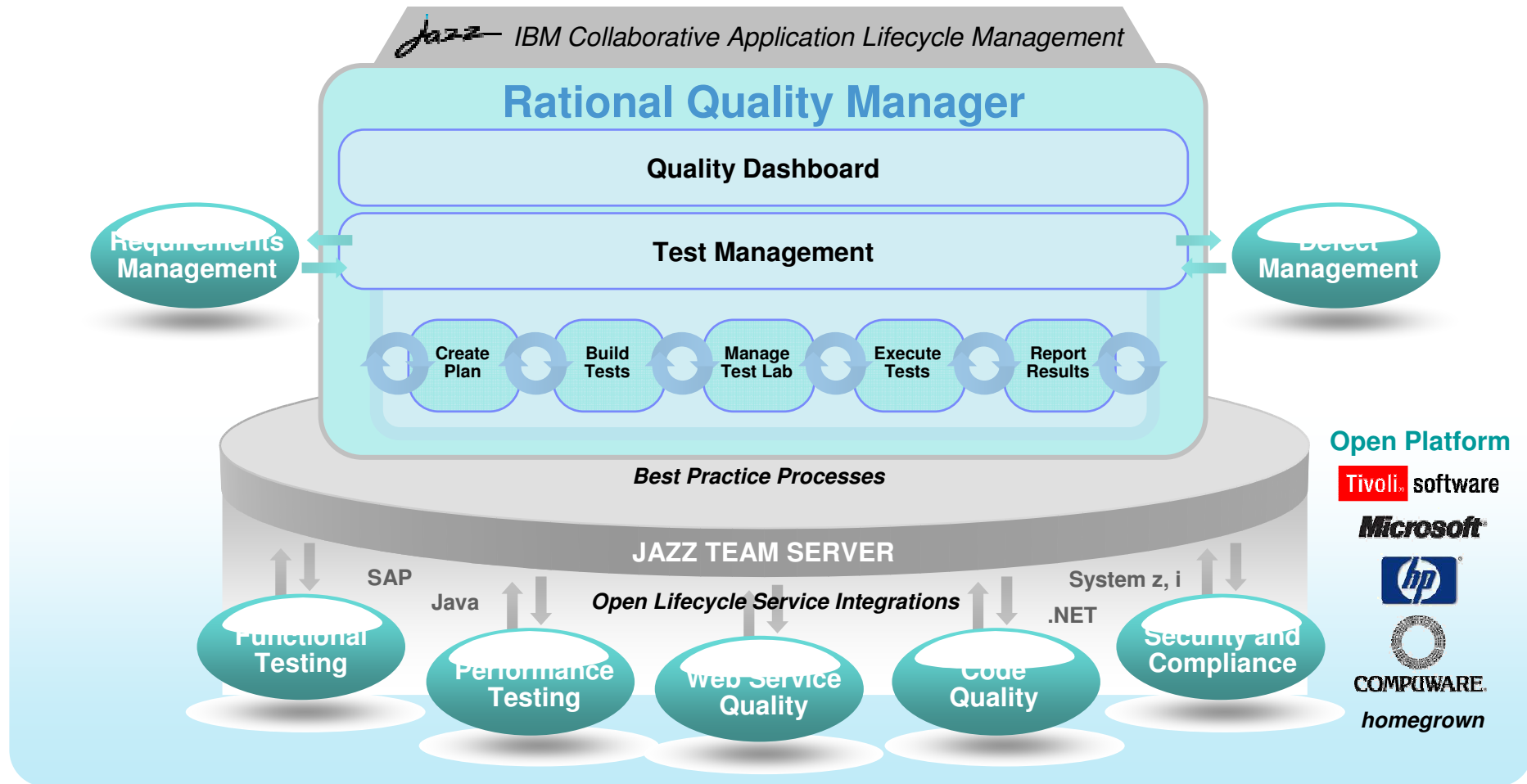
Wie agil ist Rational Team Concert ?



Was ist Rational Quality Manager ?



Zentralisiertes Testmanagement für den vollen Testzyklus für die verschiedensten Testverfahren und -umgebungen



Warum Rational Quality Manager ?



Umfassendes Qualitätsmanagement durch Integration der relevanten Daten

- ✓ Vom übergreifenden Qualitätssicherungsplan bis zu den Testscripts
- ✓ Automatisierte Abläufe ersparen Zeit und Arbeit und erhöhen die Testabdeckung
- ✓ Für Upstream und Downstream Qualitätssicherung

Integration der Testwerkzeuge und Testmaschinen

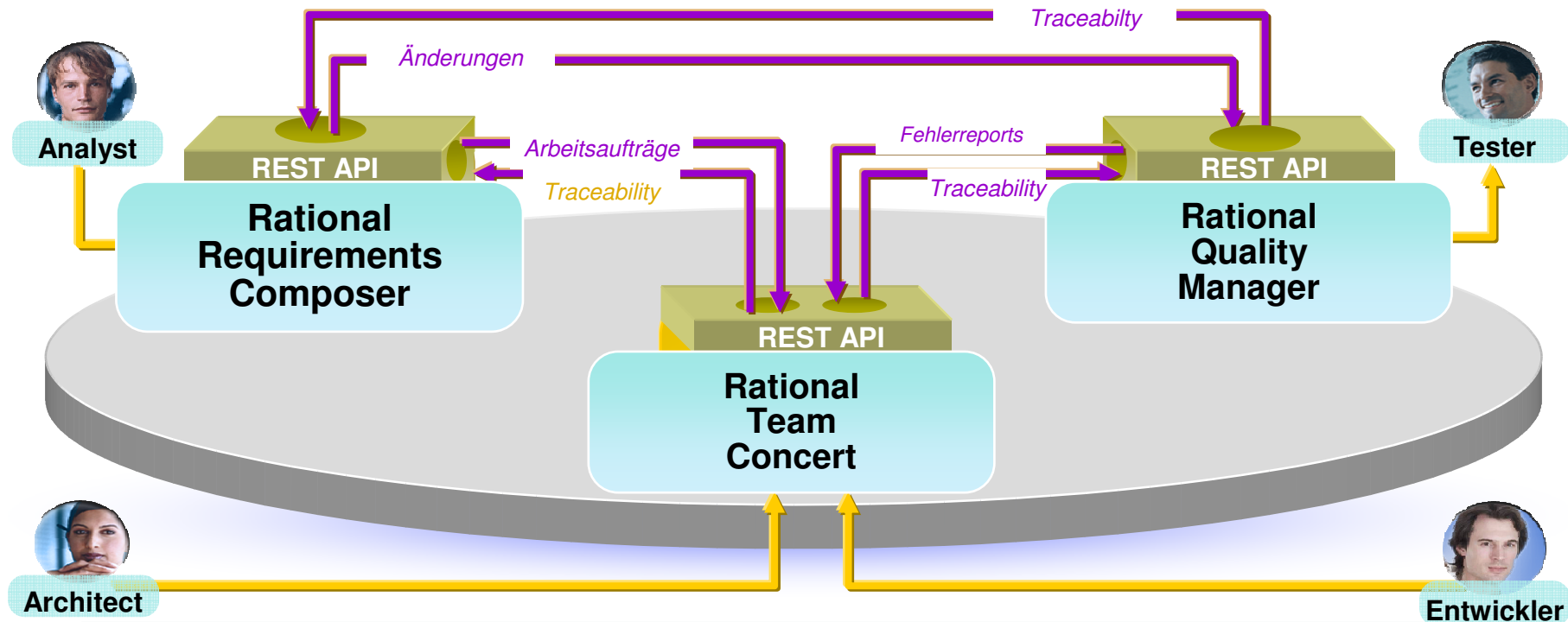
- ✓ Integriertes TestLab-Management erhöht die Auslastung der Maschinen
- ✓ Optimierte Testabdeckung für eine höhere Aussagekraft der Tests
- ✓ Integriert IBM und Nicht-IBM Testwerkzeuge für verschiedene Zielplattformen
System z, System i, SAP and .Net

Aussagefähige Reports und Dashboards

- ✓ Aktuelle Übersicht über den Qualitätsstatus im gesamten Projekt
- ✓ Reports unterstützen kontinuierliche Prozessverbesserung
- ✓ Proaktives Risk Management und Entscheidungsunterstützung



Die Integration auf der Werkbank

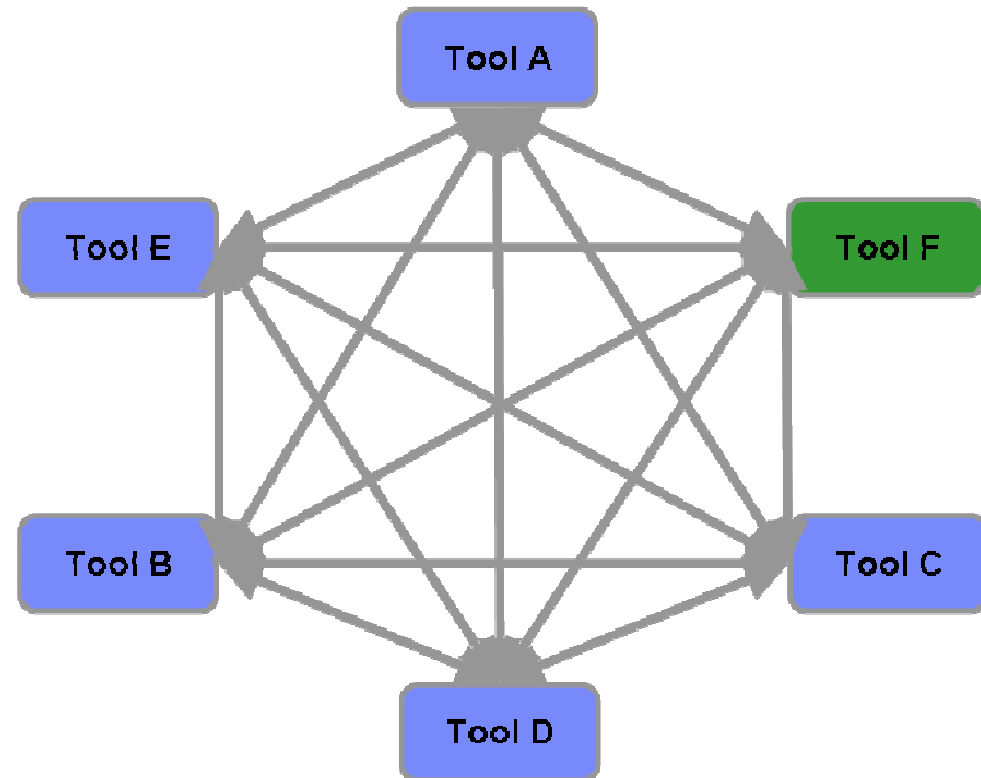


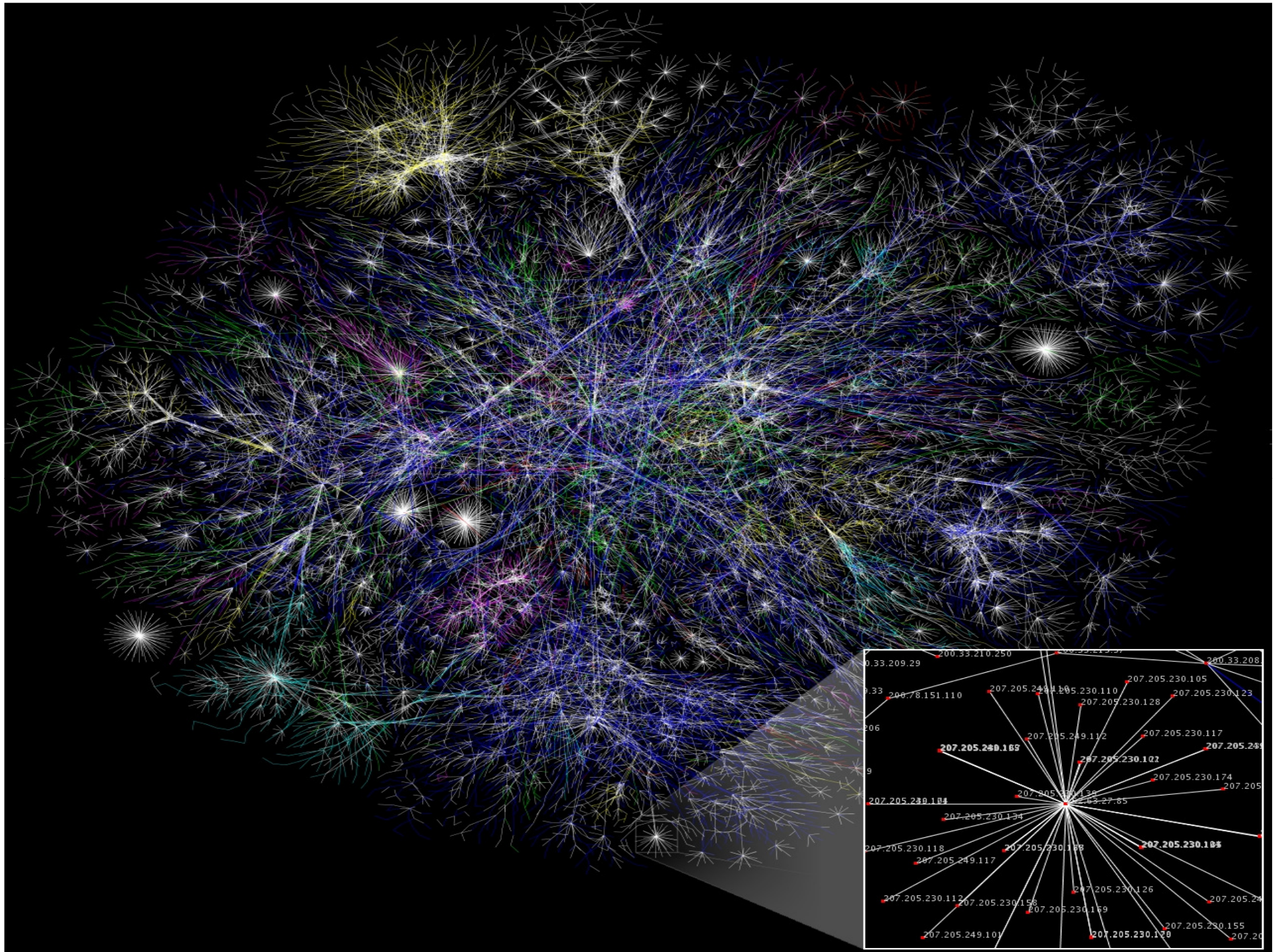
- **Koordination zwischen Business Analyst, Entwickler und Tester**
 - Enge Zusammenarbeit und transparenter Status
- **Arbeiten an einem gemeinsamen Ziel**
 - Anforderungskatalog und Iterationspläne werden synchronisiert

- **Integration von Konfiguration und Testmanagement**
 - Automatisierte Übergabe des Release in den Test
 - Eindeutige Zuordnung von Fehlern zum Release
- **Projekttransparenz mit Dashboard Mashups**
 - Aussagefähige Reports nach den Projektbedürfnissen

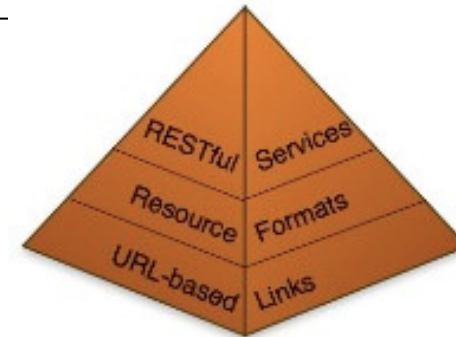
- **Agile Softwareentwicklung**
- **Werkzeuge für agile Projekte**
- **Jazz: eine Werkbank für C/ALM**

- Monolithische Systeme
 - starre Datenmodelle (Metamodelle)
 - unflexibel
 - wenig zukunftssicher
 - Hersteller-Abhängigkeit
- Föderative Systeme
 - proprietäre Schnittstellen
 - Konverter
 - quadratische Komplexität
- Komponenten-basierte Systeme
 - enge Kopplung
 - hohe Versionsabhängigkeit
- ???

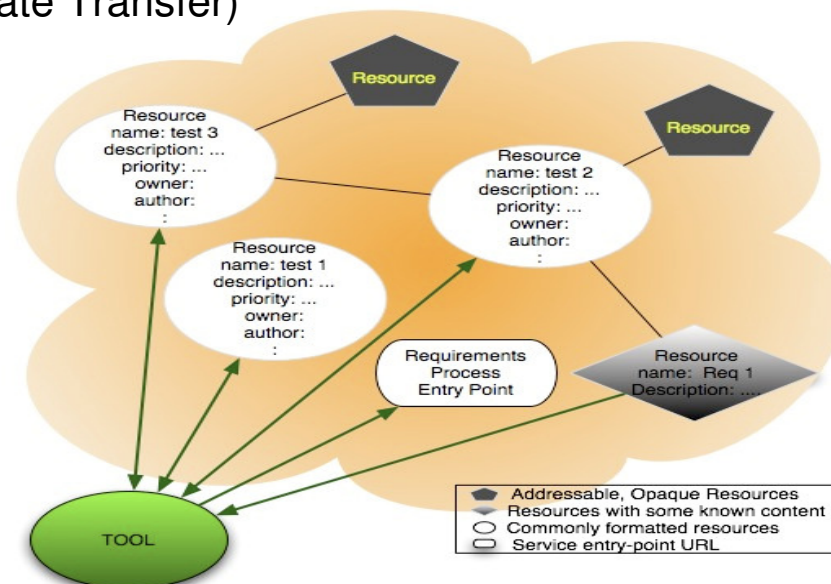




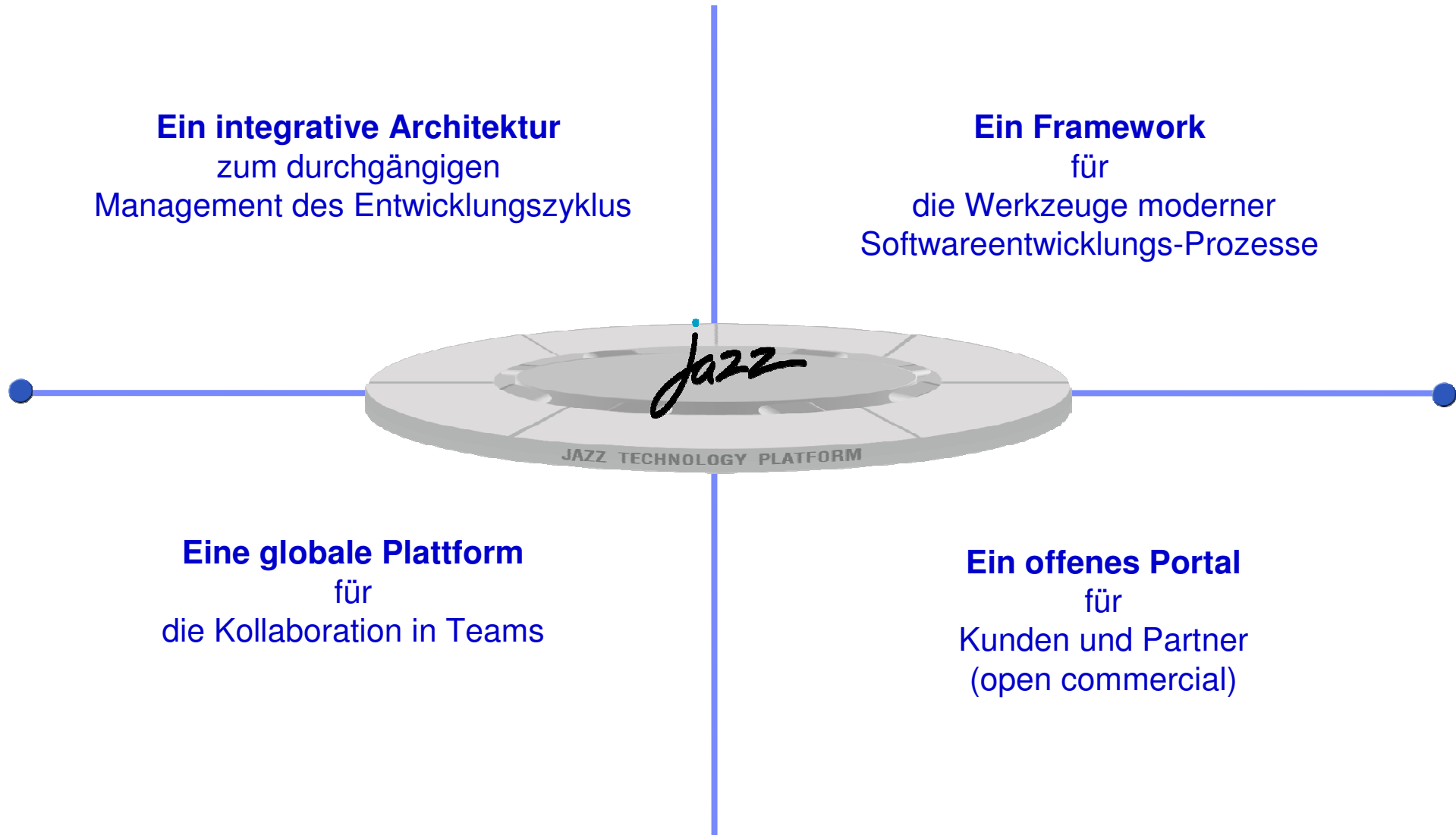
Was wäre wenn...



- Jede Ressource (Artefakt) erhält eine URL
 - Vernetzung als eingebettete Links (URLs)
- Ressourcen haben externe Representationen
 - Representationen sind werkzeugunabhängig als Standards definiert
- Lose Kopplung über gemeinsame RESTful Services (Representational State Transfer)



Jazz: Fundament für eine neue Generation von integrierten Werkzeugen

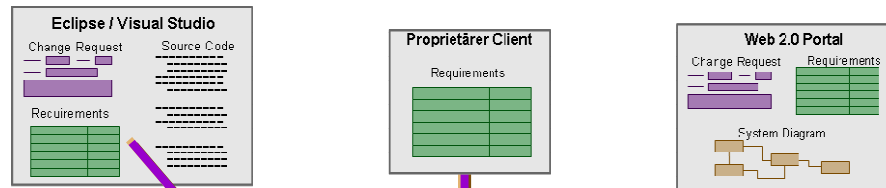


Jazz Kollaborations-Architektur auf Basis offener Services



Skalierbar Erweiterbar Verteilt Offen

Unterschiedliche Frontends



Lose Kopplung über RESTfull Services

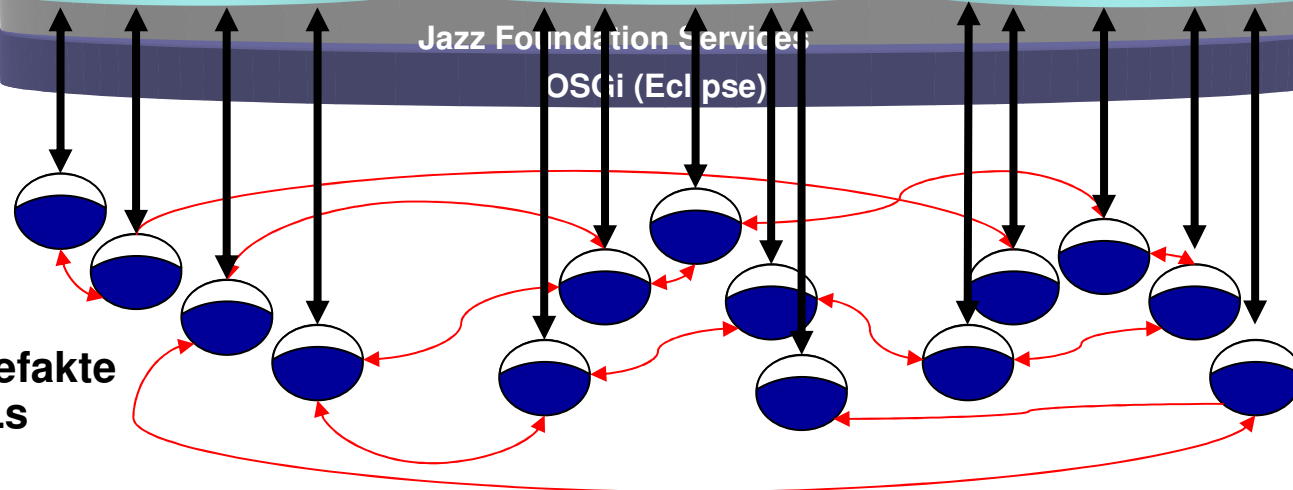


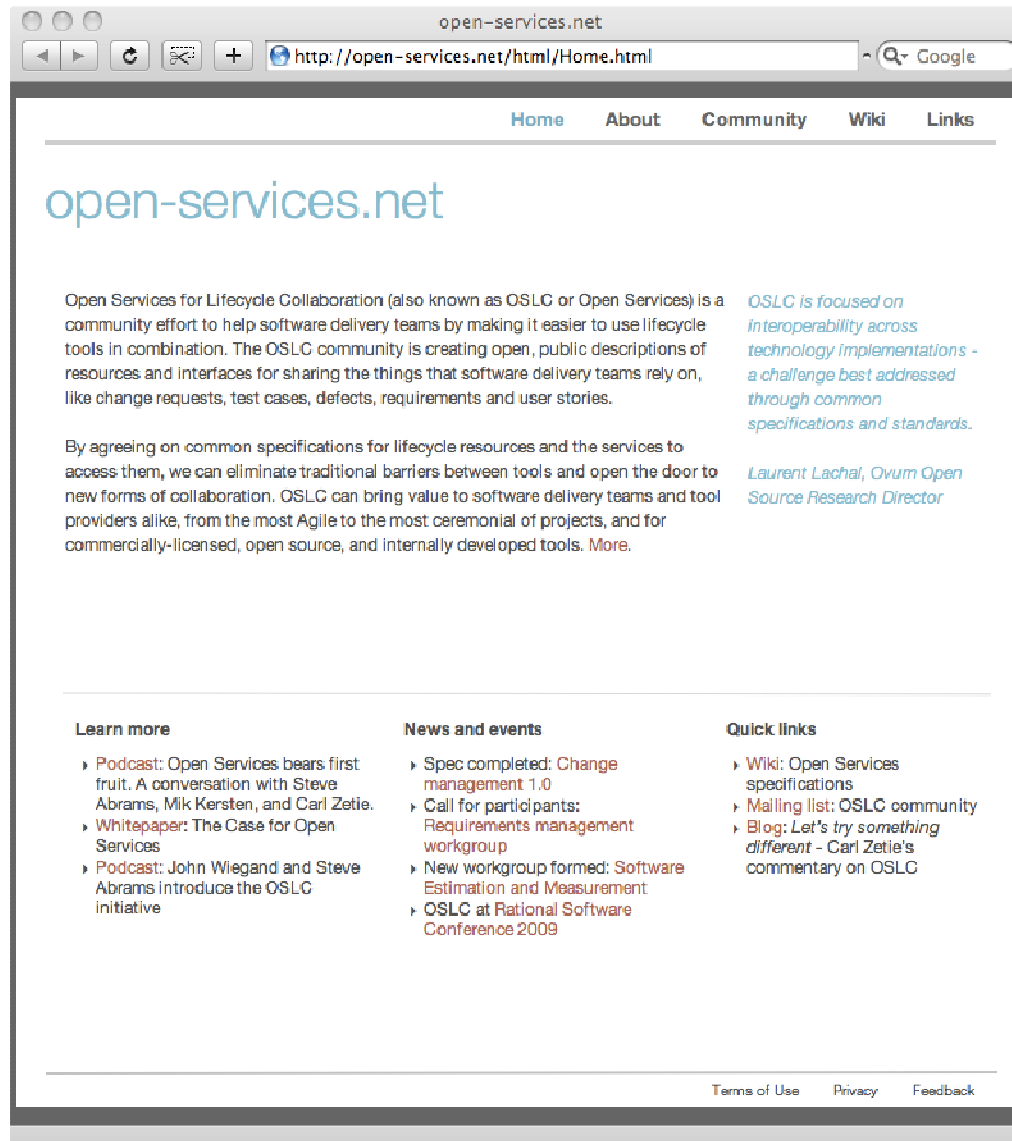
Standardisierung



Kapselung/ Plugins

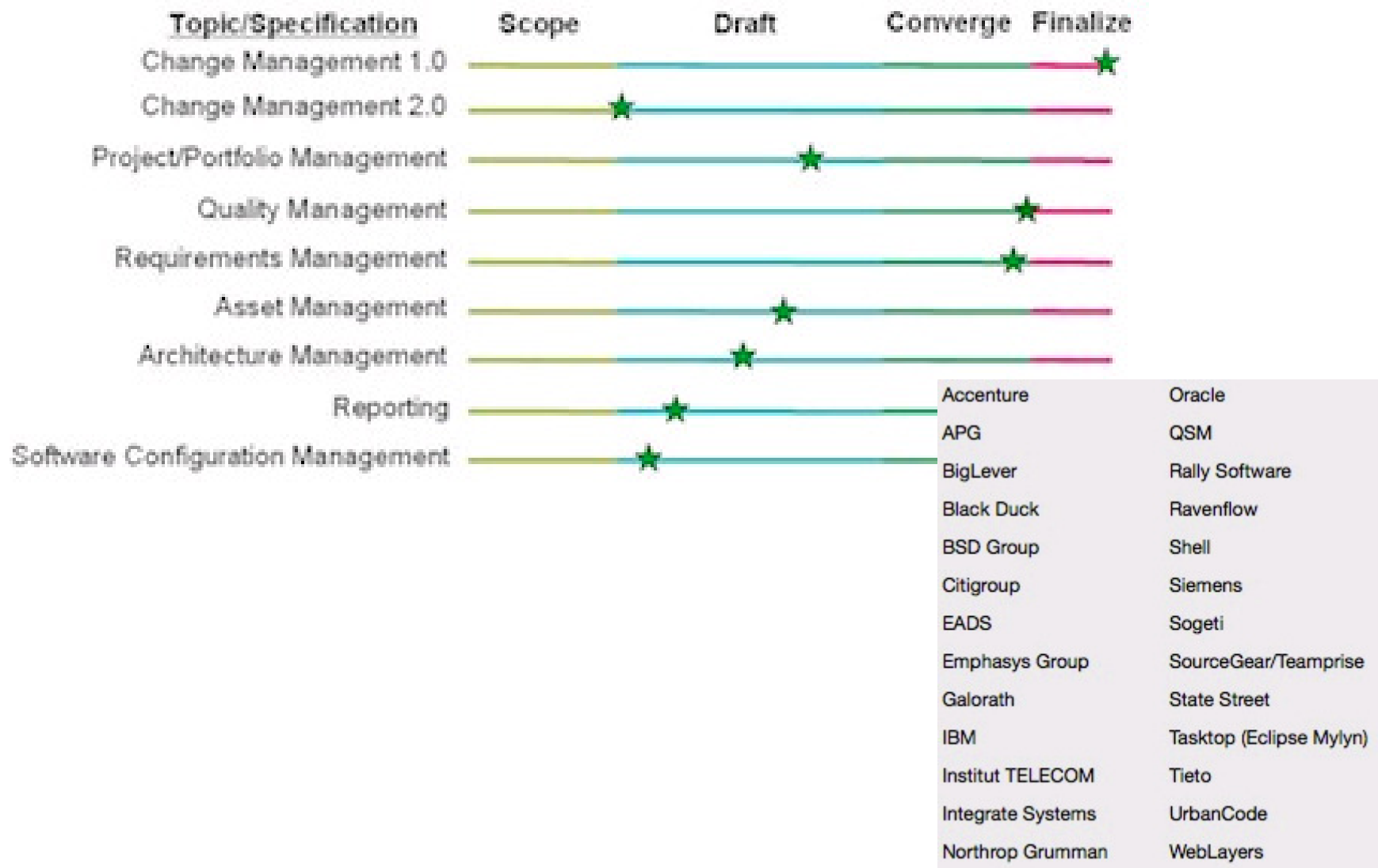
Vernetzung aller Artefakte über eindeutige URLs



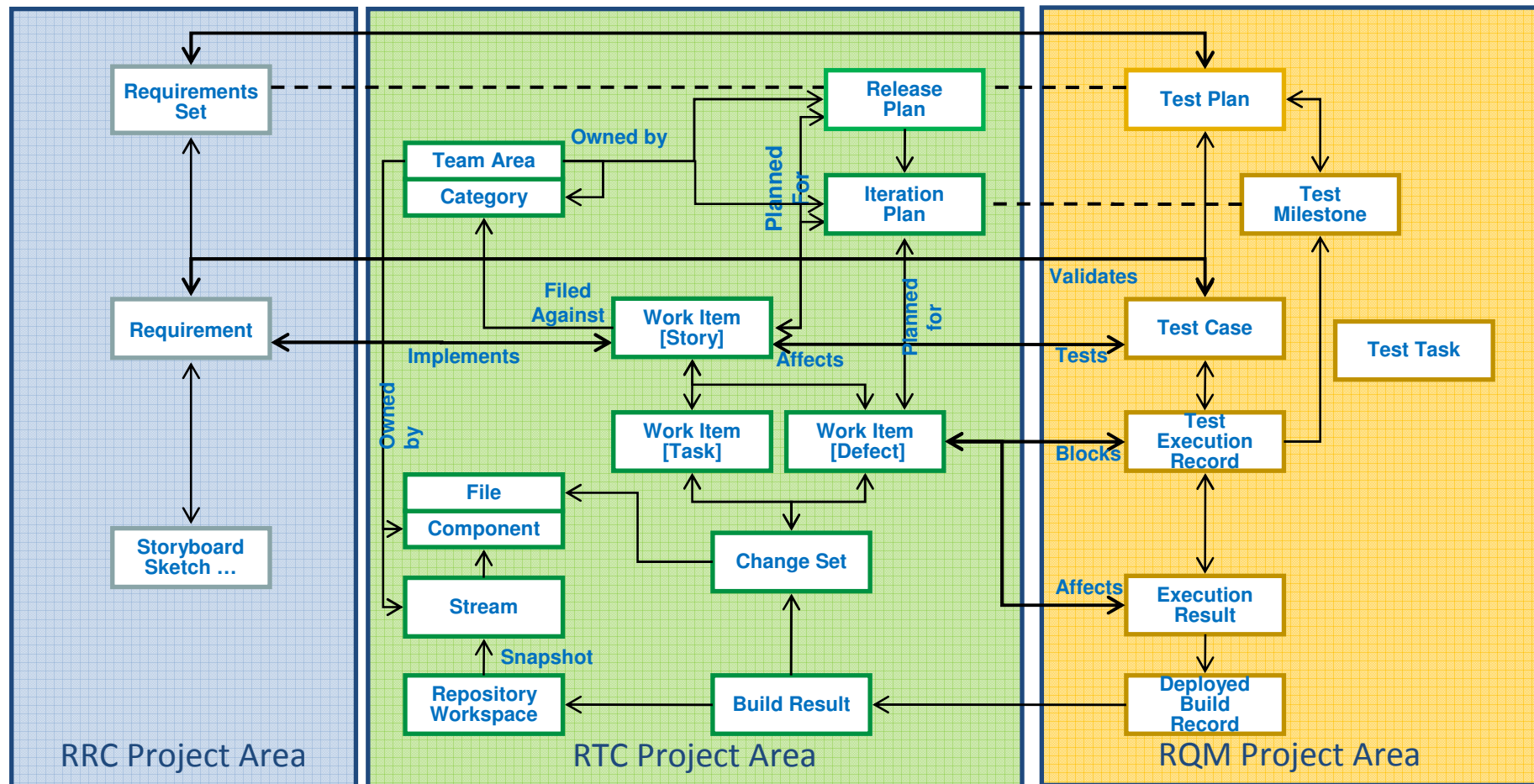


- ▶ Eine offene Community
- ▶ Standards zur ALM Integration von Werkzeugen und Daten
- ▶ Technologie-neutral
- ▶ Creative Commons Licensing
- ▶ Zugesichert patentfrei

Die OSLC Initiative



Das Web der Software-Delivery

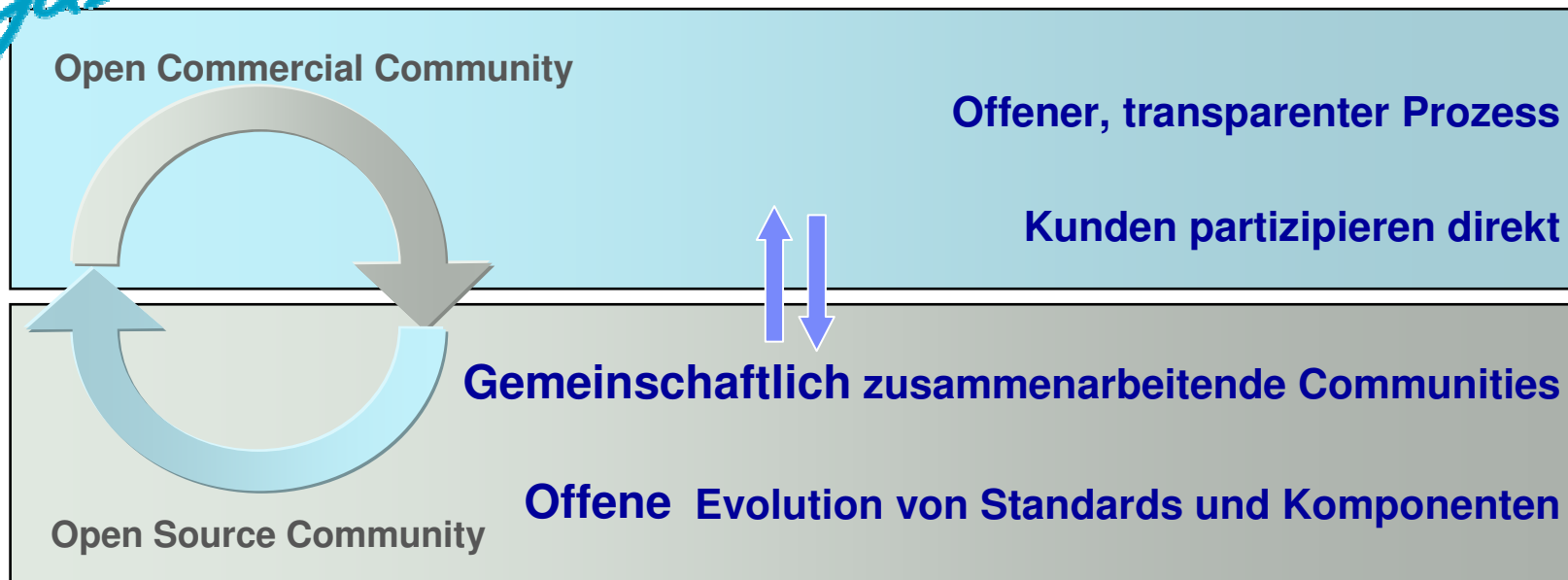


See <https://jazz.net/wiki/bin/view/Main/CALMLinkTypes> for details

Jazz steht auch für Offenheit bei der Produktentwicklung



jazz



Und jetzt in der Demo:

