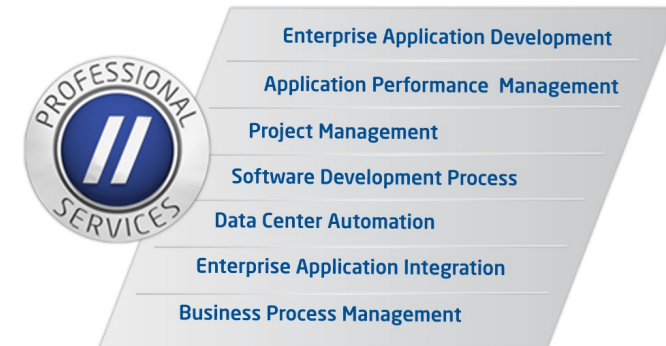


Continuous Integration & Testing meets BPM meets Reality

Stuttgarter Test-Tage am 21. & 22. März 2013

Jonas Grundler, Managing Consultant, NovaTec GmbH
Jonas.Grundler@novatec-gmbh.de



- // 2002 - 2006: Berater - Business Process Management, IBM
- // 2006 - 2011: Entwickler und Architekt, WebSphere Process Server
- // seit 2011: Managing Consultant bei der NovaTec GmbH
 - // Business Process Management
 - // BPM - unabhängige Beratung, Reviews
 - // Application Performance Management



Einführung



Was ist Continuous Integration (CI)



Was ist Geschäftsprozessmanagement (BPM)



BPM-Anwendungen Testen



CI mit BPM, Jenkins und Sonar



Continuous Integration mit...



IBM BPM (kommerziell)



Activiti (open-source)



Zusammenfassung

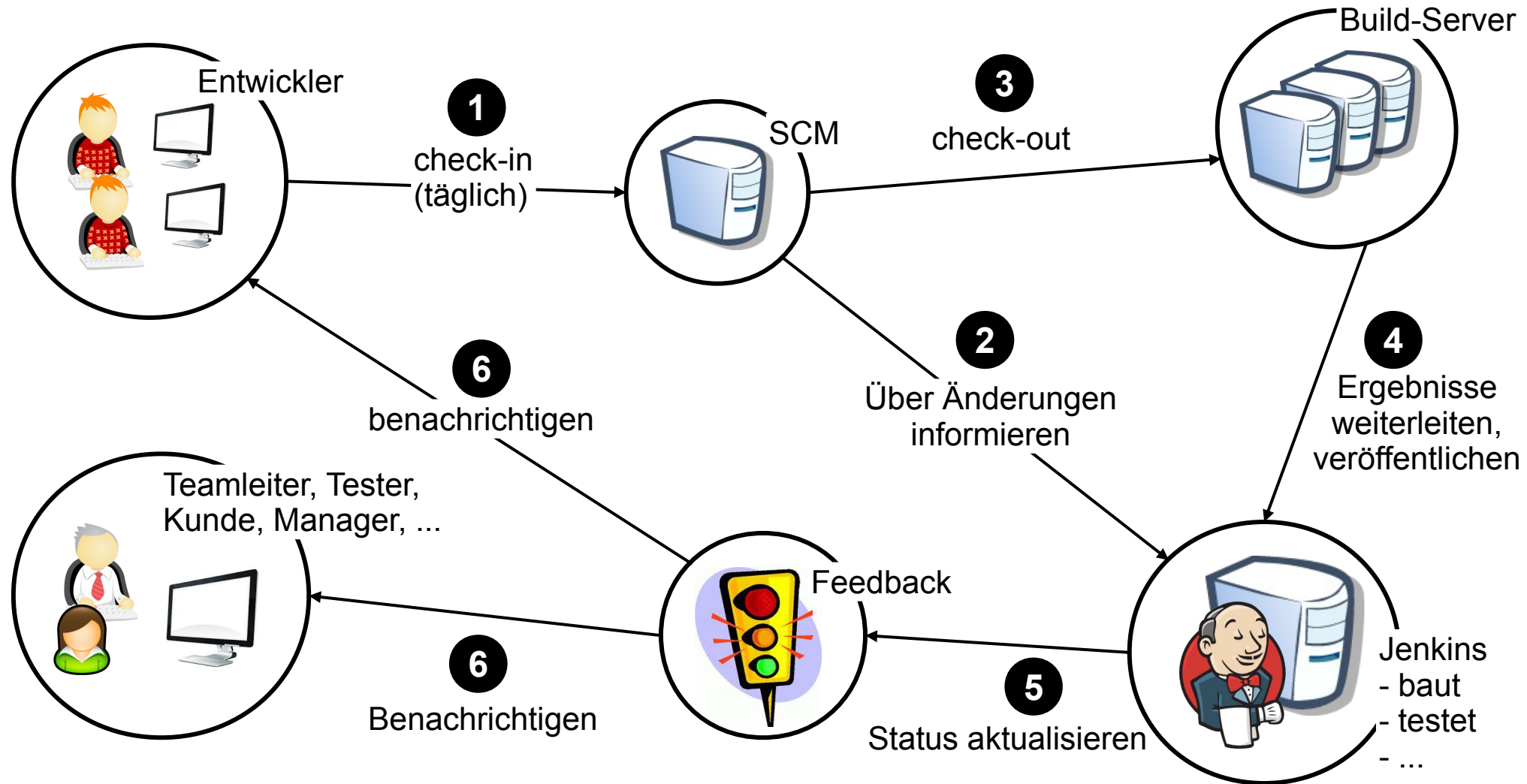
„Continuous Integration“

(Kontinuierliche Integration)

... was ist das?

Continuous Integration (CI)

Übersicht, Komponenten, Beteiligte



„Continuous Integration“

- Martin Fowler -

[...]

each person integrates at least daily

[...]

Each integration is verified by an automated build (including test)

[...]

detect integration errors as quickly as possible.

[...]

„Continuous Integration“

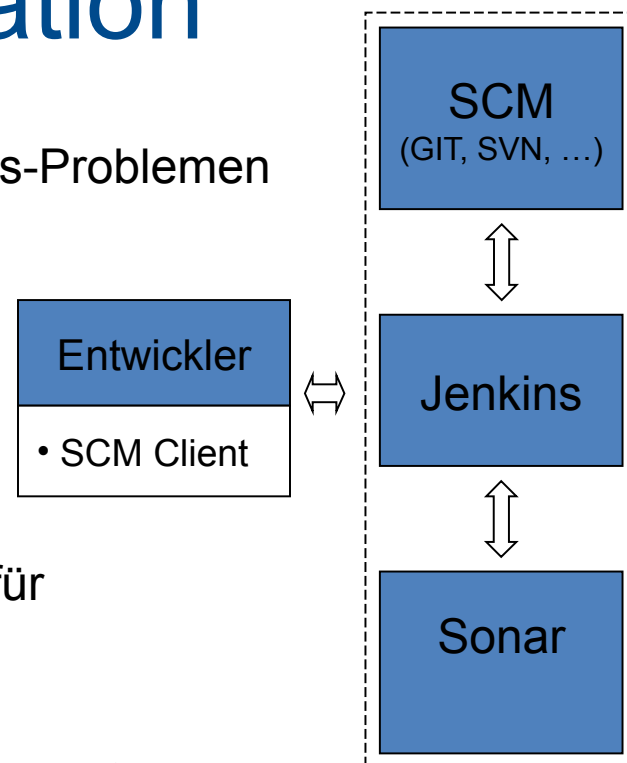
// Kontinuierliches Erkennen und Beheben von Integrations-Problemen

// Früherkennung problematischer Teile der Anwendung

// **Unmittelbares Testen** zeigt fehlerhaften Code

// Ständige Verfügbarkeit von ausführbarem Code, bspw. für Demos, weitere Tests, Marketing, etc.

// Dem Entwickler wird durch Automatisierung viel Arbeit erspart



„Continuous Integration“

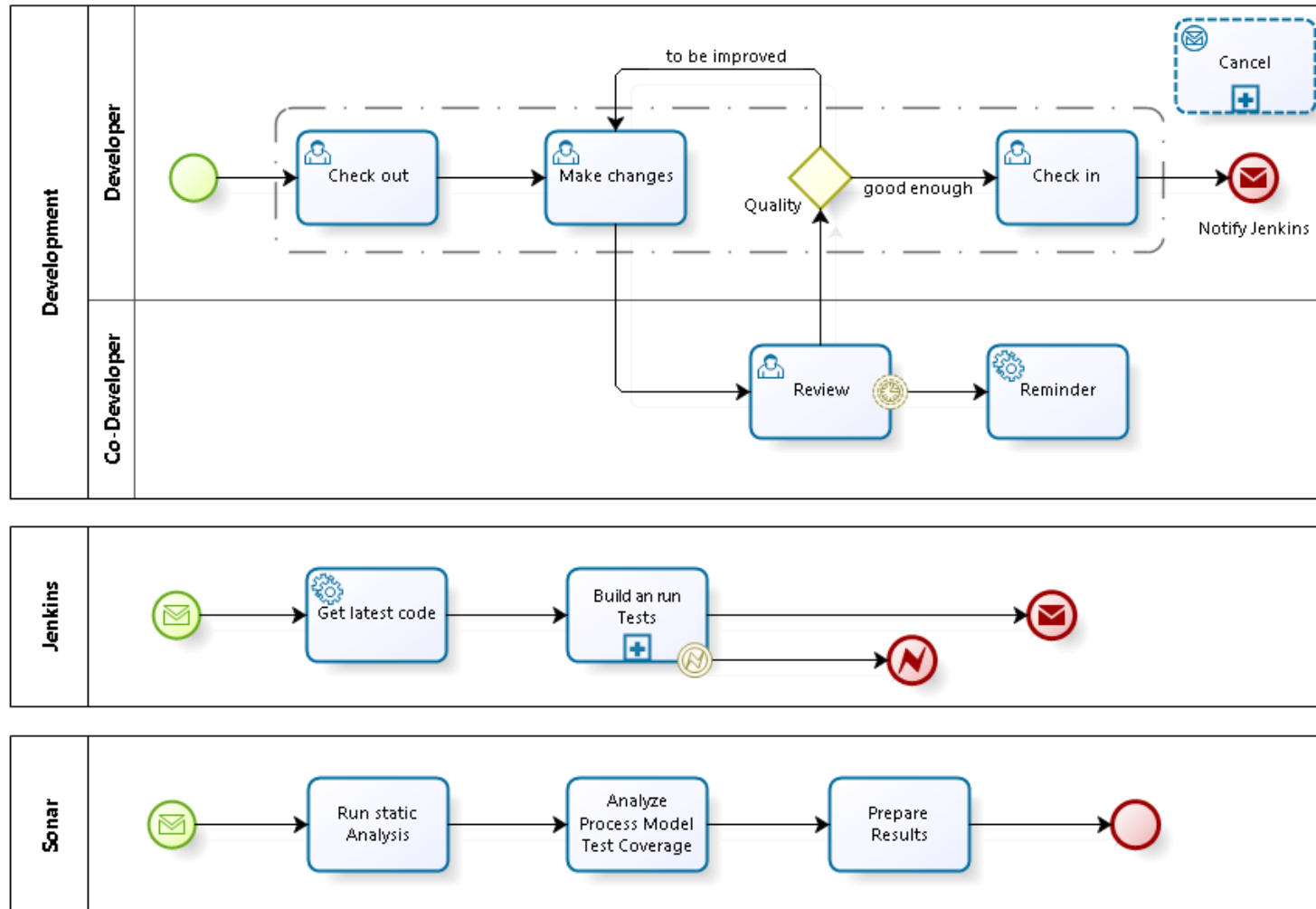
... alles in allem bekannt und verbreitet!

*... warum nicht einfach auch in Business
Process Management Projekten anwenden?*

- // Mit BPM lassen sich Abläufe in Unternehmen mit IT-Mitteln abbilden. Daraus ergeben sich enorme Möglichkeiten
- // **Graphische Beschreibung von Prozessen mittels standardisierter Sprachen, bspw. BPMN 2.0**
- // Monitoring - funktioniert mein Unternehmen, oder meine Abteilung, so wie ich es mir vorstelle?
- // Monitoring - wie lange dauert es, bis über einen Kreditantrag entschieden ist (KPIs)? Welcher Schritt sollte als erstes effizienter gestaltet werden?
- // Simulationen - was wäre wenn?
- // ...
- // Die Einführung von BPM erfordert breite Akzeptanz in Unternehmen: Fachabteilungen und IT müssen miteinander kommunizieren.
- // Weitere an BPM angrenzende Technologien sind bspw. das Case Management (unstrukturierte "Prozesse") und Business Rules Management (BRM - Geschäftsregeln).

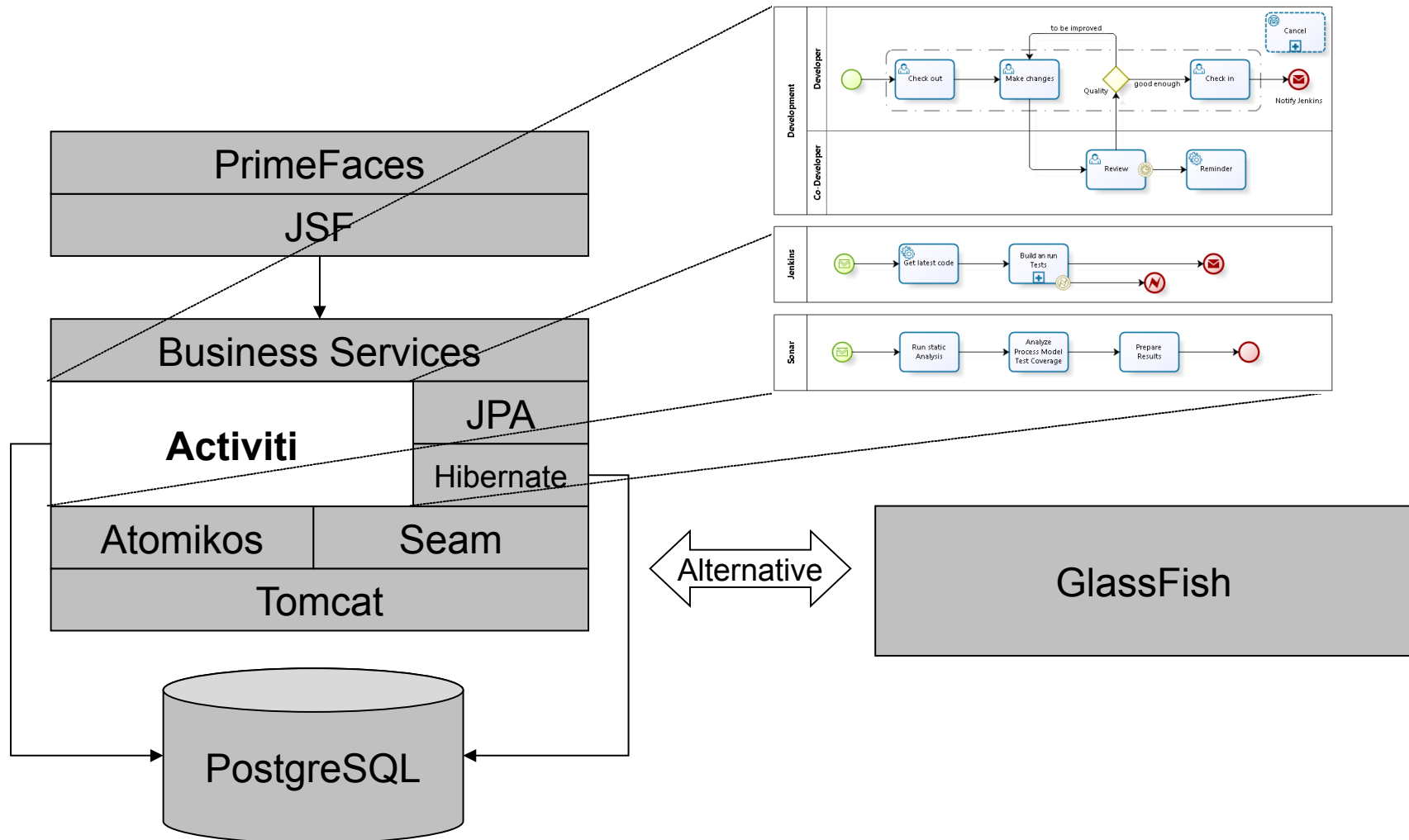
Business Process Management (BPM)

Beispielprozess in BPMN 2.0



Business Process Management (BPM)

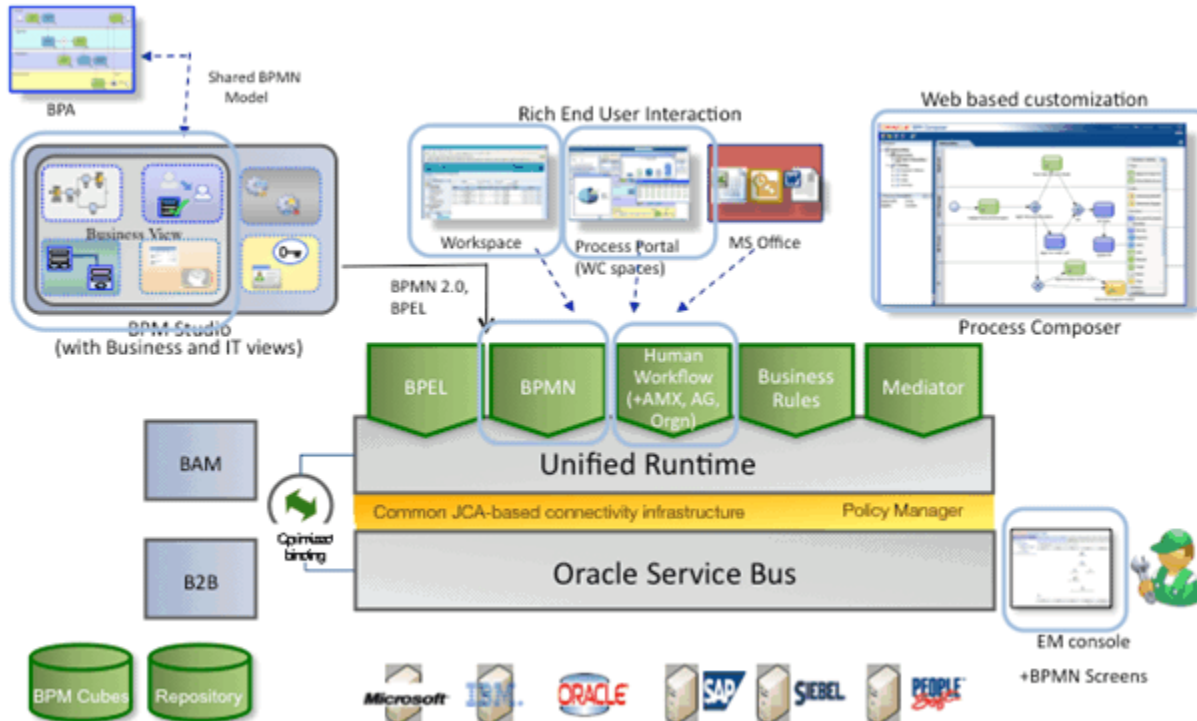
Beispielstack mit Activiti



Business Process Management (BPM)

Beispielproduktpalette - Oracle

Stuttgarter Test-Tage 2013



Product Strategy Business Process Management



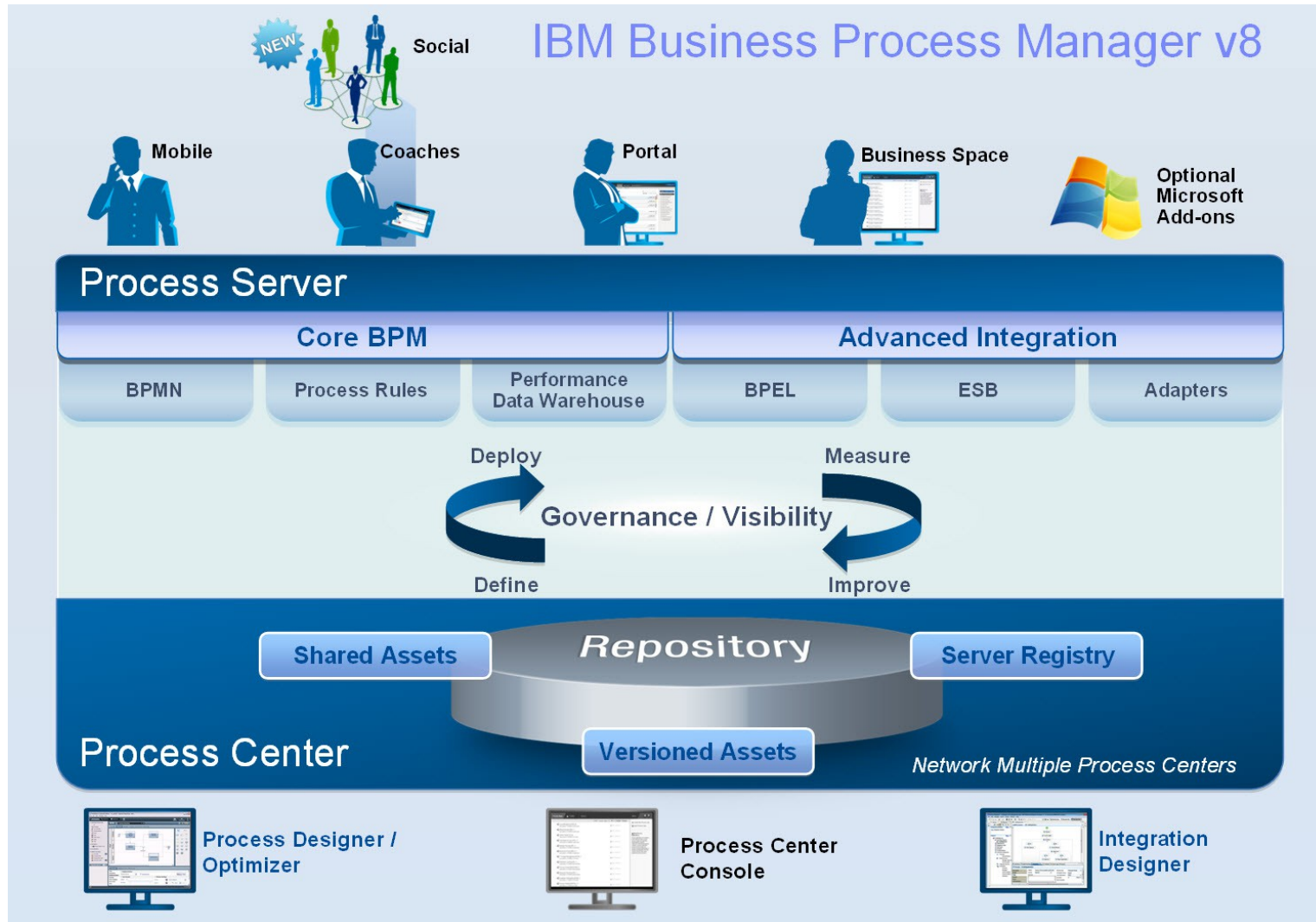
ORACLE

Siehe http://docs.oracle.com/cd/E21764_01/core.1111/e10106/ha_soa.htm

Business Process Management (BPM)

Beispielproduktpalette - IBM

Stuttgarter Test-Tage 2013



„Continuous Integration“

... warum nicht einfach auch in Business Process Management Projekten anwenden?

(...dort wo bspw. Testen eigentlich wichtiger ist denn je...)

„Continuous Integration“

Insgesamt wird der Build und das Deployment auch mit BPM hinzubekommen sein.

Die meisten Vendoren bieten Skripte für Automatisierung an.

„Continuous Integration“

*Gedanken aber mache ich mir noch übers
Testen meiner Prozesse. Wie kann das
angegangen werden?*



Einführung



Was ist Continuous Integration (CI)



Was ist Geschäftsprozessmanagement (BPM)



BPM-Anwendungen Testen



CI mit BPM, Jenkins und Sonar



Continuous Integration mit...



IBM BPM (kommerziell)



Activiti (open-source)



Zusammenfassung

Warum CI nicht *einfach* so auf BPM angewendet werden kann - Oder: Was sind die Unterschiede?

- // Eine BPM-Anwendung braucht typischerweise zusätzliche Tools. Beispielsweise den Prozess-Designer.
 - // Die Integration dieser Tools in eine CI-Umgebung ist so oftmals nicht vorgesehen (erfordert also ein wenig Forschung falls die Integration das erste mal stattfindet...)
- // Testfälle von Java-Anwendungen laufen meistens innerhalb weniger Sekunden durch und sind außerdem typischerweise synchron. BPM-Tests nicht.
 - // Weder laufen Tests innerhalb weniger Sekunden durch, noch sind Tests synchron
 - // Prozesse müssen von verschiedenen Rollen (Manager, Berater, Chef, ...) bearbeitet werden
 - // Prozesse warten u.U. Tage oder Wochen auf ein bestimmtes Ereignis - bspw. dem Eintreffen von Post

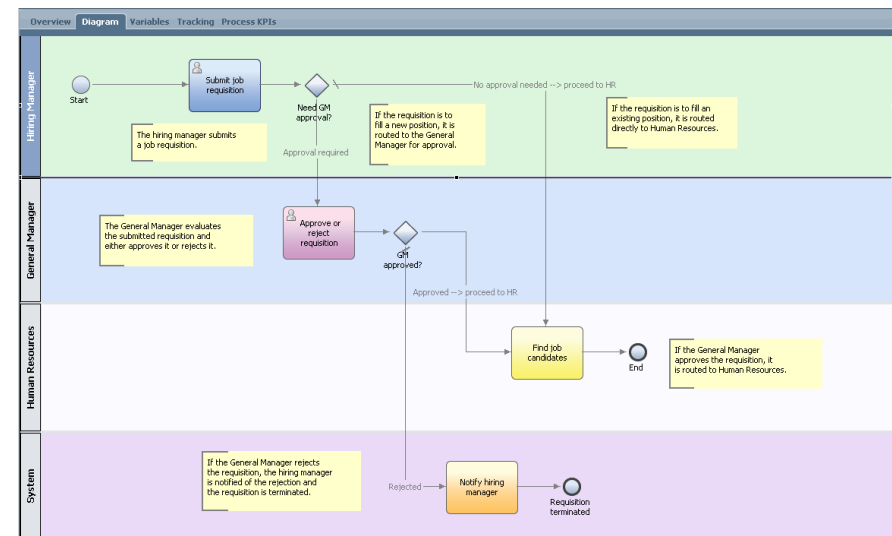
Warum CI nicht *einfach* so auf BPM angewendet werden kann. - Oder: Was sind die Unterschiede?

- // Testfälle in Java können oft sehr einfach als erfolgreich oder nicht erfolgreich bewertet werden. Prozesse sind da komplizierter...
- // Wurde der richtige Pfad im Prozessmodell genommen?
- // Ist der "Rückgabewert" der richtige?
 - > Prozesse haben meistens keinen Rückgabewert
- // Fehlerbehandlung in Java ist meistens eine Exception, die überprüft werden kann
 - > Prozesse können als Fehlerbehandlung einen weiteren Pfad navigieren. Wie wird das getestet?

- // Testen ist aufwendig. Erst recht mit BPM-Anwendungen.
- // Prozesse sind asynchron - das ist immer eine Herausforderung: ein Ergebnis steht nicht unmittelbar zur Verfügung, die Tests brauchen signifikant länger (als Java-Tests)
- // Timer gibt es an verschiedenen Stellen für: Eskalationen, Warten bis eine Nachricht eintrifft, ...
- // Prozesse werden von verschiedenen Rollen bedient (Sachbearbeiter, Manager, Kunde, ...)
- // Key Performance Indicators - KPIs - wollen auch getestet werden
- // Tests brauchen oft aufwendige Setups und Vorbereitungen.

// Je später Probleme gefunden werden, desto höher sind die Kosten zur Fehlerbehebung.

- // Versionierung ist komplex und gerade bei Prozessen mit erheblichem Aufwand verbunden.
- // Die umgebende Software (Prozesse arbeiten nie allein) ist häufig eng mit den Prozessen verzahnt.



IBM BPM 8.0 - Process Designer

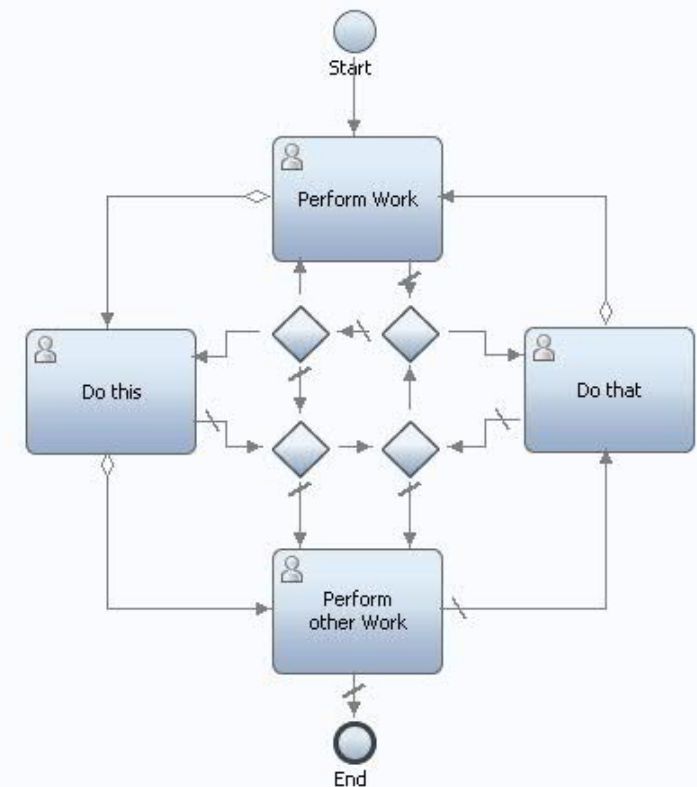
In BPM Projekten oft vernachlässigt: White-Box-Tests, Überprüfung des Prozessmodells

// White-Box-Tests

- // Wurden in den Tests alle Pfade durchlaufen? ("Model"-Coverage)
- // Macht der Prozess was ich möchte? (Wurde die E-Mail ein 2. mal verschickt, nachdem das erste Senden zu einem Fehler geführt hat?)
- // Wurden die Pfade zur Fehlerbehandlung durchlaufen?

// Prozessmodell

- // Komplexität & Verständlichkeit
- // Gibt es Patterns im Prozessmodell die
- nachweislich (?) - ungünstig sind (und durch andere
Patterns ersetzt werden können?)
- // ...



BPM-Anwendungen testen

Komplexität beherrschen mit: Metriken

- // Komplexität - einfach ausgedrückt - ist Teil des Werts eines Prozessmodells.
- // Die Verständlichkeit eines Prozessmodells, und damit auch die Wartbarkeit, hängt auch von dessen Komplexität ab.
- // Metriken helfen, die Komplexität und die Verständlichkeit zu bestimmen.

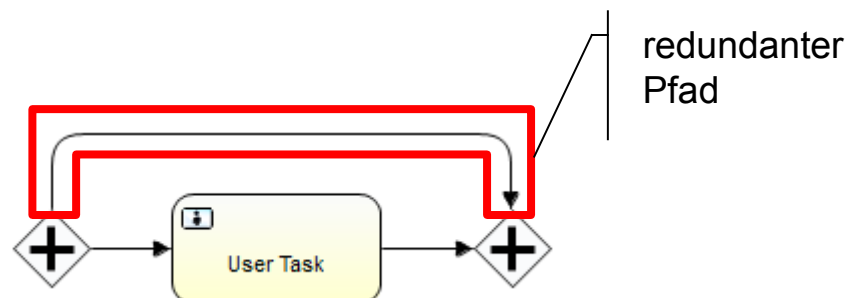
- // Eine empirische Studie hat untersucht, welche Metriken sich am besten eignen.
Zum Beispiel:
 - // Anzahl Links
 - // Anzahl Aktivitäten
 - // Anzahl Verzweigungen
 - // ...

¹⁾ Prediction Models for BPMN Usability and Maintainability (2009 IEEE Conference on Commerce and Enterprise Computing)

BPM-Anwendungen testen

Komplexität beherrschen mit: Patterns

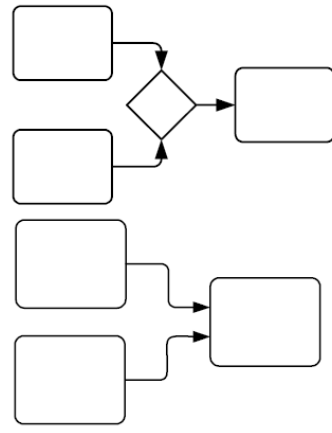
- // Es ist noch kein Meister vom Himmel gefallen.
- // Pattern-Analyse findet "ungünstige Patterns" und schlägt eine bessere Modellierung vor.
- // BPMN 2.0 Beispiel
 - // (V. Gruhn, R. Laue, Uni Leipzig, Cognitive Informatics, 2009. ICCI '09. 8th IEEE International Conference); Assessment of ~1000 process models – including the SAP R/3 reference model
 - // Das folgende Pattern wurde 28 mal gefunden



BPM-Anwendungen testen

Komplexität beherrschen mit: Konventionen

- // Die Pattern-Analyse kann mit eigenen Patterns erweitert werden. Beispielsweise um spezielle Anforderungen und Unternehmens-Konventionen einzuhalten. Ähnlich zu Checkstyle.

Merging	<p>BPMN uses the term "merge" to refer to the exclusive combining of two or more paths into one path (also known as an OR-Join). A Merging Exclusive Gateway is used to show the merging of multiple Sequence Flows (see upper figure to the right).</p> <p>If all the incoming flow is alternative, then a Gateway is not needed. That is, uncontrolled flow provides the same behavior (see lower figure to the right).</p>	
Looping	BPMN provides two mechanisms for looping	

< Alternative 1

< Alternative 2

Source: OMG BPMN 2.0 Specification, Seite 38

- // Andere Aspekte können ebenfalls teil der Überprüfungen sein - bspw. ob die Prozessmodelle dokumentiert sind.

BPM testen: verschiedene Techniken

Kosten reduzieren, Qualität erhöhen

- // Automatisiertes Testen ermöglicht...
 - // Kontinuierliches Testen und höhere Testabdeckung
 - // Einfache Regressionstests
 - // Unmittelbares Feedback an Entwickler - Fehler werden nicht auf die nächste Umgebung gepusht (Entwicklung, Test, Integration, Vorproduktion, ...)

- // White-Box-Tests erlauben präzise Tests
 - // Test-Abdeckung
 - // Hat der Prozess den richtigen Weg durchlaufen? (Man sieht einem Prozessmodell nicht an, ob die Bedingung hinter einer Verzweigung richtig ist oder nicht.)

- // Pattern-Analyse überprüft das Prozessmodell auf ungünstige Patterns, Verletzungen von Vorgaben und Konventionen.

- // Metriken geben Auskunft über die zukünftige Wartbarkeit, als auch Verständlichkeit eines Prozessmodells.

Gut, verstanden.

Aber Moment - was soll ich alles machen? Das bekomme ich doch nie richtig hin...



„Continuous Integration“



Einführung



Was ist Continuous Integration (CI)



Was ist Geschäftsprozessmanagement (BPM)



BPM-Anwendungen Testen



CI mit BPM, Jenkins und Sonar



Continuous Integration mit...



IBM BPM (kommerziell)



Activiti (open-source)



Zusammenfassung

Jenkins & Sonar

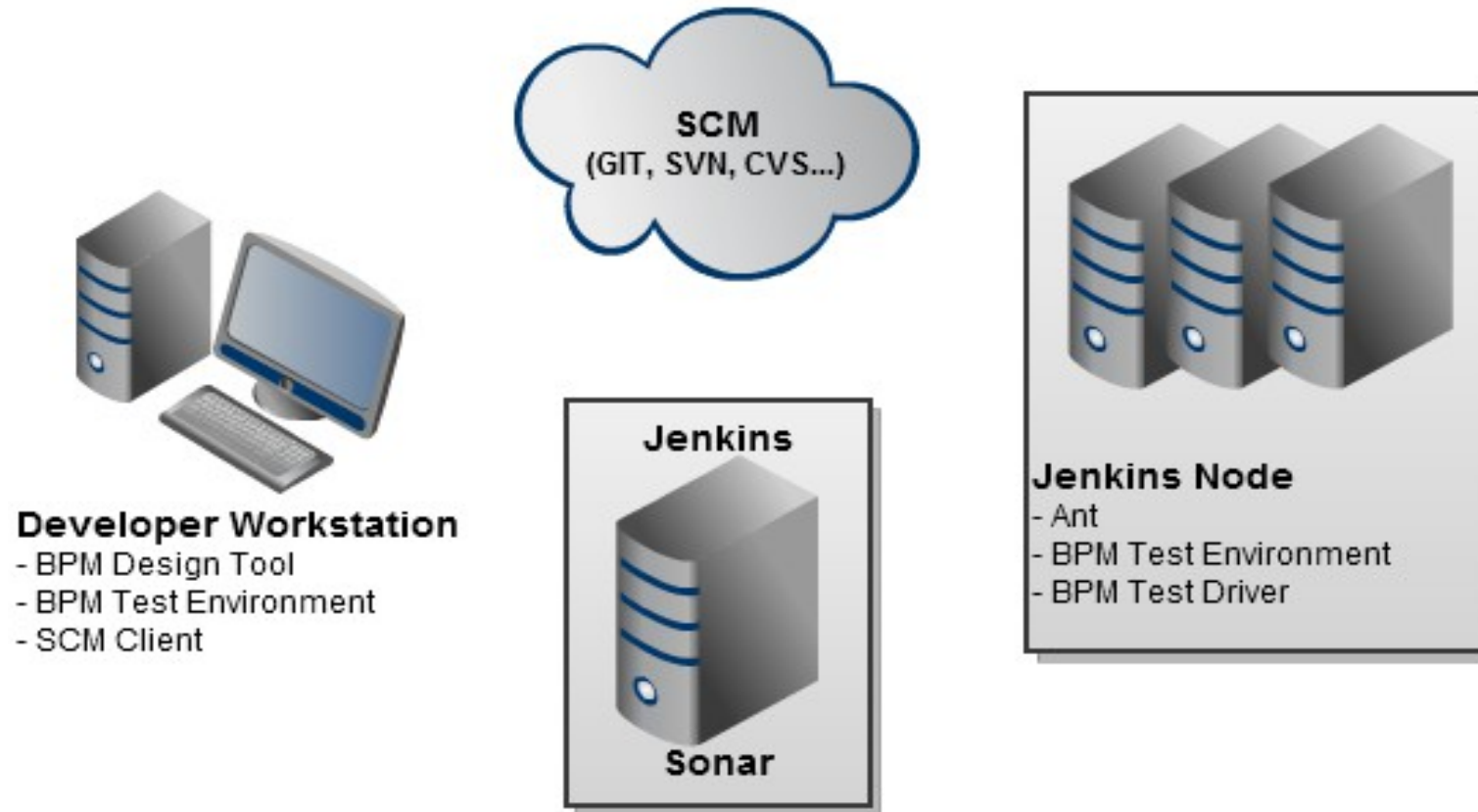
Jobs & Ergebnisse

- // Jenkins (ehem. Hudson) ist ein webbasiertes Open-Source CI Tool, programmiert in Java
 - // Automatisierung von Build, Deployment und Test von Anwendungen
 - // Kann alle möglichen Arten von Jobs ausführen - Apache Ant, Maven, Batch, shell scripts
 - // Unterstützt SCM Tools wie bspw. Git, SVN, CVS etc..
- Erweiterungen
 - Hunderte Plugins für alle möglichen Belange
- // Sonar
 - // Eine Plattform für Code-Qualität
 - // Über ein Plugin von Jenkins steuerbar
 - // NovaTec entwickelt im Zuge von Bachelor/Master/studentischen Arbeiten Plugins Prozessmodell-Analyse, Testabdeckung von Prozessmodellen



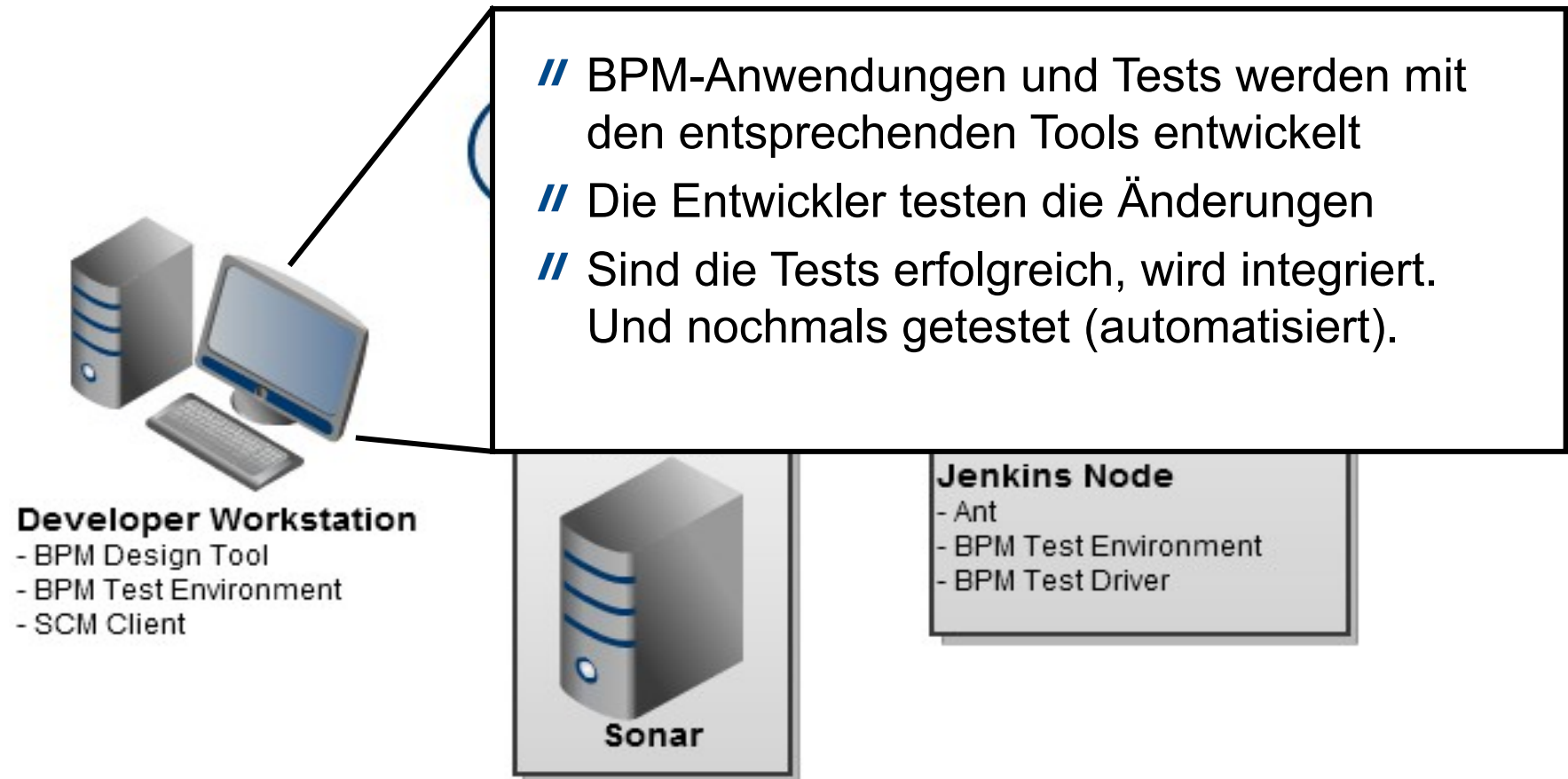
CI mit BPM, Jenkins und Sonar

Jenkins : Jobs - Sonar : Ergebnisse



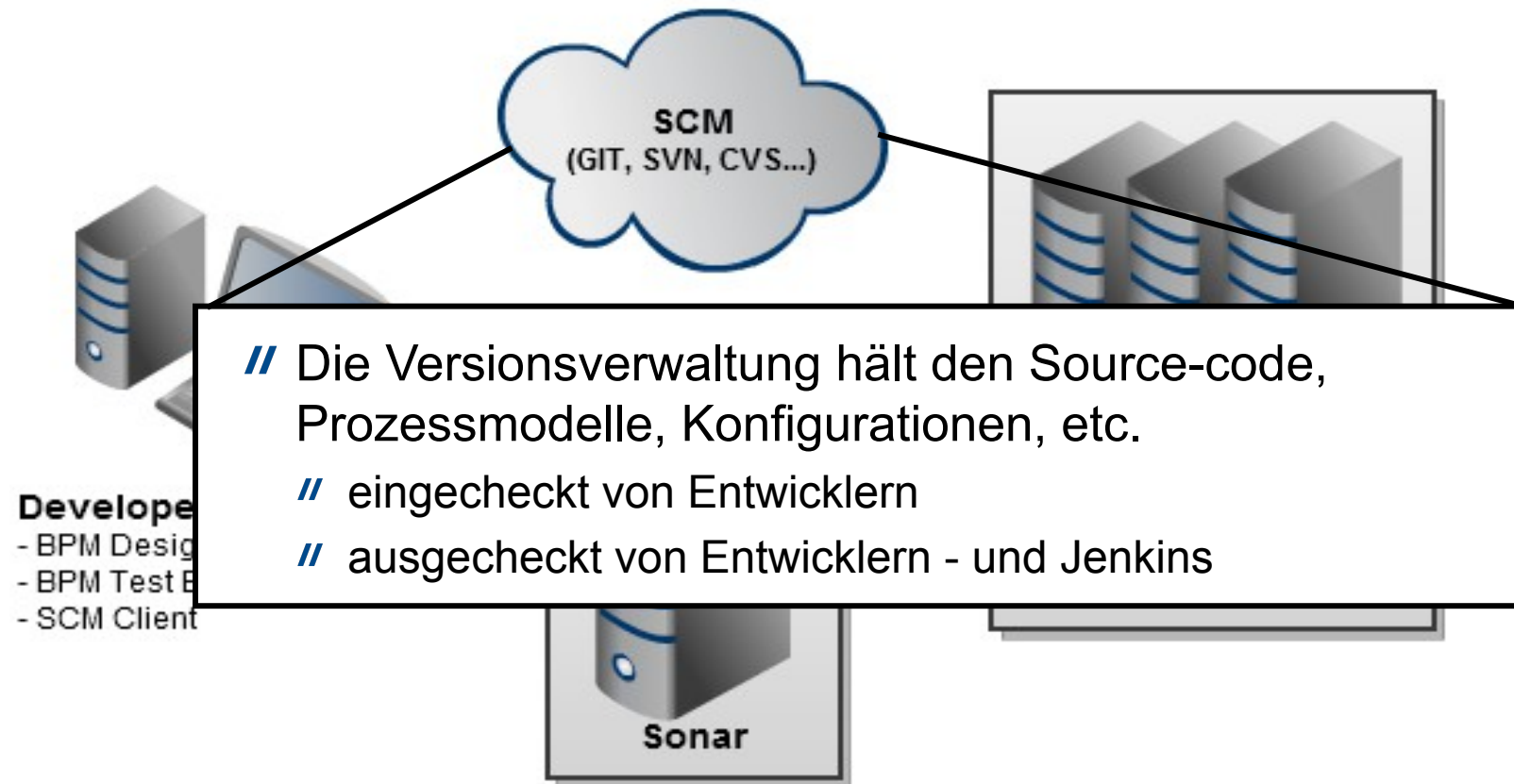
CI mit BPM, Jenkins und Sonar

Jenkins : Jobs - Sonar : Ergebnisse



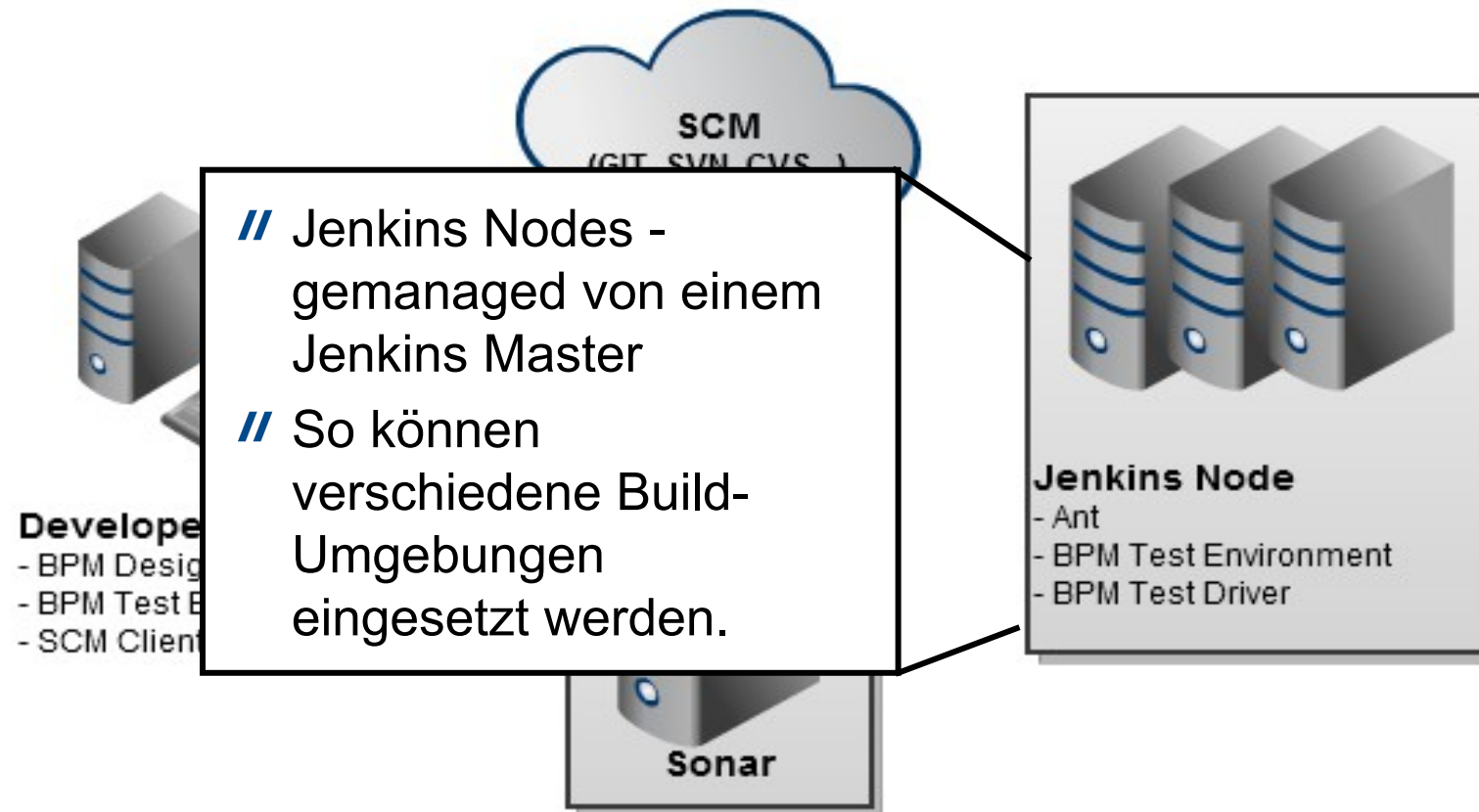
CI mit BPM, Jenkins und Sonar

Jenkins : Jobs - Sonar : Ergebnisse



CI mit BPM, Jenkins und Sonar


Jenkins : Jobs - Sonar : Ergebnisse



CI mit BPM, Jenkins und Sonar

Jenkins : Jobs - Sonar : Ergebnisse

- // Jenkins
 - // Startet die definierten Jobs
 - // Checkt den Code aus und transferiert ihn zu den Build-Servern
 - // Reicht Resultate weiter an Sonar
- // Sonar visualisiert die Test-Ergebnisse



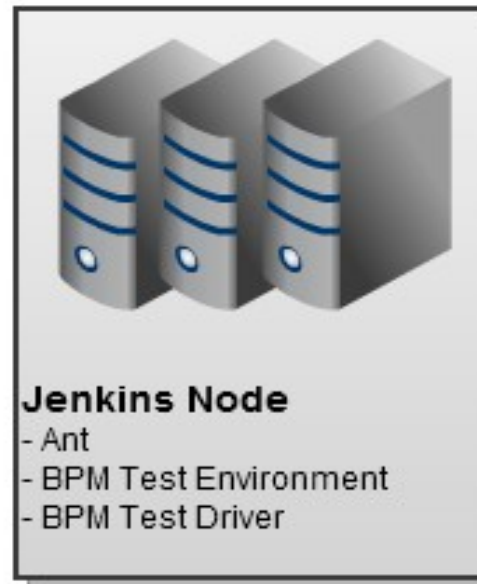
Developer Workstation

- BPM Design Tool
- BPM Test Environment
- SCM Client



Jenkins

Sonar



CI mit BPM, Jenkins und Sonar

Zusammenfassung

- // Prozessmodelle und Tests werden Jenkins zur Verfügung gestellt
 - // Je nach eingesetzter Software, sind die Prozessmodelle im "freien" SCM oder im vom Produkt zur Verfügung gestellten Repository. Jenkins kann mit jeglicher Konfiguration umgehen - zur Not über angepasste Skripte.
- // Jenkins kann den Build anstoßen, als auch das Deployment (auf dem Application Server bspw.). Insgesamt kann an dieser Stelle bereits Komplexität entstehen:
 - // Umgebung starten (bspw. Application Server)
 - // Auschecken des letzten Code-Standes und Bauen der Anwendung, was bei BPM-Anwendungen oft eine "Spezialität" ist
 - // Deployen der Anwendung und Ausführen der verschiedenen Tests
- // Sonar
 - // Analysiert die Test-Resultate (Code/Model-Coverage)
 - // Führt statische Checks auf das Prozessmodell durch (Komplexität, ...)
- // Schlussendlich steht dem Entwickler eine Reihe nützlicher Information bereit: Testabdeckung des Prozessmodells, Patternanalyse, Ergebnisse der Tests, Prozessmetriken, usw.



Einführung



Was ist Continuous Integration (CI)



Was ist Geschäftsprozessmanagement (BPM)



BPM-Anwendungen Testen



CI mit BPM, Jenkins und Sonar



Continuous Integration mit...



IBM BPM (kommerziell)



Activiti (open-source)



Zusammenfassung

- // Die Herausforderung, CI auf IBM BPM anzuwenden, ist die entsprechenden Produkt-Interfaces zu kennen bzw. die Umgebung sinnvoll zu skripten
 - // "Headless WID", Process Center, Process Server
 - // Erstellen der Artefakte die für eine Anwendung benötigt werden (bspw. EAR-Files oder ZIP-Files)
 - // Deployment / Undeployment von Anwendungen
- // Insgesamt können IBM BPM Anwendungen mit CI-Methoden entwickelt und getestet werden. Je nach Technologie und Anwendung entsteht dabei mehr oder weniger Aufwand. (BPEL oder BPMN basierte Prozesse)
 - // Der Integration Designer bietet bspw. eine Komponente zur Erstellung von Testcases, die dann bei Updates ausgeführt werden können
- // Die NovaTec arbeitet an der Entwicklung von Plugins, welche
 - // Patterns und Konventionen in Prozessmodellen überprüfen
 - // Testabdeckung von Tests auch in Bezug auf Prozessmodelle ermitteln (für BPEL und für BPMN)

Continuous Integration mit Activiti

Zusammenfassung

- // Als Open-Source Java Projekt lässt sich Activiti erwartungsgemäß einfach in eine CI-Umgebung integrieren:
 - // Eigens für bspw. JUnit vorgesehene Klassen werden von Activiti bereitgestellt, um das Testen so einfach wie möglich zu gestalten. So kann bspw. vor dem Start eines JUnit-Tests der entsprechende Prozess deployed werden.
 - // Activiti als Process Engine kann embedded betrieben werden: es muss kein Server gestartet werden, um einen Prozess auszuführen
 - // Activiti bietet Unterstützung für Mocks
- // Die NovaTec arbeitet an der Entwicklung von Plugins, welche
 - // Patterns und Konventionen in Prozessmodellen überprüfen
 - // Testabdeckung von Tests auch in Bezug auf Prozessmodelle ermitteln (für BPEL und für BPMN)



Einführung



Was ist Continuous Integration (CI)



Was ist Geschäftsprozessmanagement (BPM)



BPM-Anwendungen Testen



CI mit BPM, Jenkins und Sonar



Continuous Integration mit...



IBM BPM (kommerziell)



Activiti (open-source)



Zusammenfassung, Übungen

// Continuous Integration

- // Wird auf breiter Java-Front eingesetzt
- // Jenkins als CI-Server ist weit verbreitet

// & Testing

- // Gute und regelmäßige Tests sind unerlässlich in der Softwareentwicklung.
- // Sonar ist ein beliebtes Tool zur Qualitätskontrolle und lässt sich auf einfache Weise in Jenkins integrieren.

// meets BPM

- // Business Process Management (Geschäftsprozessmanagement) Bestandteil von Lösungen in verschiedenen Bereichen.

// meets Reality

- // Testen von BPM-Anwendungen hat noch nicht den Reifegrad von Java erreicht.
- // Die heutigen Tools rund um BPM erlauben aber mit etwas Arbeit
 - > BPM-Anwendungen mit CI-Methodik zu entwickeln
 - > BPM-Anwendungen automatisiert zu testen
 - > BPM-Anwendungen zu analysieren

// Testen Sie Prozesse !

- // mit Activiti, Eclipse, Jenkins, Maven, PostgreSQL, Sonar, Tomcat
- // mit kommentierten Beispielprozessen und Beispieltests
- // in unterschiedlichen Konfigurationen

// Diskutieren Sie die Herausforderung, Prozesse zu testen!

- Wir führen IT-Projekte zum Erfolg -

Stuttgarter Test-Tage am 21. & 22. März 2013

Vielen Dank für Ihre Aufmerksamkeit!

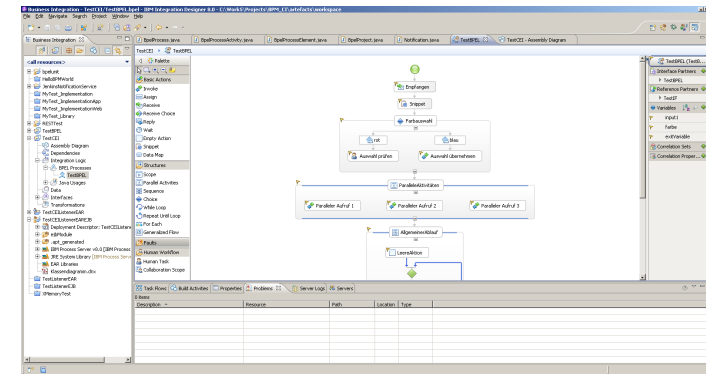
NovaTec – Unternehmensgruppe

Hauptniederlassung
Dieselstr. 18/1
D-70771 Leinfelden-Echterdingen

Telefon: +49 711 22040-700
Fax: +49 711 22040-899

E-Mail: info@novatec-gmbh.de
Internet: www.novatec-gmbh.de

- // Use the test component framework which is built into IID
 - // instead of writing your own testing framework
 - // test cases can be saved and executed by Jenkins later
 - // supports testing long-running BPEL processes with human tasks (!)
- // Use headless IID functionality or custom scripts to start/shutdown servers, deploy/undeploy applications
- // Restart the server used to run the test cases from time to time - but not with each "check-in"
- // Configure IBM BPM to emit default BPEL CEI events without any prior event definition configuration (!)
 - // trace string: WBILocationMonitor.CEI.bpc.*=all: WBILocationMonitor.CEI.task.*=all
 - // analyze CEI events to assess any test cases



- // Use the installation service to notify Jenkins about new deployments of a changed BPD



- // CEI Events can be used to verify that all paths have been taken by the defined test cases (e.g. using the NovaTec Model Coverage Plugin¹ which consumes CEI Events)
 - // For configuration of BPDs emitting CEI events see http://pic.dhe.ibm.com/infocenter/dmndhelp/v8r0mx/topic/com.ibm.wbpm.main.doc/topics/cmon_ev
- // Write test cases using the REST API
 - // The REST API Tester is your friend: <http://<server>:<port>/bpmrest-ui/>
 - // Note that that there might be "wait times" between different steps of the process model

„Continuous Integration“

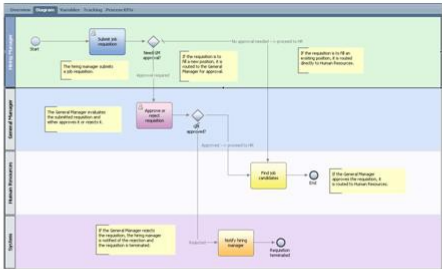
... a well-known software development practice!

Too many things I have to think about with testing. No way I can do this manually!



„Continuous Integration“

...works with IBM BPM and Tools!



„Continuous Integration“

- // Using Headless Build ANT Script to Generate Deployable EAR File for Server Deployment
 - // <http://www.ibm.com/support/docview.wss?uid=swg27023323>
- // Using ANT scripts for testing in headless WebSphere Process Server
 - // <http://publib.boulder.ibm.com/infocenter/dmndhelp/v7r0mx/index.jsp?topic=%2Fcom.ibm.wbit.help>
- // Building SOA composite business services, Part 6: Automate build and deployment steps
 - // <https://www.ibm.com/developerworks/library/ws-soa-composite6/>
- // Automating End-to-End Testing using Headless WebSphere Integration Developer
 - // <http://www.ibm.com/support/docview.wss?uid=swg27017772>
- // Jenkins
 - // <http://jenkins-ci.org/>
- // Sonar
 - // <http://docs.codehaus.org/display/SONAR/Hudson+and+Jenkins+Plugin>