

NaturalONE

Einführung von NaturalONE beim Kernkraftwerk Emsland

Martin Kottenstede, kub GmbH

12. November 2015



Einführung von NaturalONE beim Kernkraftwerk Emsland

Agenda

Vorgehen zur Einführung

Entwicklungsprozess

Projektzuschnitt

Initiale Befüllung der Repositories

Wartungs-Projekte

Weiterentwicklungs-Projekte

Deployment, Entwicklung Test&Prod.

Besonderheiten mit Natural for Ajax



Vorgehen zur Einführung



POC mit Software AG Global Consulting

Vorbereitung der Anforderungsanalyse beim Kunden

Vorbereitung des Projektplans beim Kunden

Bildung der Entwicklerteams

Vorbereitung der Server und Entwicklerarbeitsplätze

Entwicklerschulung NaturalONE/Natural for Ajax (SAG)

Kickoff

Projekteinführung

Versionsverwaltungssystem



Auswahl eines Versionsverwaltungssystems

Aufbau des Hauptentwicklungsstamms (Trunk)

Entwickler-Workshop

Aufbau des Wartungszweigs (Branch)

Aufbau des Entwicklungszweigs

Definition der Softwareverteilung (Deployment)

Bereitstellung der Softwareverteilung

Ticketsystem (Issue-Tracking-System)



Auswahl und Einführung Ticketsystem

Anbindung der Versionsverwaltung an das Ticketsystem

Implementierung von Userexits (Hooks) im Subversion

Einführung einer ticketbasierten Produktionsübernahme

Definition der neuen Softwareverteilung

Bereitstellung der neuen Softwareverteilung

QS Maßnahmen der Softwareverteilung

Entwicklungsprozess



Entwickler-PC

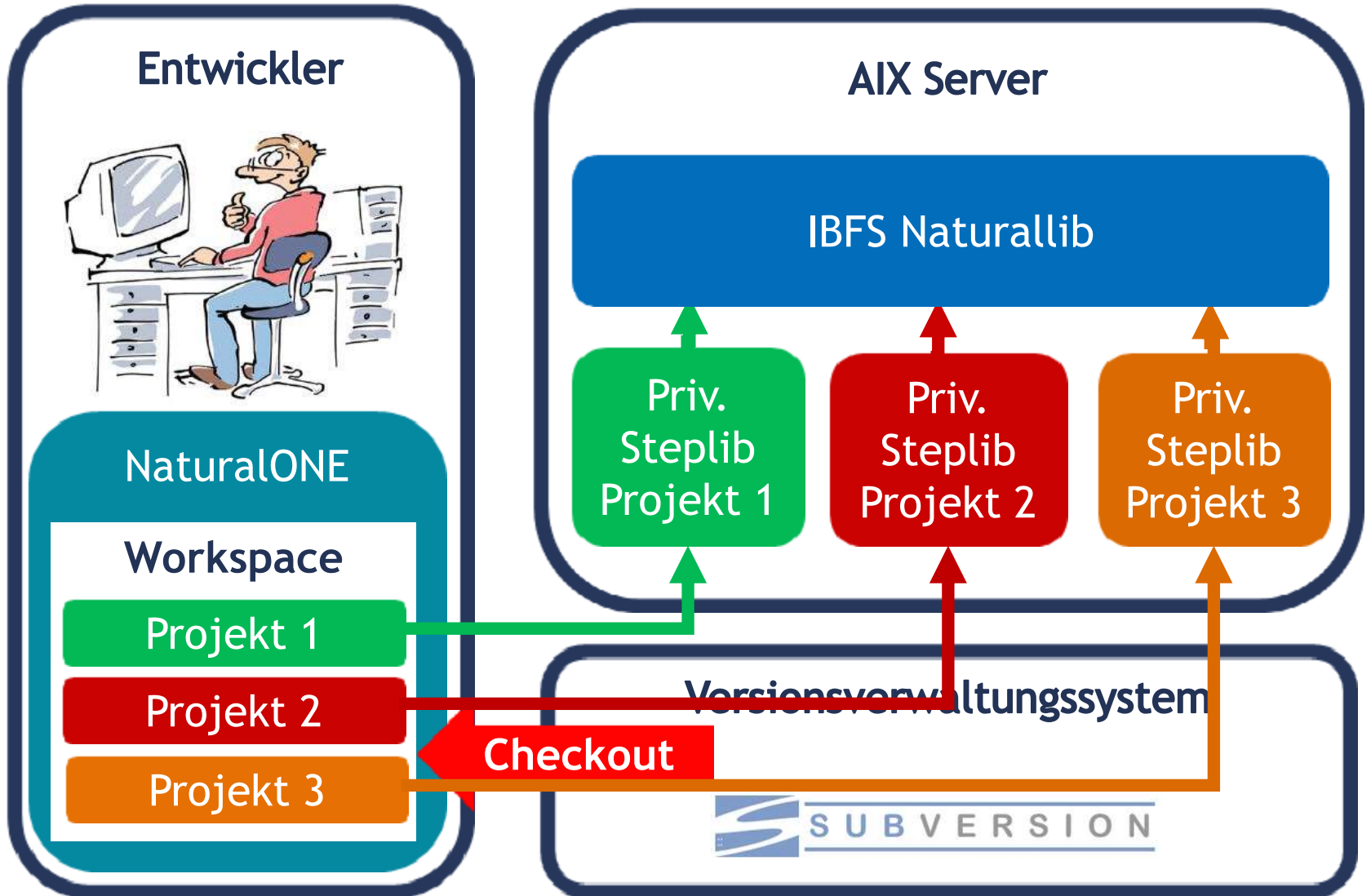


AIX Server



Windows Server





Warum wir den Localmode und keinen Servermode verwenden

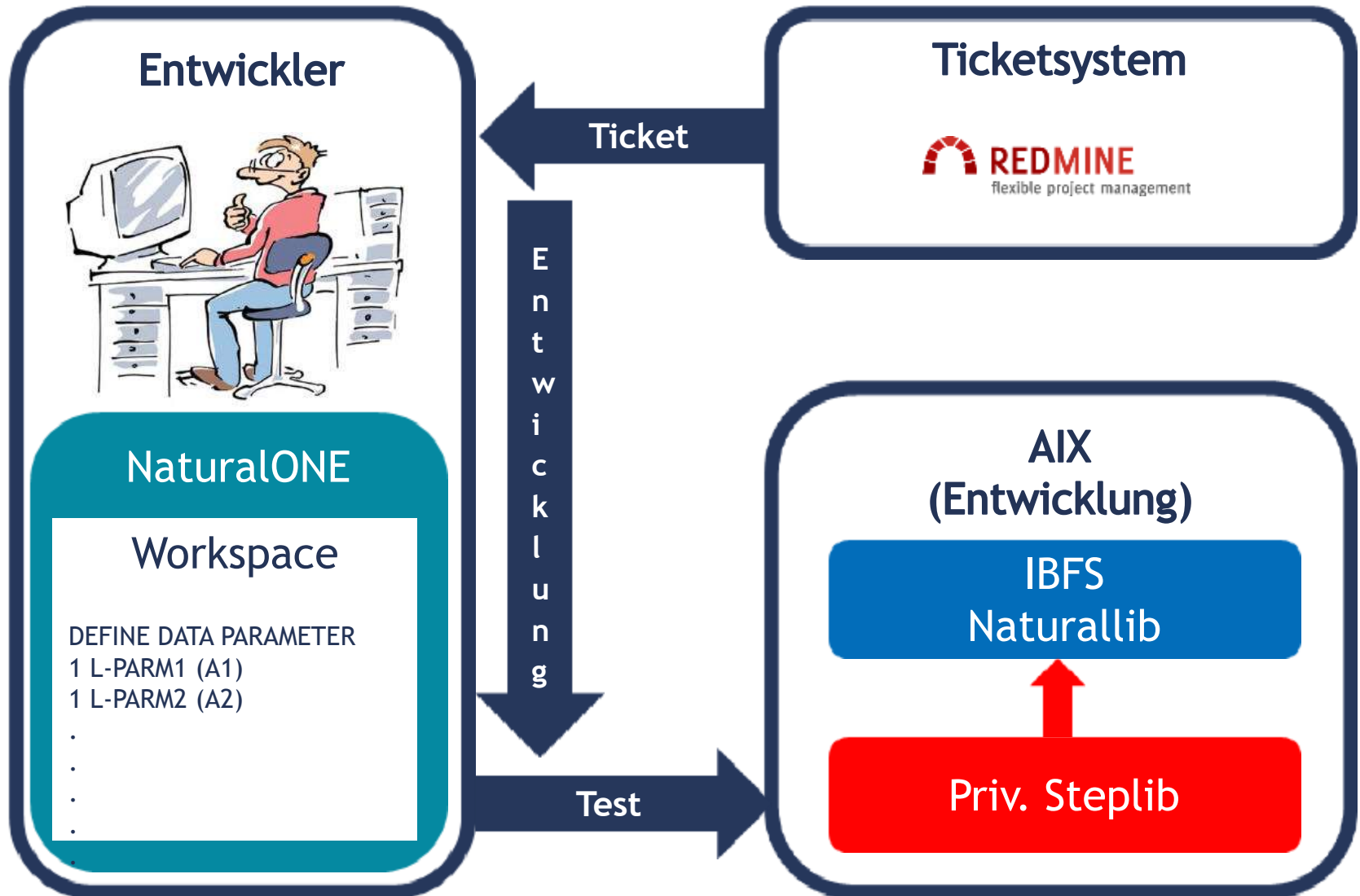
Eine parallele Bearbeitung von Softwarequellen ist innerhalb des Teams möglich (Kein “Lock” auf einzelne Module durch den Entwickler)

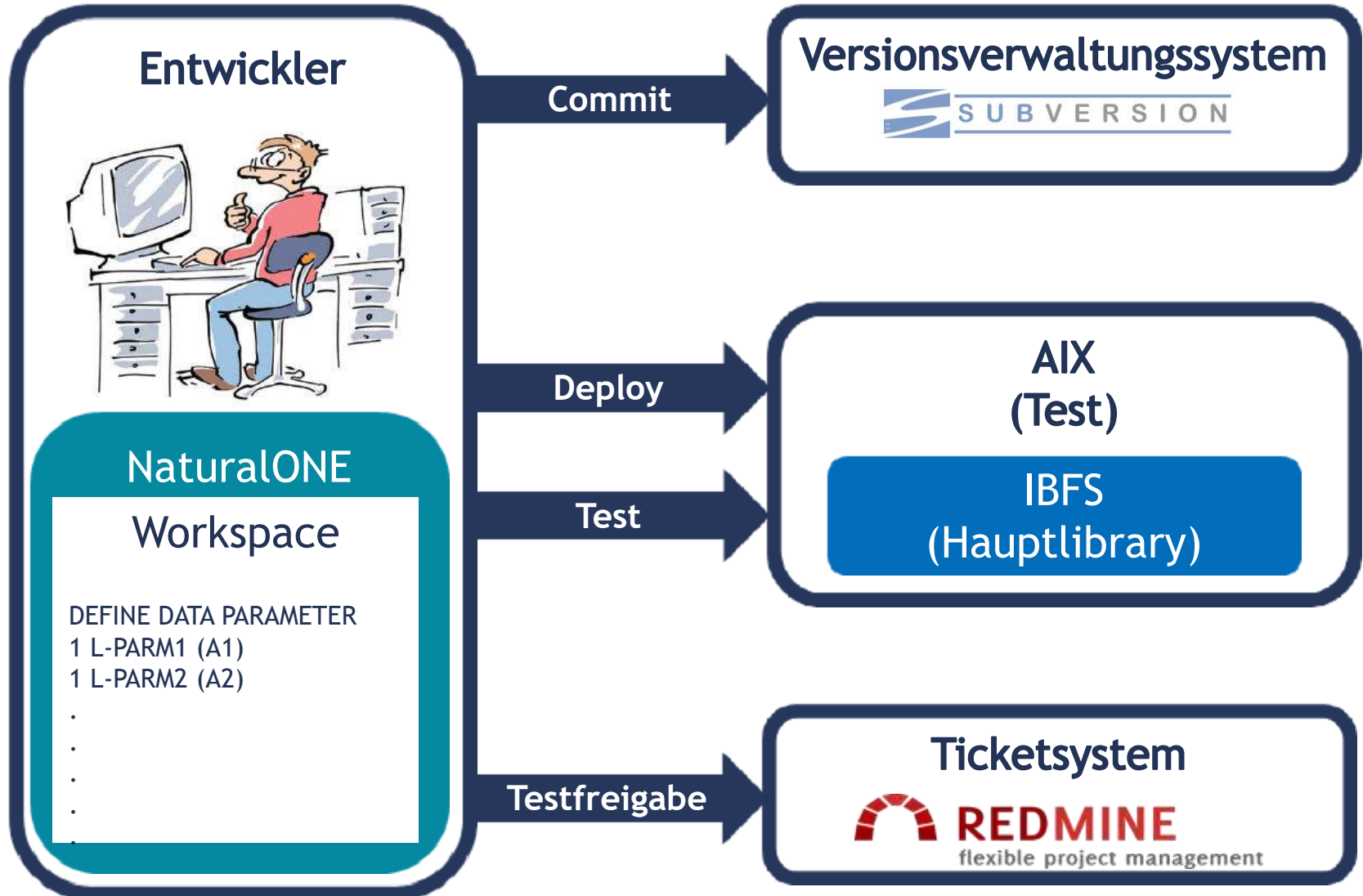
Keine Störungen durch Softwareänderungen innerhalb des Teams. Änderungen werden lokal im Workspace implementiert und getestet.

Mehr Transparenz durch die Anbindung eines Ticketsystems an die Repository der Versionsverwaltung.

Keine Verwaltung von Steplibs auf dem Naturalserver notwendig

Einfache “Zusammenführung” von Änderungen im Projekt durch Nutzung der Mergefunktion des Versionsverwaltungssystems





Projektzuschnitt



IBFS2020

GUI

Modernisierung der vorhandenen Natural-Maps auf moderne Browser-technologie

Beginn Sept. 2011

PL1

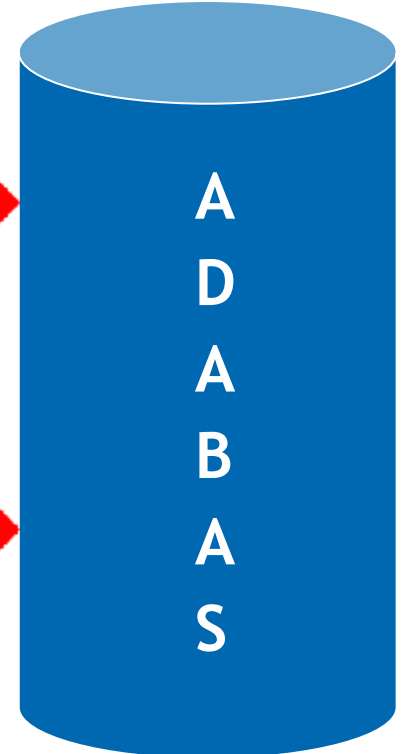
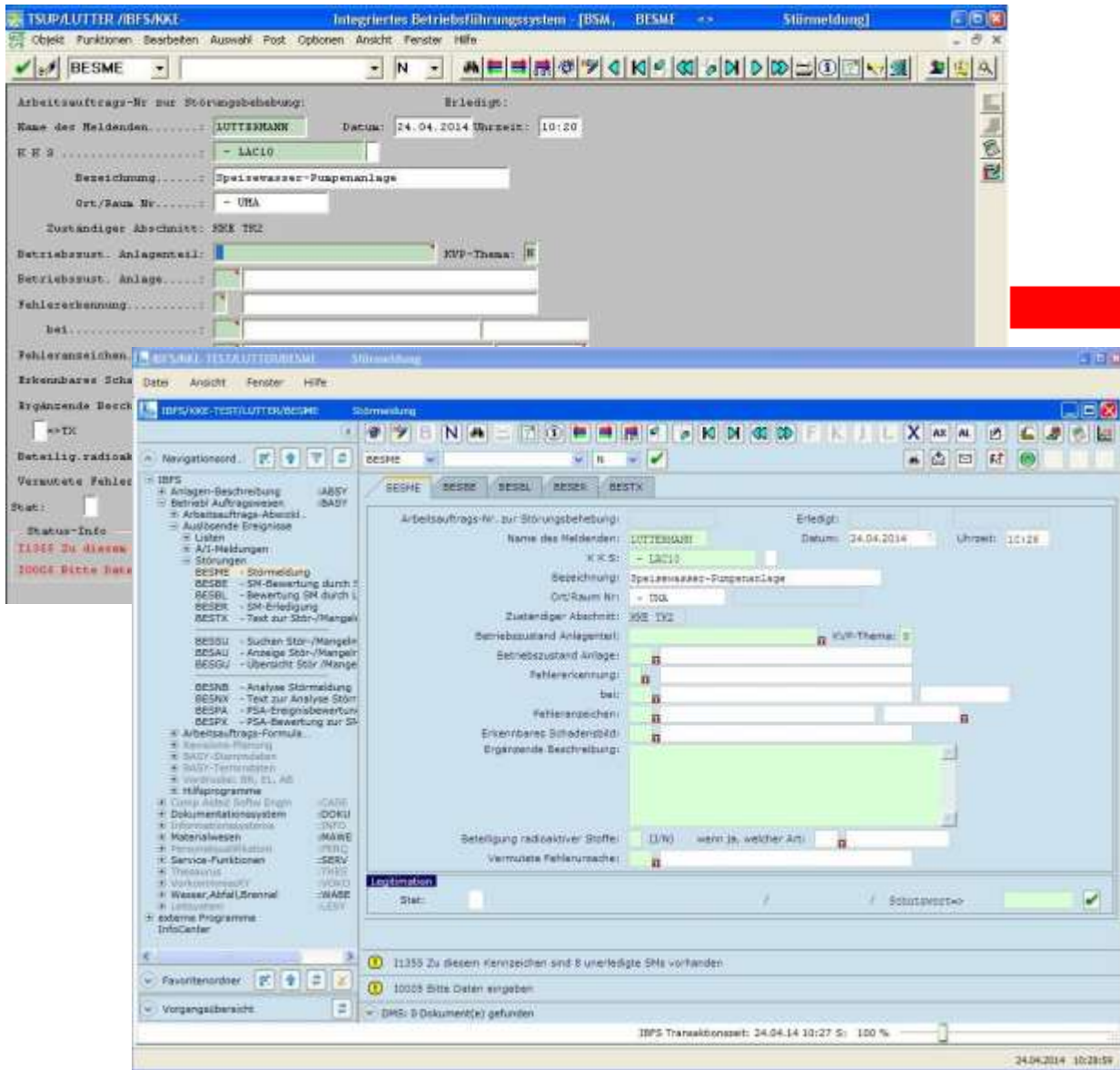
Ablösung der vorhandenen PL1-Module durch Natural-Substitute

Beginn Sept. 2011
Ende Juni 2013

FORM

Umstellung der vorhandenen Formulare (PRESCRIBE) auf PDF/A

Beginn Aug. 2013



Initiale Befüllung der Repositories



Es gibt nur eine Natural Hauptlibrary im Produktivsystem

Pro Teilprojekt werden alle Module der Hauptlibrary benötigt

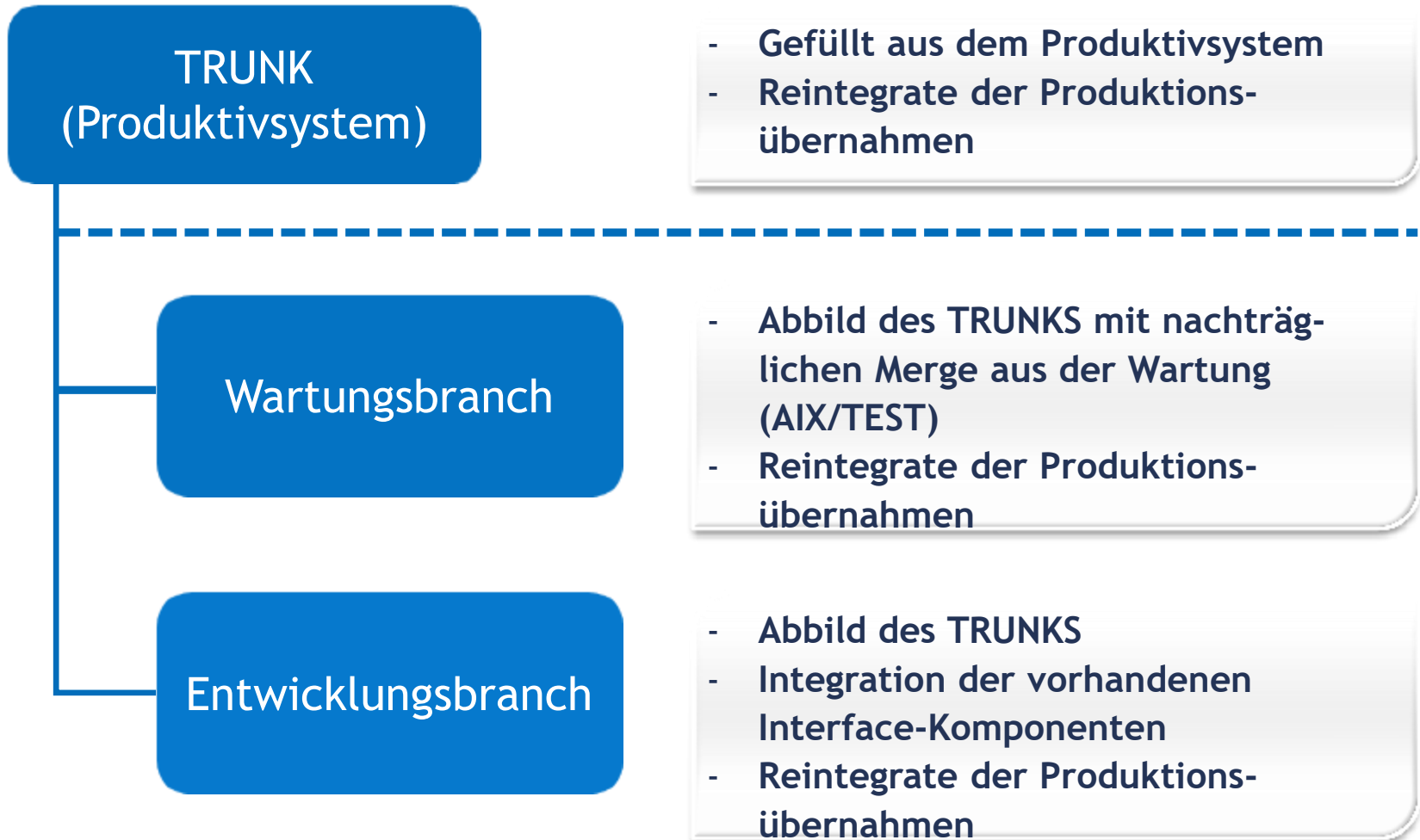
Alle Teilprojekte benötigen eine eigene Codebasis mit gleicher Herkunft

Alle Teilprojekte benötigen einen eigenen FUSER (AIX/TEST)

Alle Teilprojekte benötigen eine eigene Web-App im Tomcat (AIX/TEST)

Eine gemeinsame Datenbank für alle Teilprojekte

DAS PRODUKTIONSSYSTEM MUSS WARTBAR BLEIBEN



Definition von Ignorelisten im Subversion

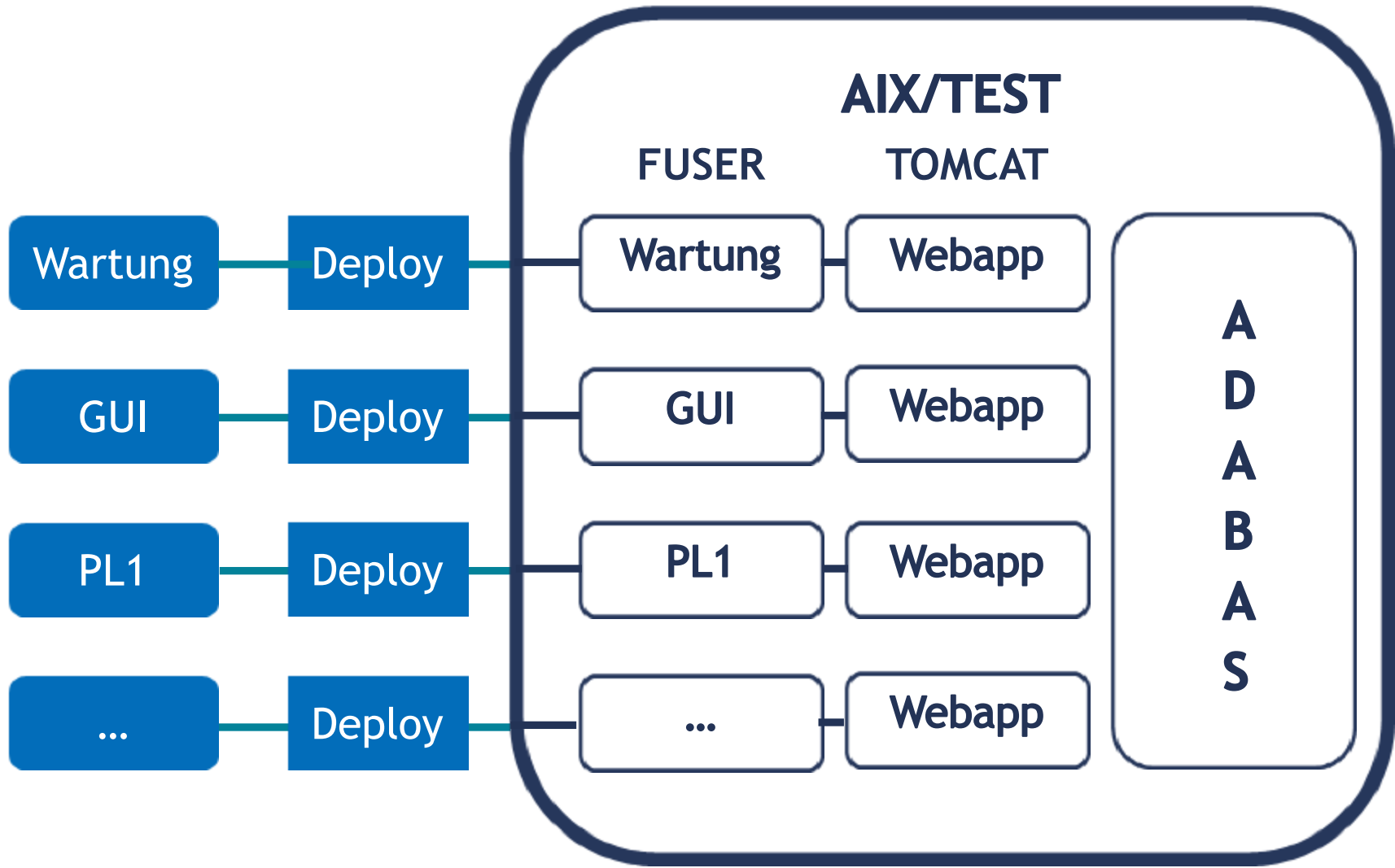
Befüllung des TRUNKS mit dem Produktivsystem

Definition der Excludelisten im Trunk

Bildung des Wartungsbranches

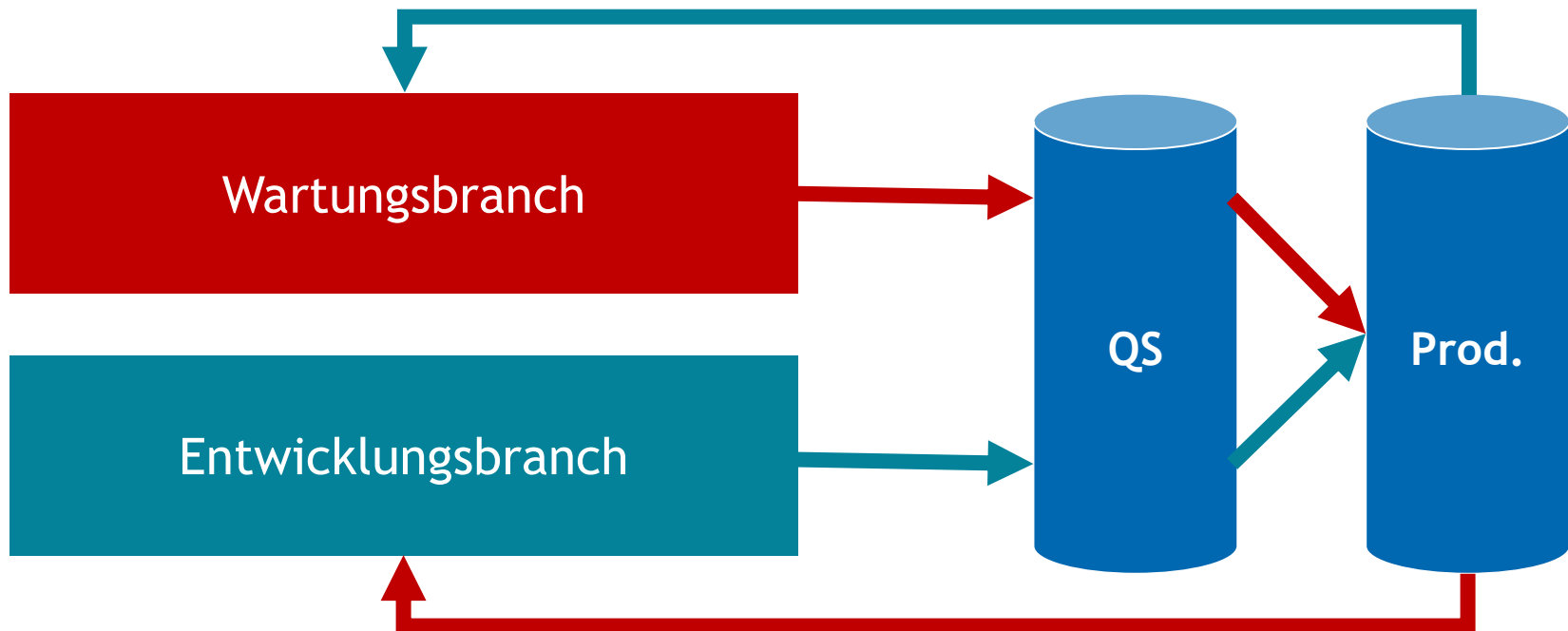
Bildung des Entwicklerbranches

Erstellen der Deployskripte



Wartungsprojekte und Entwicklungsprojekte





Die Branches laufen völlig unabhängig voneinander

QS-Exporte aus den Branches werden vorher durch Skripte QS-geprüft

Fachtests und Abnahme auf QS und anschließende Produktivsetzung

Änderungen in den Trunk mergen und in den Branch reintegrieren

Deployment, Entwicklung, Test & Produktion



Versionsverwaltung



Ticketsystem



Entwickler



Ticket Status "In Bearbeitung"

Entwicklung und Test im Workspace

Commit der Änderungen in Subversion

Commit erzeugt Changeset im Redmine

Verteilung(Deploy) und Test auf ENTW (AIX)

Verteilung und Test auf TEST (AIX)

Ticketstatus "Testphase"

Zuweisung Ticket zur Testabteilung

ENTW

TEST

Testabteilung



Betreuung Teilsystem



TEST

QS

PROD

Ticketsystem



Funktionstests durch die Testabteilung

Veranlassung Programmexport auf QS (AIX)

Ticketstatus “QS- und Integrationstest”

Ticketzuweisung an “Betreuung Teilsystem”

Fachtests durch “Betreuung Teilsystem”

QS-Abnahme durch Fachabteilungen

Ticketbasierte Produktivsetzung

Autom. Merge der Änderungen in die Branches

Besonderheiten mit Natural for Ajax



Erstellung von Custom-Controls möglich

Erstellung von Control-Templates möglich

Webpages sind erweiterbar mit Java/Java-Skript

Nur die XML-Pages werden im Subversion gehalten

Deploy einzelner Webpages möglich (eigenes Deployverfahren)

Versionsverwaltungssystem sehr zu empfehlen

Warum eine Versionsverwaltung?

Nachvollziehbarkeit der Änderungen in den Modulen

Rücksetzbarkeit der Änderungen in den einzelnen Modulen

Unabhängiges Entwickeln und Testen im Team

Gezieltes Zusammenführen von Entwicklungszweigen (Merging)

Gezieltes Überführen von Änderungen aus anderen Entwicklungszweigen

Enorme QS-Verbesserung durch den Einsatz von Userexits (Hooks)

Warum eine Ticketverwaltung?

Gezieltes Definieren von Aufgaben und Arbeitspaketen

Gezielte Steuerung der Arbeitsabläufe durch Workflow

Bildung von Changesets durch Anbindung an Subversion

Dadurch Produktivsetzung von Einzeltickets und Versionen möglich

Dokumentation der Arbeitsergebnisse (Testdokumentation etc.)

Verbessertes Projektmanagement

Vielen Dank für Ihre Aufmerksamkeit

Martin Kottenstede

kub GmbH
www.kub.de
Phone: +49 591 9010690
info@kub.de

