MT AG

business by integration

Was ist professionelle APEX Entwicklung?

Tobias Strohmeyer

Düsseldorf, 15.09.2015

Agenda

1. Was versteht man darunter?
2. Architekturen
3. Versionierung
4. Ticket Systeme
5. Vorgehensweisen
6. Enterprise APEX

MT AG

Was versteht man darunter?

MT AG

Herkömmliche Praxis

Was ist professionelle APEX Entwicklung?

- 1 einzelner Entwickler
- 1 Umgebung (Produktion)

- Probleme:
  - Änderungen werden direkt in Produktion durchgeführt
  - Keine Nachverfolgung von Problemen
  - Keine Transparenz über Versionsentwicklung
  - Fehlende Dokumentation

MT AG

Entwicklungsumgebung

Professionelle APEX Entwicklung

- Aufsetzen einer Entwicklungsumgebung (DEV)
  - Gekapselt von der Produktionsumgebung (PROD)
  - Identisches System (gleiche Versionen der Komponenten)

- Entwicklung fortan auf der DEV
- Export von Applikation und DB-Objekten -> Import in PROD
  - Erstellung von Skripten für Export und Import
  - *Problem: Deploymentskripte, ungetestet und können fehlschlagen*
- Test der Anwendung in DEV durch Tester (Anwender)
  - *Problem: Konflikt zwischen Entwickler und Tester*

MT AG

Testumgebung

Professionelle APEX Entwicklung

- Aufsetzen einer Testumgebung (TEST)
  - Gekapselt von PROD und DEV

- Test von Änderungen an der Applikation
  - Nutzung echter Produktionsdaten
- Test der Deploymentskripte
  - Keine Probleme bei Deployment nach PROD = geringere Downtime
- Weniger Konflikte zwischen Entwickler und Tester

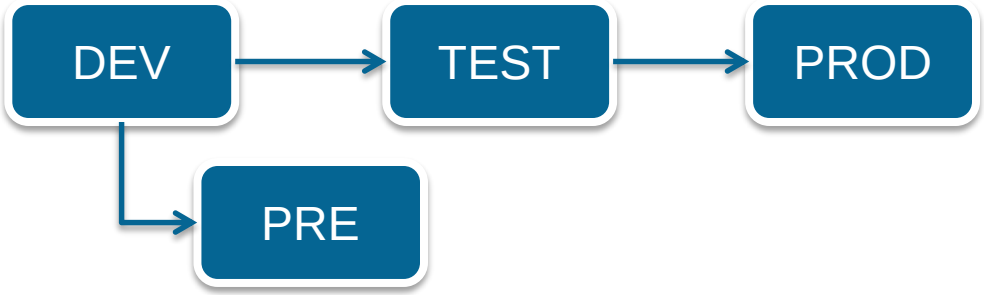
MT AG

Pre-Testumgebung

Professionelle APEX Entwicklung

- Aufsetzen einer Pre-Testumgebung (PRE)
  - Analog zu TEST aufgebaut

- Test der Deploymentskripte
- Keine Konflikte mehr zwischen Tester und Entwickler
  - Verringerung der Down-Time
  - Tester sind ungestört auf der TEST (größere Testszenarien möglich)



MT AG

Architekturen

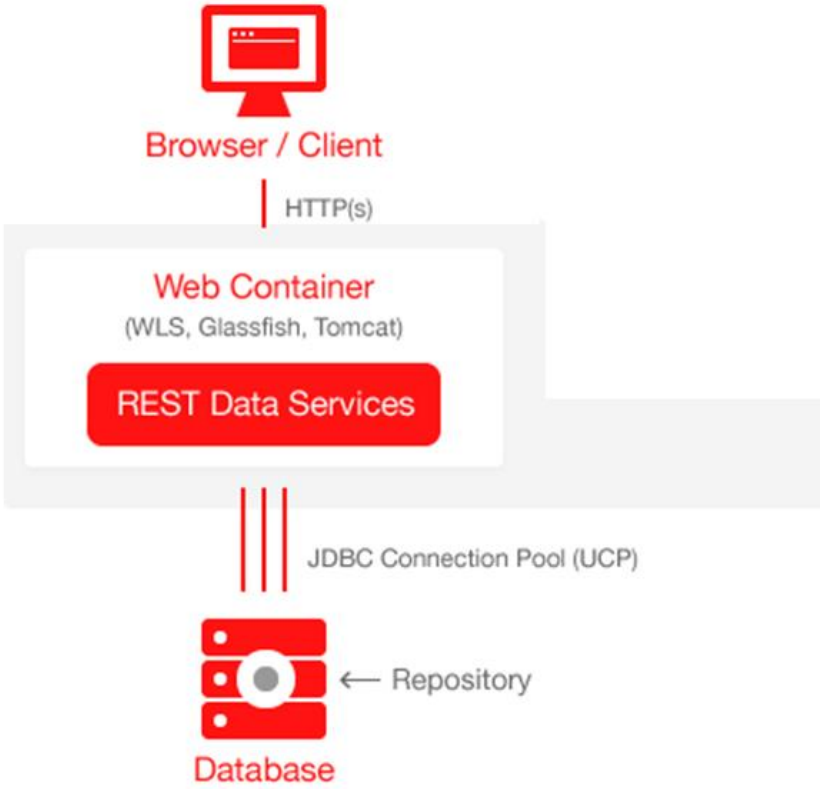
MT AG

Aufbau der Entwicklungsumgebung

Architekturen

- Web Container (Tomcat)
  - ORDS (auch standalone möglich)
- Datenbank (Oracle)

- Alternative (nicht empfehlenswert)
  - Oracle HTTP Server (OHS)
  - Embedded PLSQL Gateway (EPG)

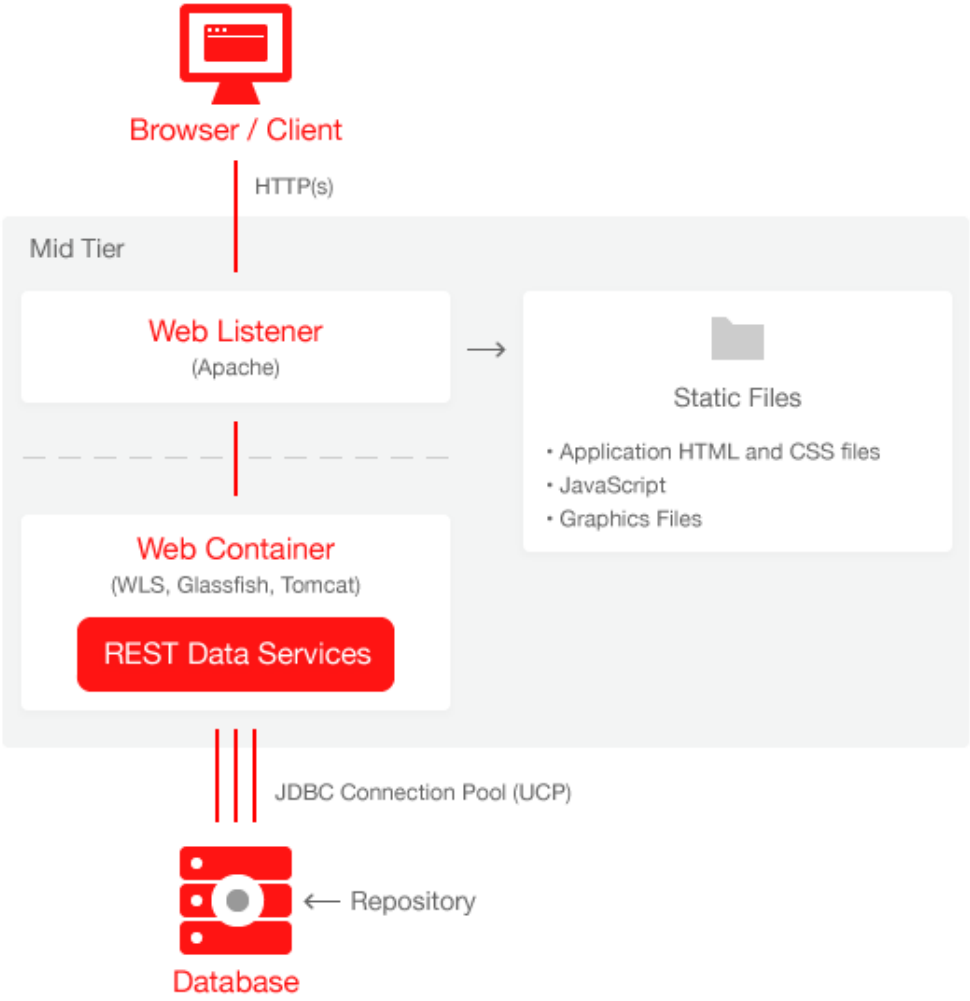


MT AG

Aufbau der Produktionsumgebung

Architekturen

- Webserver (Apache)
  - Zugangskontrolle (SSO)
  - Static Files
  - Module (Kompression, Proxy)
- Web Container (Tomcat)
  - Virenscanner
  - Security (kein Durchgriff zur DB)
  - ORDS
- Datenbank (Oracle)
  - Security



MT AG

ORDS Konfiguration

Architekturen

- **../ords/defaults.xml**
- http://krisrice.blogspot.de/2012/05/apex-listener-jdbcsecurity-setup.html

```
<!-- Zeit bis die Anzahl Verbindungen auf das minimale Limit zurückgesetzt werden -->
<entry key="jdbc.InactivityTimeout">1800</entry>
<!-- Anzahl offener Verbindungen initial -->
<entry key="jdbc.InitialLimit">3</entry>
<!-- Anzahl Anfragen die maximal über eine Verbindung gehen -->
<entry key="jdbc.MaxConnectionReuseCount">1000</entry>
<!-- Anzahl Verbindungen die maximal möglich sind -->
<entry key="jdbc.MaxLimit">10</entry>
<!-- minimale Anzahl offener Verbindungen -->
<entry key="jdbc.MinLimit">1</entry>
```


MT AG

Versionierung

MT AG

Allgemeine Anforderungen

Versionierung

- Koordinierter Zugriff durch mehrere Nutzer
- Protokollierung von Änderungen von Dateien
- Archivierung einzelner Stände
- Wiederherstellung älterer Dateiversionen

Subversion SVN

Versionierung

- Server-Client Prinzip
- Funktionen:
  - Checkout
  - Update
  - Commit
- Unterstützt Locking
- Versionsnummerierung fortlaufend
- Schwieriger Merging-Prozess

Git

Versionierung

- Verteiltes Versionierungswerkzeug
- Kommt theoretisch ohne zentralisierten Server aus
- Einfaches Arbeiten an der Lokalen Version
- Distributierung auf Wunsch des Nutzers
- Unterstützung beim Merging von Dateien
- Keine Netzwerklatenz und wenig Speicherplatzbedarf
- Weitere Befehle wie push, pull, fetch

- Kein Locking auf Dateien und Ordner

Ticket Systeme

MT AG

Allgemeine Anforderungen

Ticket Systeme

- Priorität des Vorfalls
- Version
  - Wann aufgetreten, Bis wann zu lösen?
- Betroffene Komponenten
  - Gruppierung in Anwendung, Seiten, Prozesse, Funktionen
- Aufwände
  - Schätzung, bereits geleistet, noch offen
- Ticketstatus und Workflow
- Untertickets
- Personen
  - Rollen wie Erfasser, Bearbeiter, Tester, Schätzer

APEX Team Development

Ticket Systeme

- Bereits integriert in APEX
  - Angebunden an die Anwendungen
  - Betroffene Komponenten sind Anwendungen und Seiten

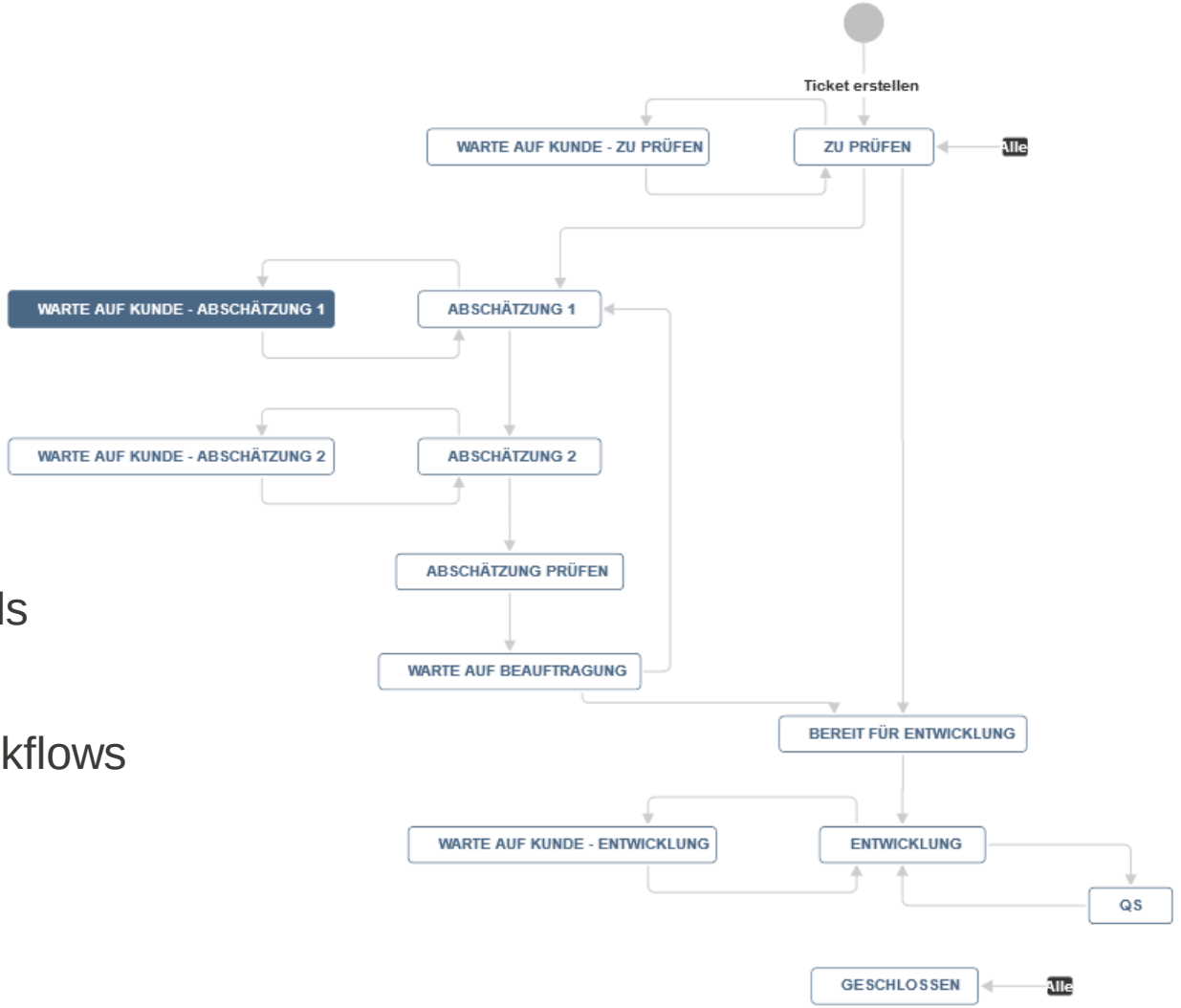
- Zuordnung zu Milestone, Release, Feature, ToDo
- Bug lässt sich als Duplikat markieren
- Unterstützt Tagging

- Keine Workflows
- Kein Frontend für die Benutzer
  - -> Möglichkeit ein eigenes Frontend zu realisieren

MT AG

JIRA

Ticket Systeme

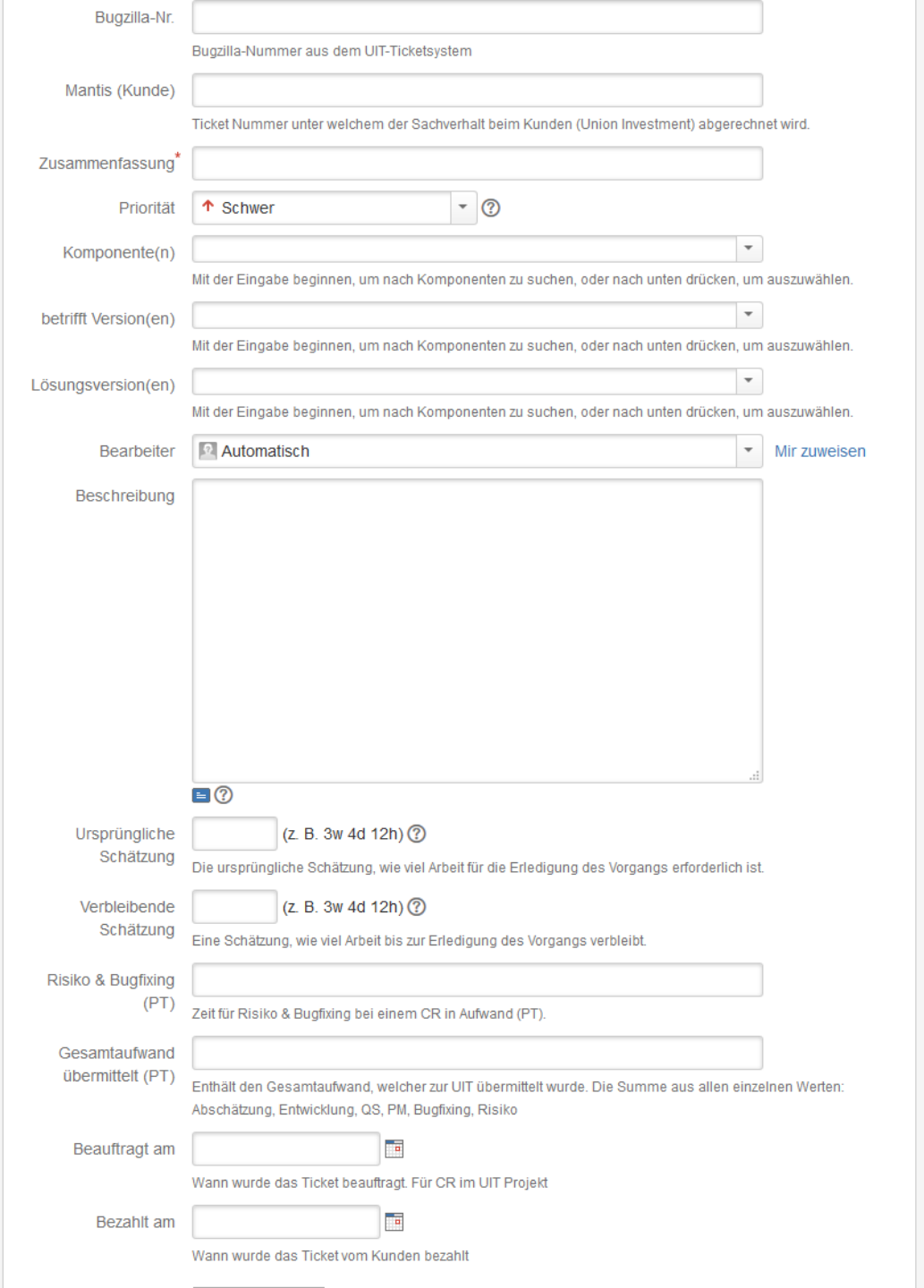


- Anbindung an Dev Tools
  - Git, Subversion, …
- Benutzerdefinierte Workflows
  - beliebig komplex
  - Events und Trigger

JIRA

Ticket Systeme

- Ticket Attribute
  - Projekt
  - Bugzilla Nummer
  - Zusammenfassung
  - Priorität
  - Betroffene Komponenten / Versionen
  - Bearbeiter
  - Aufwandsschätzung



Mantis Bug Tracker

Ticket Systeme

- Anbindung an Git und Subversion
  - Versionskontrolle
- Roadmaps
- Changelogs
- Dokumentation
- Projektmanagement

MT AG

Dokumentation und Testing

MT AG

Bisher …

Dokumentation und Testing

- Dokumentation
  - Nicht vorhanden
  - oder in Word-Datei
  - keine Verteilung an alle Nutzer
  - fehlender Zugriff auf Ressourcen für andere Entwickler

- Testing
  - durch den Entwickler selber
  - ohne fachliche Abnahme

Dokumentation

Dokumentation und Testing

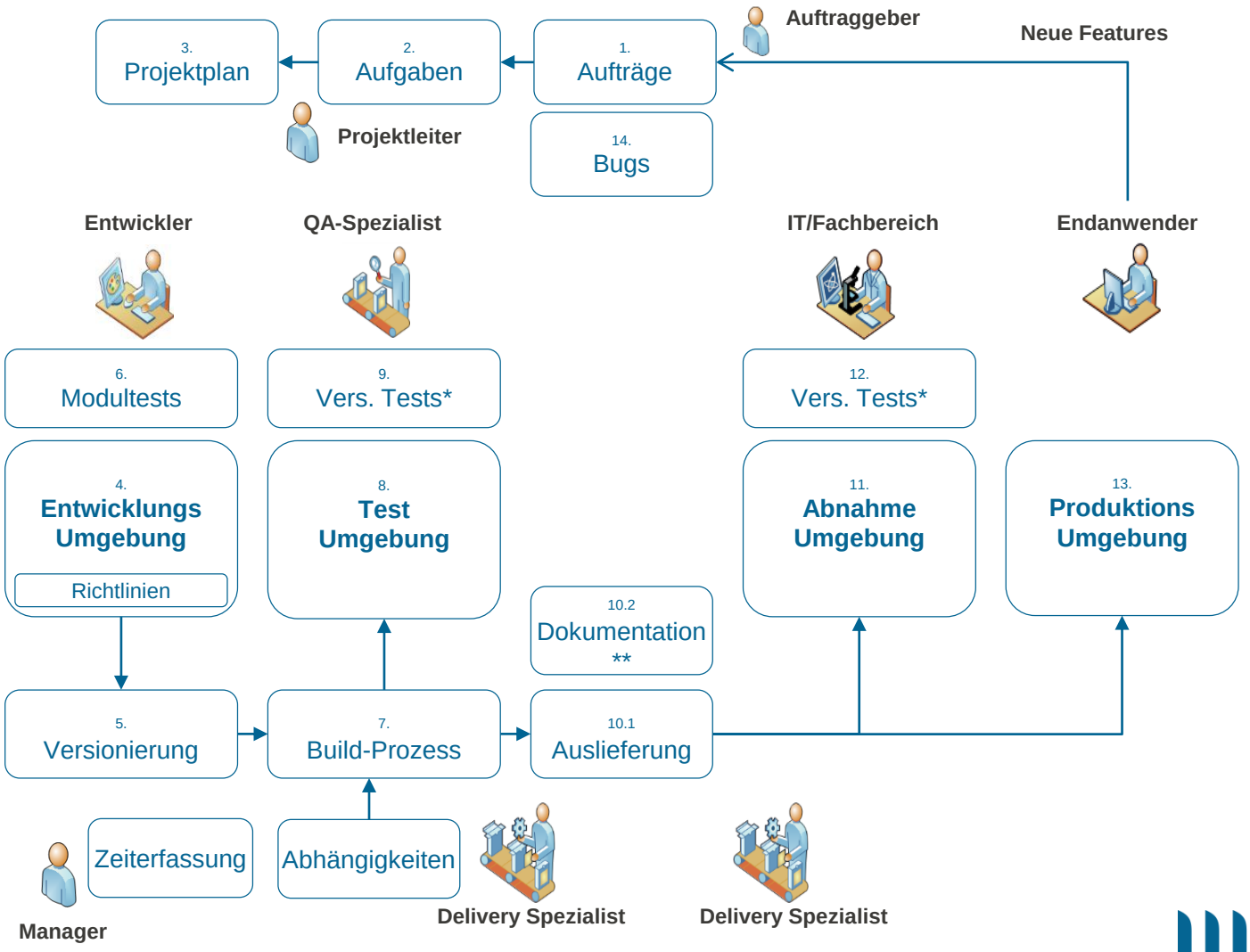
- Dokumentation von Änderungen an der Applikation
  - Change Management
    - Nachverfolgung von Änderungswünschen
  - Release Management
    - Nachverfolgung von realisierten Änderungen
    - Kann Change Requests und Tickets referenzieren
    - Freigabe durch Tester != Entwickler
    - Verknüpft mit Subversion

- Realisiert im APEX Team Development, Jira und Mantis

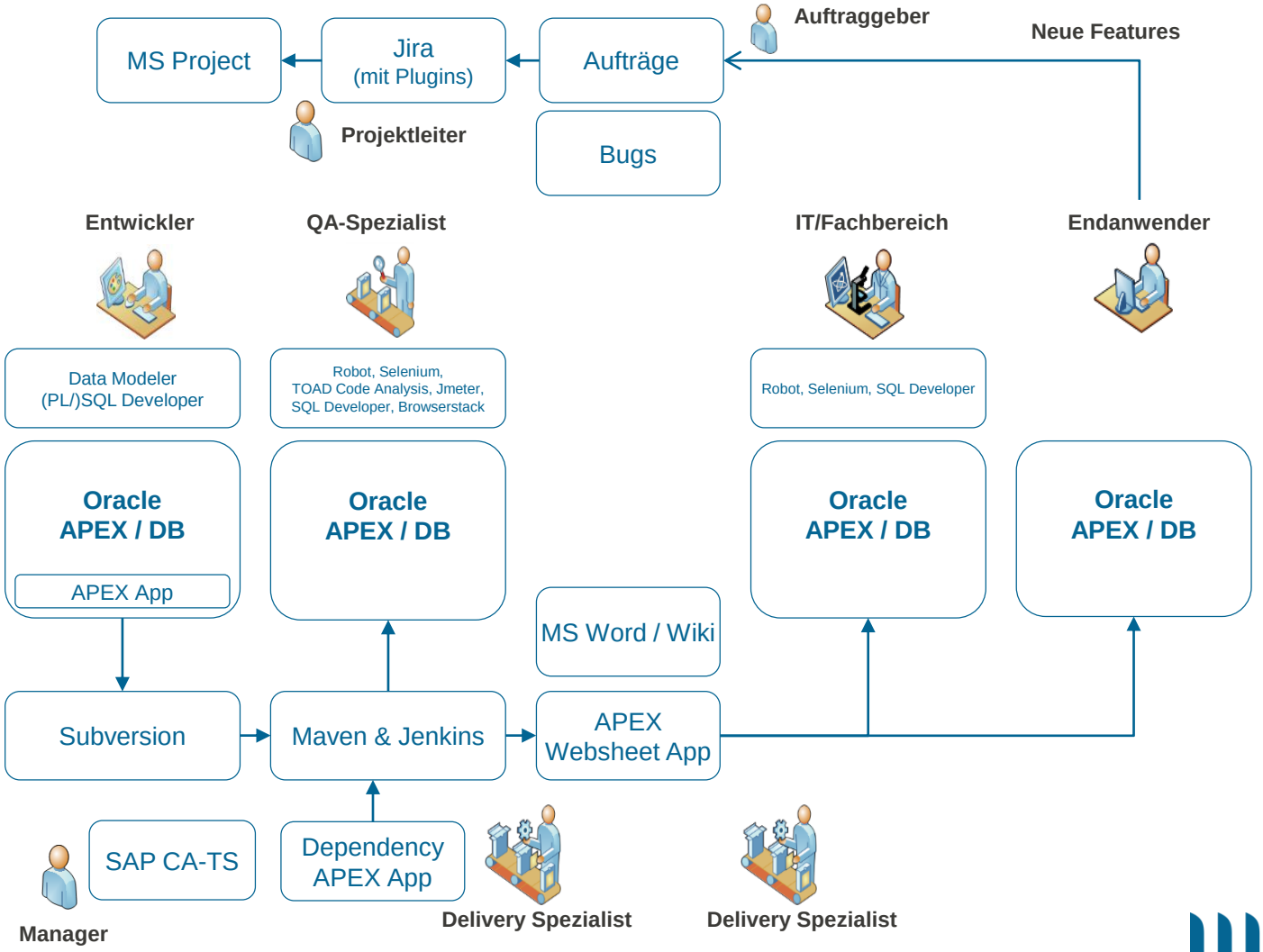
Vorgehensweisen

MT AG

Vorgehensweisen



Entwicklungstools



MT AG

Enterprise APEX

MT AG

Enterprise APEX

Das Team

- Projektleiter (1)
  - Steuerung des Projekts hinsichtlich der Ziele, Inhalte und Termine
  - sorgt für Transparenz im Projekt
- Architekt (1)
  - technisches und fachliches Gesamtverständnis
  - Konzeption der Anwendung und des Datenbankmodells.
- Datenbankmodellierer (1)
  - Konzeption des Datenbankmodells zusammen mit dem Architekt
  - Struktur und Realisierung des Datenmodells
  - Kümmert sich um ER-Diagramme und Stammdaten

Enterprise APEX

Das Team

- Datenbankentwickler (2)
  - kümmert sich um zentrale Logik
    (Berechtigungen, Schnittstellenlogik und fachlich komplexe Logik)
- APEX Entwickler (3)
  - Seitenentwicklung und die seitenspezifische Logik
  - koordiniert Änderungen (Tabellen => Datenbankmodellierer, zentrale Funktionen => Datenbankentwickler, Sonderfälle/Ausnahmen => Architekt)
- Der Unit Test Entwickler
  - Datenbank- und APEX Entwickler erstellen Tests für Komponenten, für dessen Programmierung er selbst nicht zuständig ist (Vieraugenprinzip und Wissenstransfer)

MT AG

Enterprise APEX

Das Team

- Qualitätssicherer (2)
  - Test jeden Sachverhaltes an Hand festgelegter
  - arbeitet auf einem Testsystem.
- Auslieferer
  - Schwerpunkt in Auslieferungen
  - inkrementell und vollständig
  - Funktion kann von dem Datenbankmodellierer oder dem Datenbankentwickler wahrgenommen werden.

Allgemeine Richtlinien

APEX Entwicklung

- APEX Seiten thematisch Gruppieren
- Seiten-IDs ab 200 aufwärts nutzen
  - Seite 1 bis 199 APEX spezifisch (z.B. Login 101)
  - 10er Inkrementierung verwenden
- Global Page auf Seite 0
- Definition von Themenbereichen
  - Testseiten können einfacher identifiziert werden
  - Nutzung von Seiten Aliasen

Namenskonventionen

APEX Entwicklung

- 3 stelliges Anwendungskürzel
  - Genutzt für selbständige Datenbankobjekte
  - Klare Erkennung dieser Objekte
- 4 stelliges Tabellenkürzel
  - Genutzt in Spalten und andere Tabellenabhängigen Objekten
- 2 stelliges Kürzel für PL/SQL Objekte

- Trennung durch “_“

Namenskonventionen

Tablespaces

- <Schemaname>_DATA
  - Daten des Schemas
- <Schemaname>_IDX
  - Indexe des Schemas
- <Schemaname>_LOB
  - LOBs des Schemas

MT AG

Namenskonventionen

Datenbankobjekte

- Tabellen
  - SFA_<TABELLE_NAME>
  - SFA_R_<TABELLE_NAME>
- Spalten
  - <TABELLENKÜRZEL>_<SPALTE_NAME>
- Primary Keys
  - SFA_<TABELLENKÜRZEL>_PK
- Foreign Keys
  - SFA_<KÜRZEL_AUSGANGSTAB>_<KÜRZEL_REFERENZIERTE_TAB>_FK
- Unique Keys
  - SFA_<TABELLENKÜRZEL>_UK

MT AG

Namenskonventionen

Datenbankobjekte

- Check Constraints
  - SFA_<TABELLENKÜRZEL>_<BEZEICHNUNG>_CK
- Indexe
  - <KEY_NAME>_I
  - SFA_<TABELLENKÜRZEL>_<BEZEICHNUNG>_I
- Trigger
  - SFA_<TABELLENKÜRZEL>_<TRIGGERTIME><SCOPE>_<ACTION>_TRG
- Sequenzen
  - SFA_<TABELLENKÜRZEL>_SEQ

Namenskonventionen

Datenbankobjekte

- Views
  - SFA_<VIEW_NAME>_V
  - SFA_<QUELLE>_LOV
  - SFA_P<SEITENNUMMER>_<VIEW_NAME>_V
  - SFA_P<SEITENNUMMER>_<QUELLE>_LOV
- Packages
  - SFA_<PACKAGE_NAME>_PKG oder SFA_P<SEITENNUMMER>_PKG

MT AG

business by integration

Vielen Dank.

MT AG

Balcke-Dürr-Allee 9
40882 Ratingen

Telefon: +49 (0) 21 02 309 61-0
Telefax: +49 (0) 21 02 309 61-101

E-Mail: info@mt-ag.com
www.mt-ag.com