

# Automation Middleware



# Inhaltsverzeichnis

- Team
- Was ist AMW?
- Warum AMW?
- Realisierung, Demo
- Ausblick

# Team

- Puzzle ITC
  - Thomas Philipona: PL
  - Oliver Schmid: Entwickler
  - Antonio Romano: Entwickler
  - Beat Schwaller: Entwickler, Scrum Master
- Mobiliar
  - Nick Müller: PL, Team Chef Middleware
  - Thomas Bien: Anforderungen
  - Yves Peter: Anforderungen, Product Owner

# Was ist AMW?

- AMW = **Automation MiddleWare**
- Von der Mobi bei Puzzle in Auftrag gegeben
- Tool zur Automatisierung von:
  - Serveraufbau
  - Applikationsverteilung
  - Konfigurationsverteilung
- Kombination von CMDB und Chef/Puppet
- JEE Web-Applikationen

# Warum AMW?

## Mobi Java Landschaft

- Total 15 Umgebungen
- Pro Applikation und Umgebung ein Linux / Applikationsserver
- Ca. 100 Applikationen auf rund 1500 Servern
- Regelmässige grosse Releases
- Technische Änderungen über Umgebungen hinweg
- ➔ **Riesige Masse unterschiedlicher Konfigurationen**

# Momentane Lösung

- Konfiguration wird im SVN abgelegt
- Konfiguration besteht aus Property- und Konfigurationsdateien
- Scripts für Bau und SSH für Verteilung
- Separates Tool für die Installation der Applikation (EAR, WAR, etc.)

# Ziele AMW

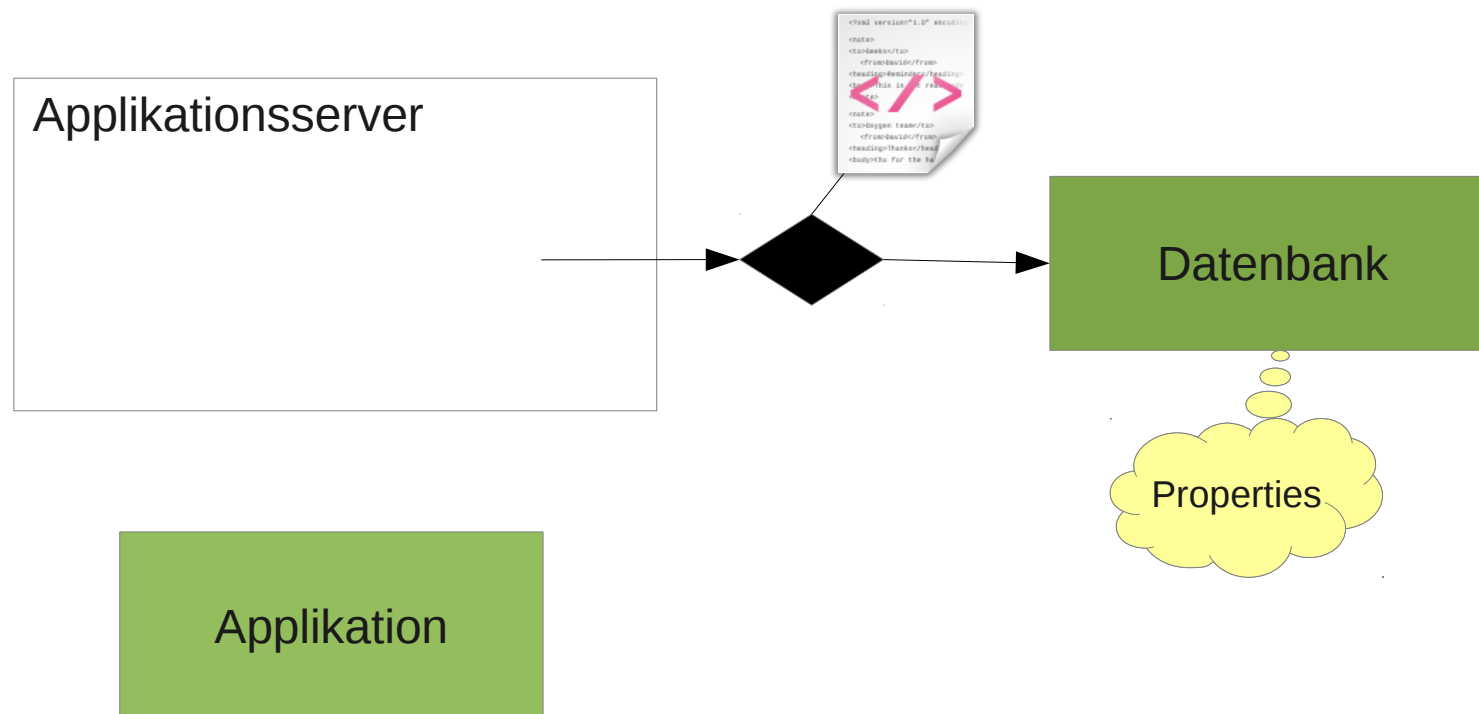
- Redundanzen von Konfigurationen vermeiden
- Abbilden von Service Abhängigkeiten
- Versionierung der Konfiguration
- Konfiguration serverunabhängig verwalten
- Nur ein Tool für Konfiguration und Applikations-Deployment
- Deployment von Applikation und Konfiguration kann vom Umgebungsmanagement gemacht werden

# Use Case

- Neuentwicklung: Web-Applikation
  - Applikation soll auf einem JBoss 5 installiert und mit einer Oracle DB verbunden werden
  - Applikation soll auf 2 Entwicklungs-Umgebungen (A und B) sowie einer Produktiven Umgebung (P) installiert werden
- Vorgehen mit AMW:
  - Applikation mit Ressourcen erfassen und abbilden
  - Umgebungsspezifische Konfigurationen generieren und auf unterschiedliche Umgebungen deployen



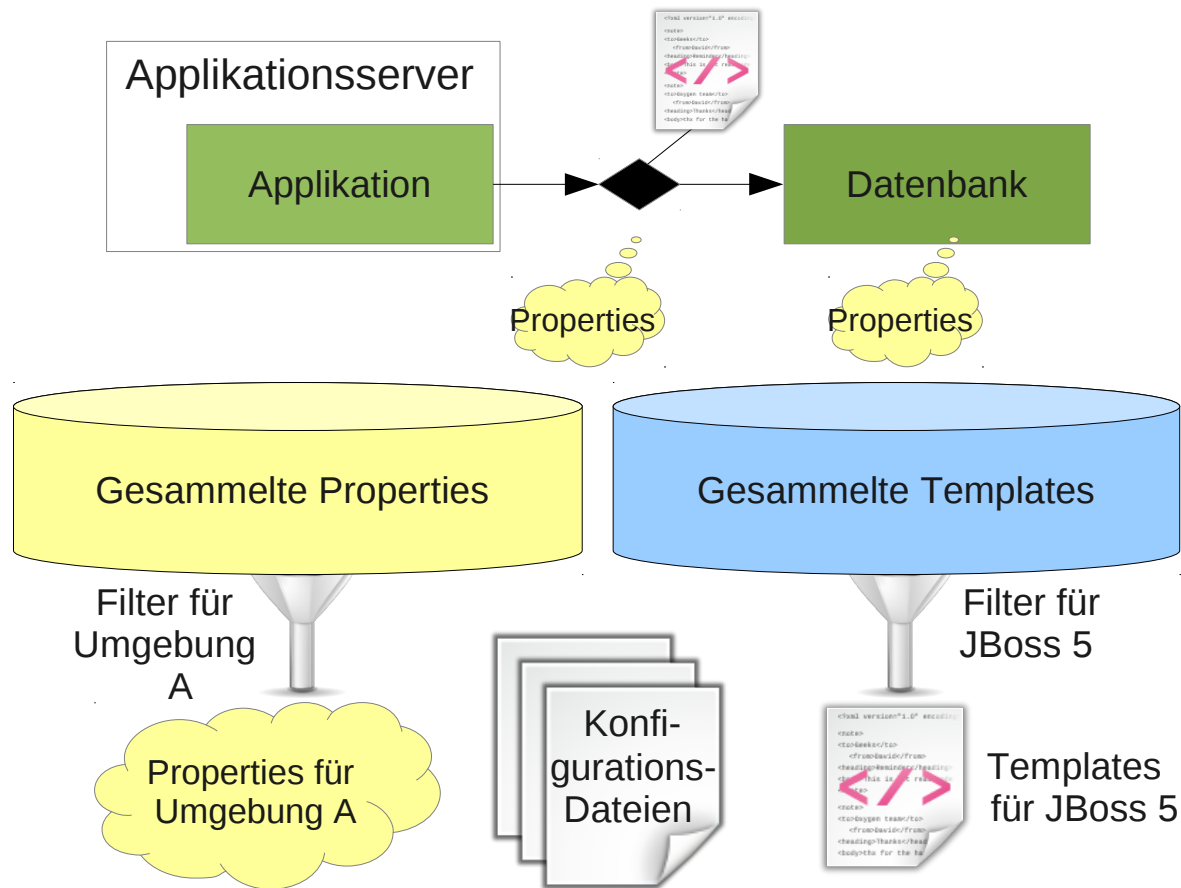
# Applikation erfassen



# Deployment-Job

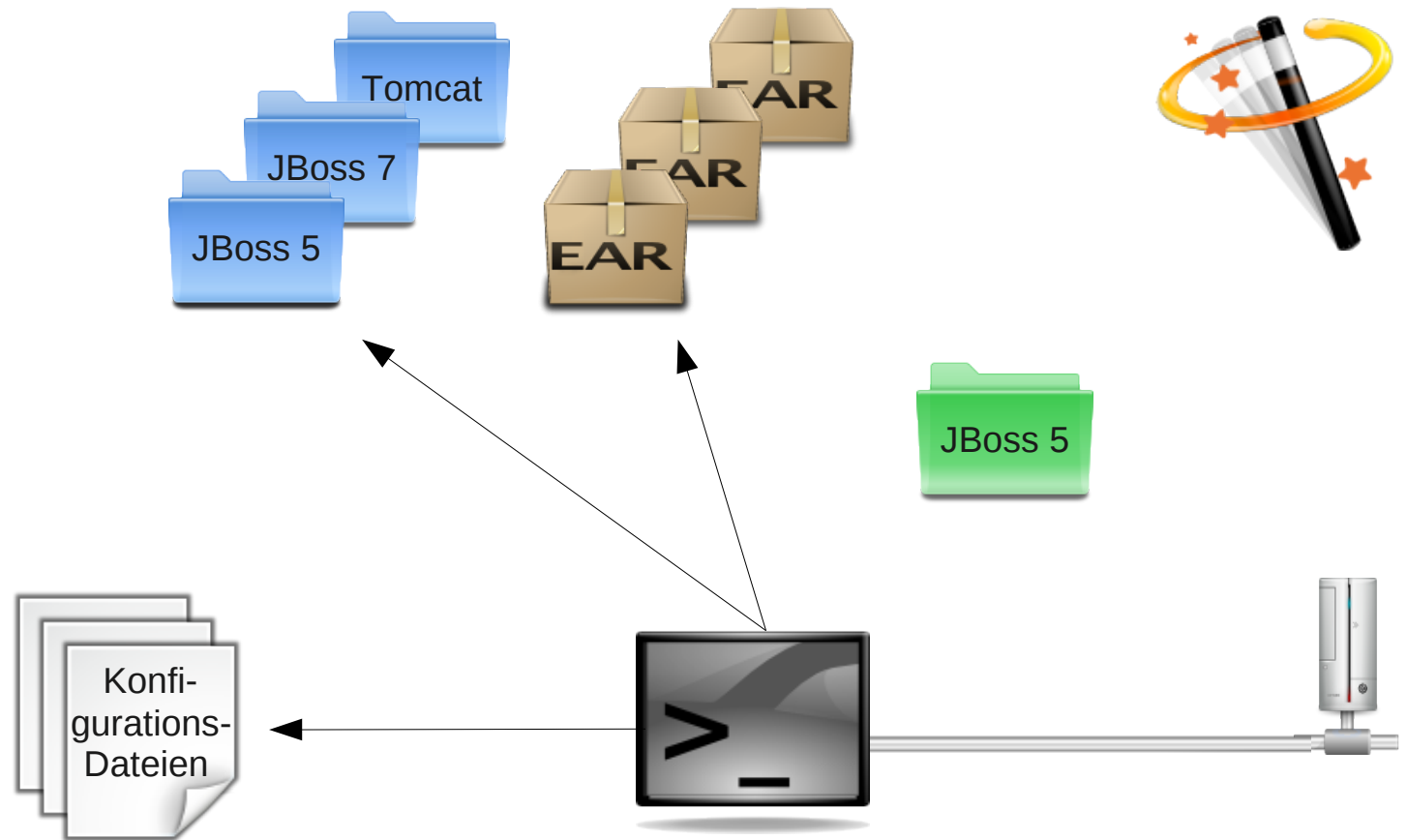
- Startet Generierung der Konfigurationsdateien sowie das Deployment auf einen Host
- Ermöglicht die Planung von Deployments
- Definiert den zu Deployenden Server sowie die Ziel-Umgebungen

# Generierung der Konfiguration



450		progress	Applikationsserver Applikation	default	A
-----	--	----------	-----------------------------------	---------	---

# Ausführung des Deploy-Skripts



450


**success**
**Applikationsserver**  
 Applikation

default

A

# Zusatz-Features

- Rollengesteuerte Datenerfassung und Anzeige
- Deployment-Verwaltung
- Versionierung

# Ausblick

- Mobi: Migration auf AMW
- Infrastruktur Shakedown Test
- Weiteres:
  - Vergleich zwischen Soll und Ist Konfiguration
  - Code veröffentlichen möglich (Open Source)
  - Bestellungsverwaltung
  - Anbindung an Linux Automatisierung
  - Anbindung an CMDB direkt/indirekt