



TECHNISCHE  
UNIVERSITÄT  
DRESDEN

# Ein Prototyp zur zustandsorientierten Instandhaltung

---

Jakob Krause, Sebastian Cech

*TU Dresden - Fakultät Informatik*

1. Motivation
2. Der Prognoseprozess
3. Die Frameworkarchitektur
4. Implementierung des Prototypen
5. Zusammenfassung und Ausblick


## Maschine

### Beanspruchungen:

- Biologisch
  - Chemisch
  - Physikalisch
  - Mechanisch
- 



### Wirkungen:

- Verschleiß
  - Verformung
  - Ermüdung
  - Korrosion
  - Brüche
  - Verschmutzung
  - usw.
- 

Maschinenausfälle

## Folgen von Maschinenausfällen:

- Stillstandzeiten in der Produktion
- Überschreitung von Lieferfristen
- Entstehung von Kosten

## Konventioneller Lösungsansatz:

- Präventive Instandhaltung unabhängig vom realen Zustand der Maschinen (periodische Wartung)

## Probleme:

- Unnötige Instandsetzung (zu früh)
- Maschinenausfälle (zu spät)
- Daraus resultierende Kosten

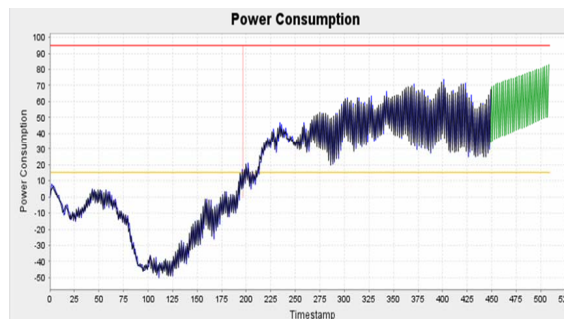
## Lösungsansatz: Zustandsorientierte Instandhaltung

Maschinen/Prozess



Prozessdaten

Modellierung und Prognose  
des Maschinen-Verhaltens



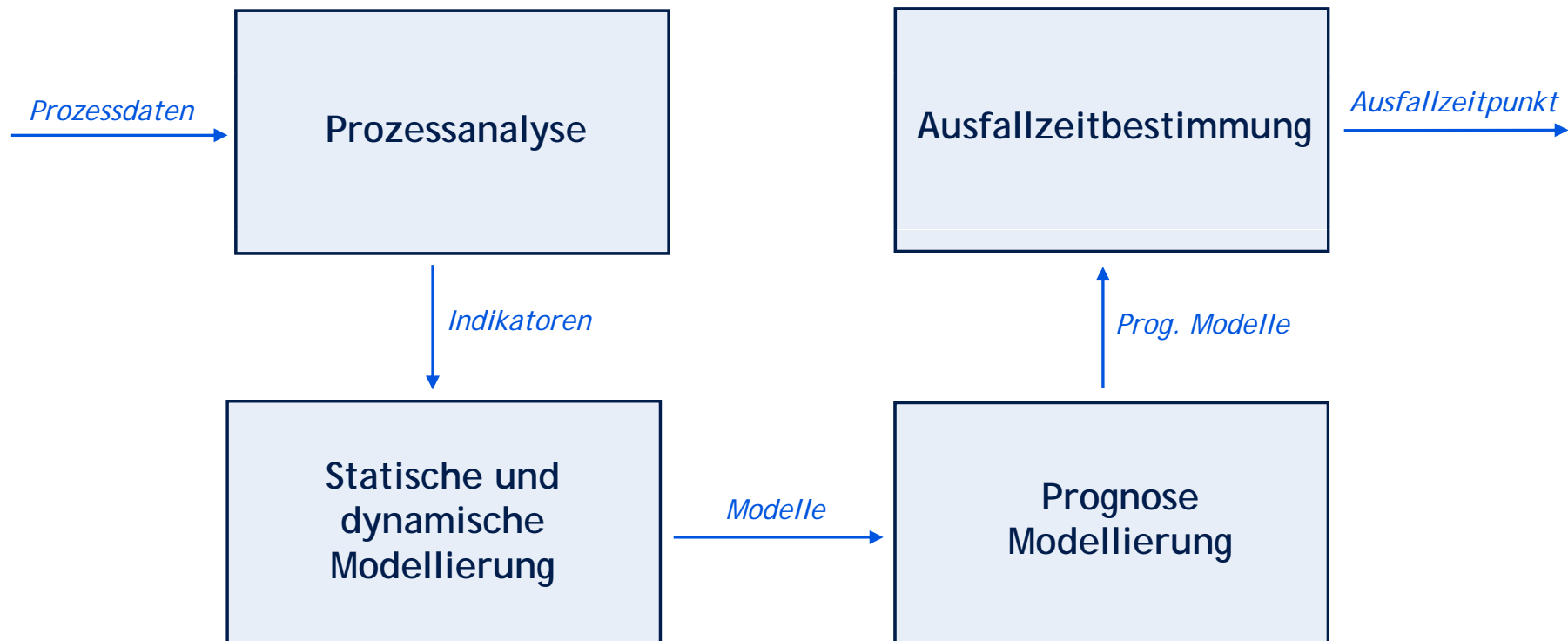
Erwartete  
Ausfallzeit

Planung von  
Wartungsarbeiten



Wartung

1. Motivation
2. Der Prognoseprozess
3. Die Frameworkarchitektur
4. Implementierung des Prototypen
5. Zusammenfassung und Ausblick



## 1. Prozessanalyse

- Analyse des Maschinenprozesses durch Spezialisten
- Aufdeckung von Zusammenhängen in den Prozessgrößen
- Identifizierung von Zuverlässigkeits-Indikatoren\*
- Bestimmung von Grenzwerten für Zuverlässigkeits-Indikatoren

## 2. Statische und dynamische Modellierung

- Erstellung von statischen Modellen (Zusammenhang zwischen Ein- und Ausgangsgrößen eines Systems)
- Erstellung von dynamischen Modellen (Verhalten eines Indikators über die Zeit)
- Validierung der Modelle

\*Indikatoren: Prozessdaten die den Alterungsprozess (Zuverlässigkeit) von Maschinen widerspiegeln



## 3. Prognose Modellierung

- Erstellung von Prognosemodellen basierend auf dynamischen Modellen
- Erstellung der Prognose
- Validierung der Prognosemodelle

## 4. Ausfallzeitbestimmung

- Identifizierung von Zustandswechseln (Bereichen in denen die Prognose festgelegte Zuverlässigkeits- Grenzwerte überschreitet)

## Probleme:

- Starke Unterschiede in den Eigenschaften der Prozessdaten
- Dynamische Änderung der Prozessdaten-Eigenschaften
- Stetig zunehmende Datenmenge

## Zusammengefasst:

- Forderung nach flexibler Anpassung des Modellierungsprozesses an die Prozessdaten-Charakteristika

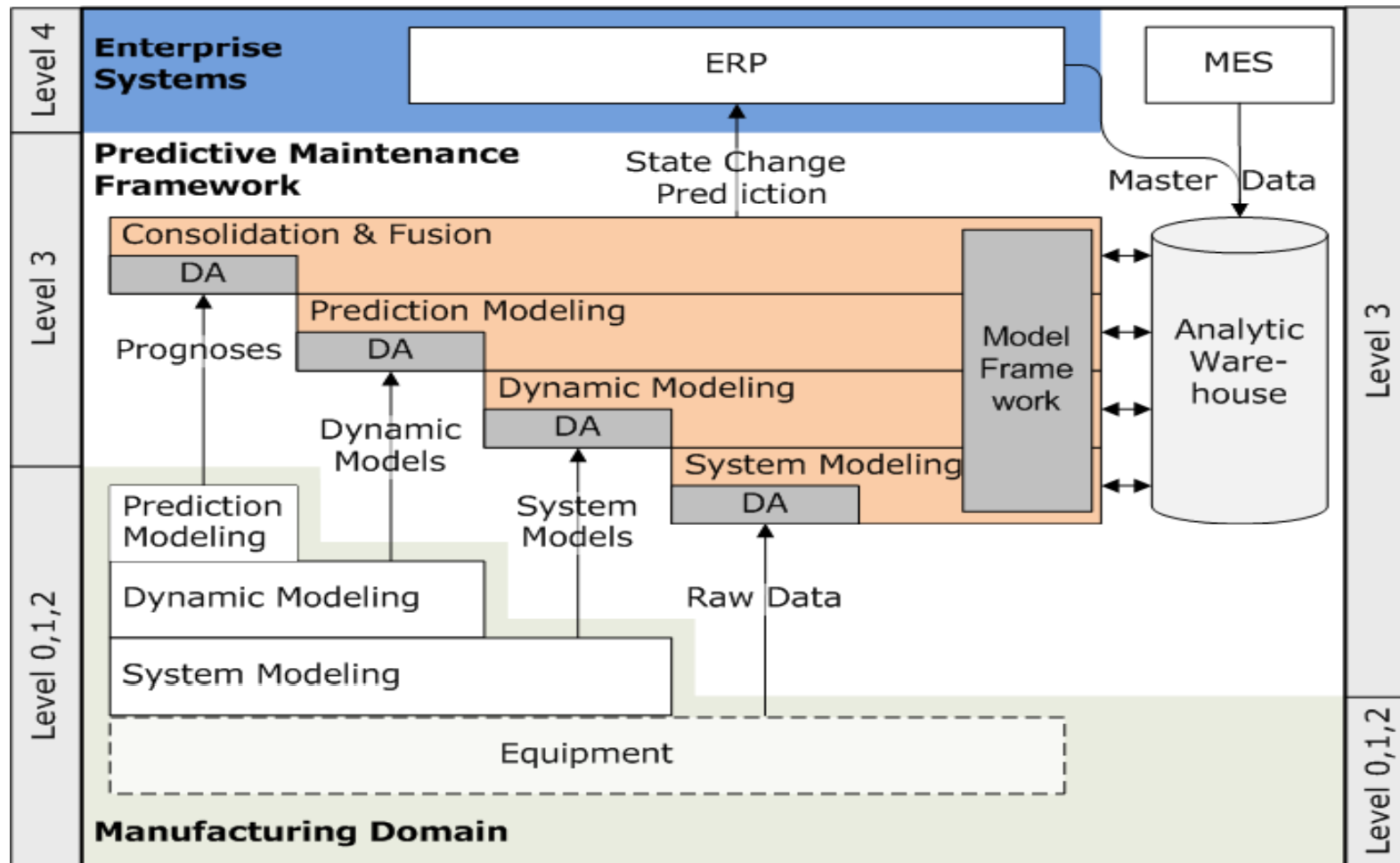
## Lösung:

Framework für zustandsorientierte Instandhaltung

1. Motivation
2. Der Prognoseprozess
3. Die Frameworkarchitektur
4. Implementierung des Prototypen
5. Zusammenfassung und Ausblick

## Aufbau:

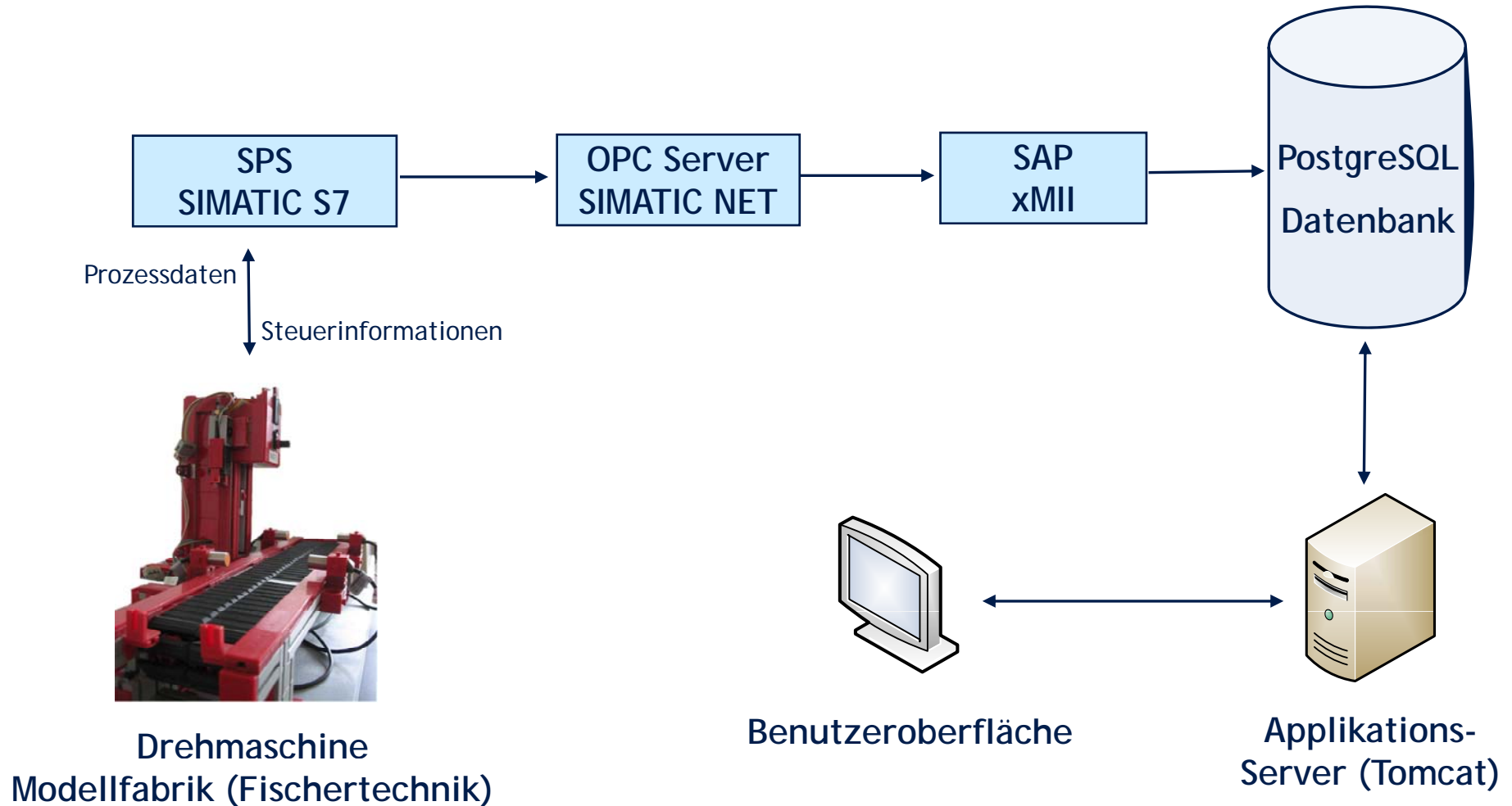
- Realisierung des Model-Frameworks in Schichten:
  - Statische Modellierung
  - Dynamische Modellierung
  - Prognose Modellierung
  - Konsolidierung
- Anbindungsmöglichkeit auf jeder Schicht zur Datenakquise (DA)
- Persistierung relevanter Daten im **Analytic Warehouse** (operationale Datenbank)
- Anbindung an ERP - Systeme zur bedarfsorientierten Wartung
- Service Orientierte Architektur (Heterogenität unterschiedlicher Systeme)



Die Frameworkarchitektur

1. Motivation
2. Der Prognoseprozess
3. Die Frameworkarchitektur
4. Implementierung des Prototypen
5. Zusammenfassung und Ausblick

## Technische Basis:

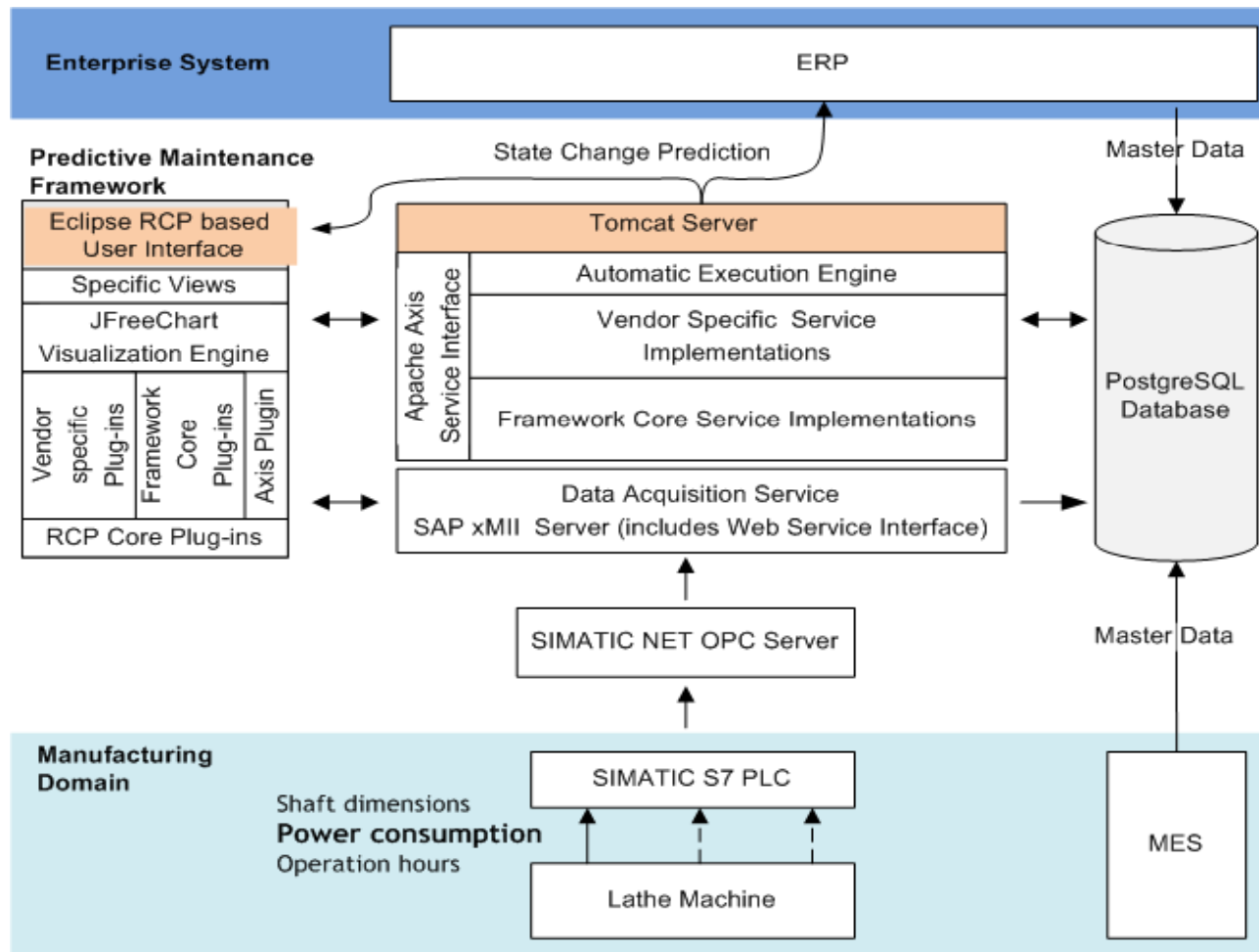


## Modellbildungs- und Prognose- Algorithmen im Modelframework: (Anpassung von Modellen an Zeitreihen und deren Prognose)

- Repräsentation als WebServices mit einheitlicher Schnittstelle  
Vorteile:
  - Flexible Erweiterung des Frameworks um neue Algorithmen und externe Werkzeuge (z.B.: X12-ARIMA)
  - Verteilte Bearbeitung und Verwaltung des Prognosevorgangs
- Derzeitiger Vorrat:

ARIMA	Polynom	Prognose
Maximum-Likelihood	Variate Differenzen	Lineare Kalman Prog.
Yule-Walker	Methode kl. Quadrate	
Kalman Filter		





Einbettung des Prototypes in die Frameworkarchitektur

## Benutzeroberfläche:

- Modulare Eclipse RCP Anwendung
- Realisierung von:
  - Mechanismen zum Datenzugriff
  - Mechanismen zur Visualisierung
  - Unterstützende Modellbildungsalgorithmen
- Zwei Sichten
  - System Integrator
  - Operator



Die Benutzeroberfläche

1. Motivation
2. Der Prognoseprozess
3. Die Frameworkarchitektur
4. Implementierung des Prototypen
5. Zusammenfassung und Ausblick

## Zustandsorientierte Instandhaltung:

- Kostengünstigere und effizientere Instandhaltung möglich
- Unterstützung der allwissenden Fabrik

## Prototyp:

- Modellierung von ausgewählten Indikatoren (z.B. Leistungsverbrauch)
- Prognostizieren des Ausfallzeitpunktes
- Periodische Modellaktualisierung

## Ausblick:

- Flexible Anbindung an ERP-Systeme
- Technische Anbindung sowie Modellierung weiterer Indikatoren
- Integration weiterer angepasster Modellierungsalgorithmen
- Test der Architektur in realer Produktionsumgebung



ENDE

Vielen Dank für Ihre Aufmerksamkeit!

---

Jakob Krause: [jk329036@inf.tu-dresden.de](mailto:jk329036@inf.tu-dresden.de)

Sebastian Cech: [sebastian.cech@inf.tu-dresden.de](mailto:sebastian.cech@inf.tu-dresden.de)

*Das zu Grunde liegende Projekt wird finanziert von SAP Research*