



# Kopplung heterogener Prozesssimulationen mit Hilfe einer durchgehenden Datenverwaltungsplattform

Tamás Farkas, Nilesh Thakare, Jan Erik Menzler, Tobias Kleinert, Herbert Pfeifer

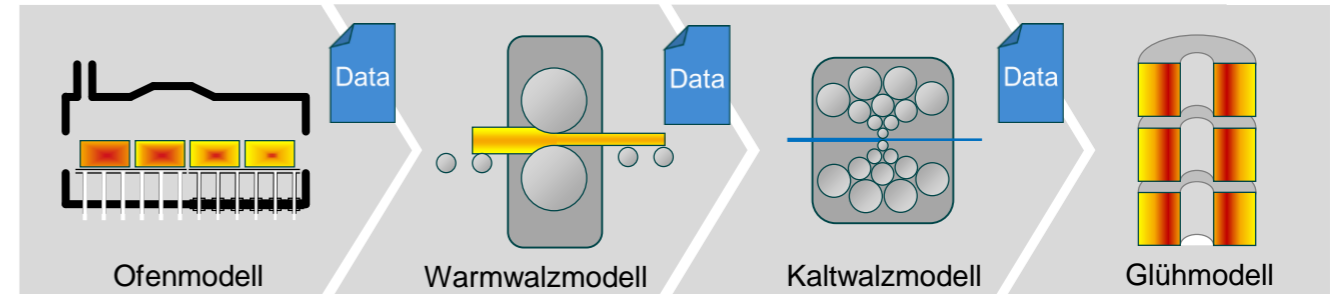
# Inhalt

---

- Motivation
- Datenverwaltungs- und Simulationsplattform
  - Programmierschnittstelle
  - Datenhaltung
  - Simulationen
- Use-Cases
- Zusammenfassung

# Motivation

- Heterogene Produktionsschritte
  - Prozessmodelle bilden einzelne Schritte ab
- Kein automatisierter Datenaustausch
  - Aufwendig, erfordert Expertenwissen

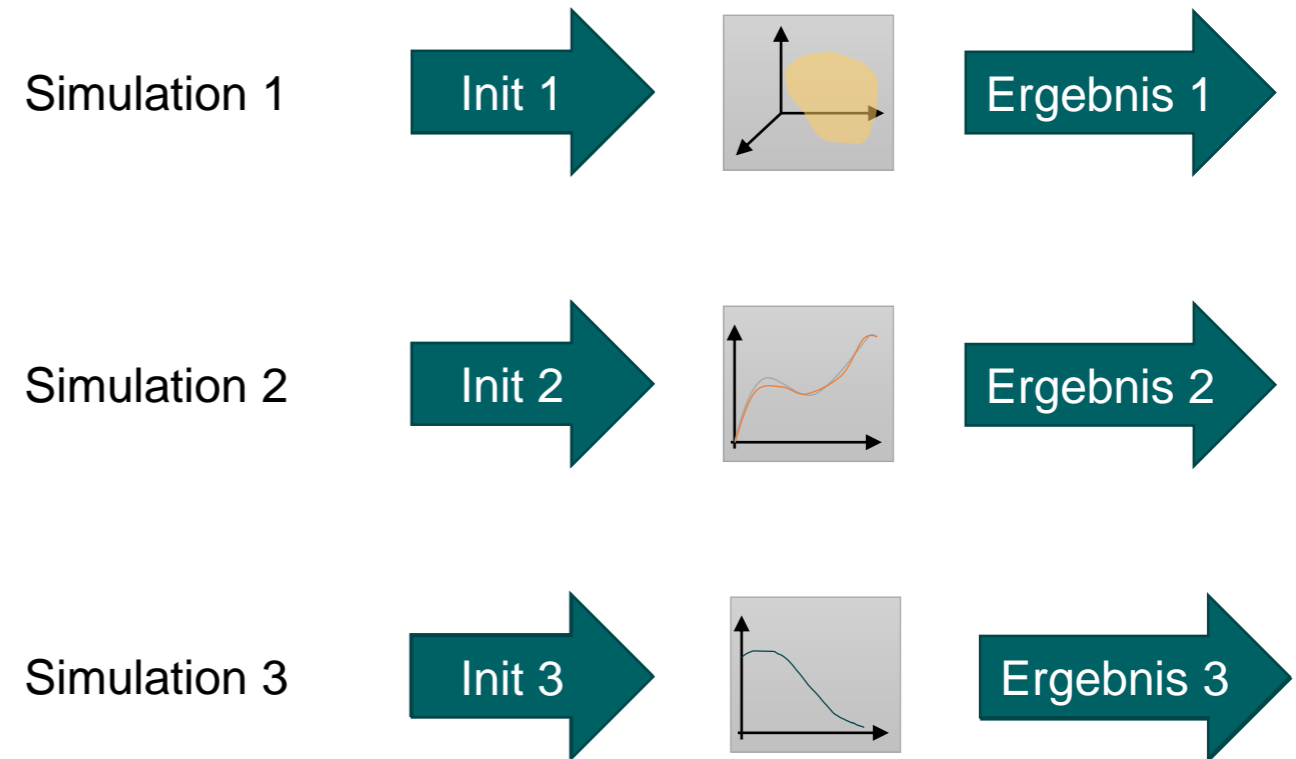


## Lösungsansatz:

- Analyse bestehender Modelle
- Modellanpassung / -erweiterung
- Aufbau einer Datenverwaltungs- und Simulationsplattform, um die Datenaustausch zu ermöglichen
- Einheitliches Execution Environment für Prozessmodelle
- Modellvalidierung an industriellen Use Cases

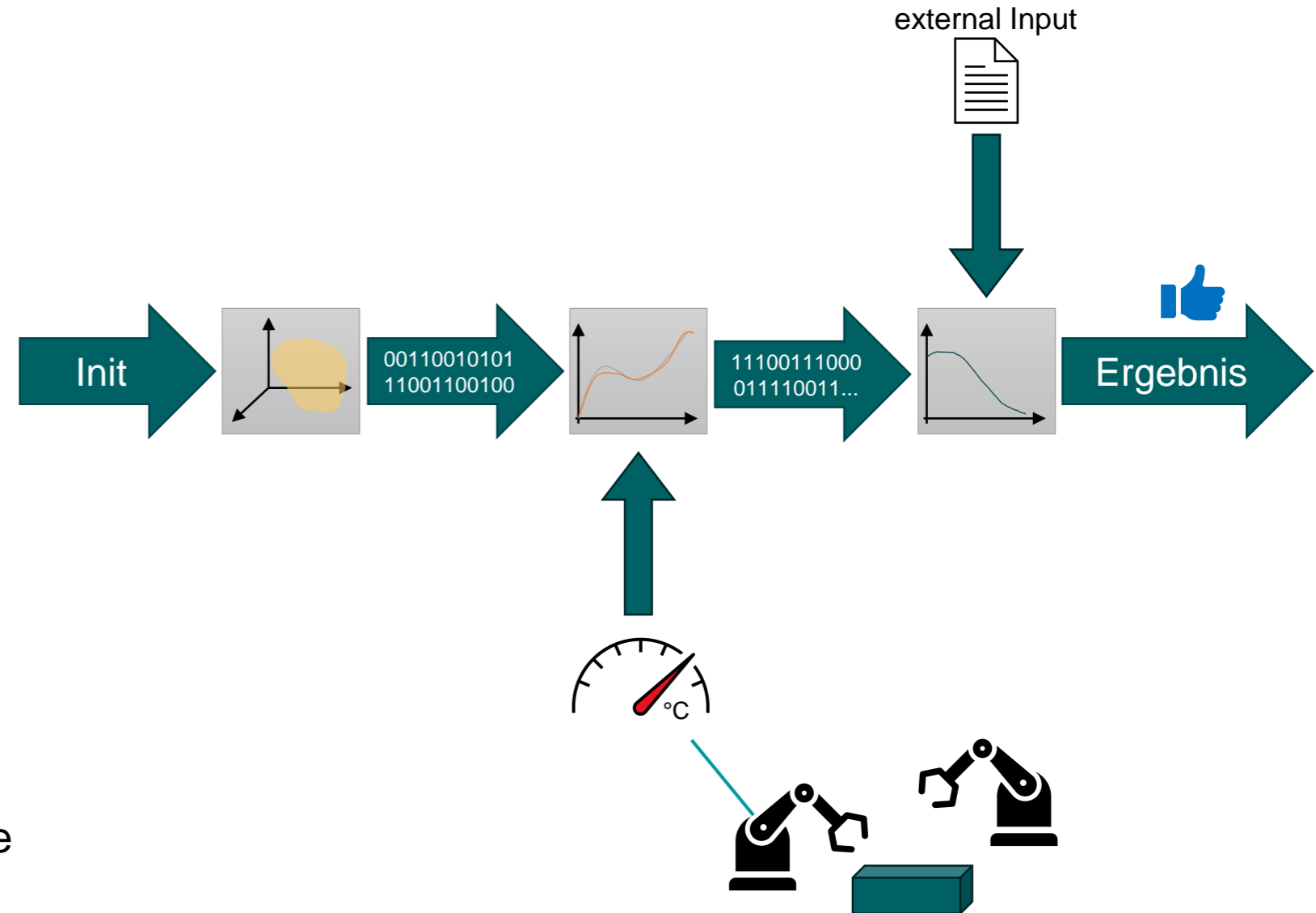
# Datenverwaltungs- und Simulationsplattform

- Eine **Simulation** besteht aus:
  - einem ausführbaren Programm
  - Eingang
  - Ausgang
- Keine einheitliche Struktur
  - Verschiedene Betriebssysteme
  - Datenformate
- Simulationskopplung



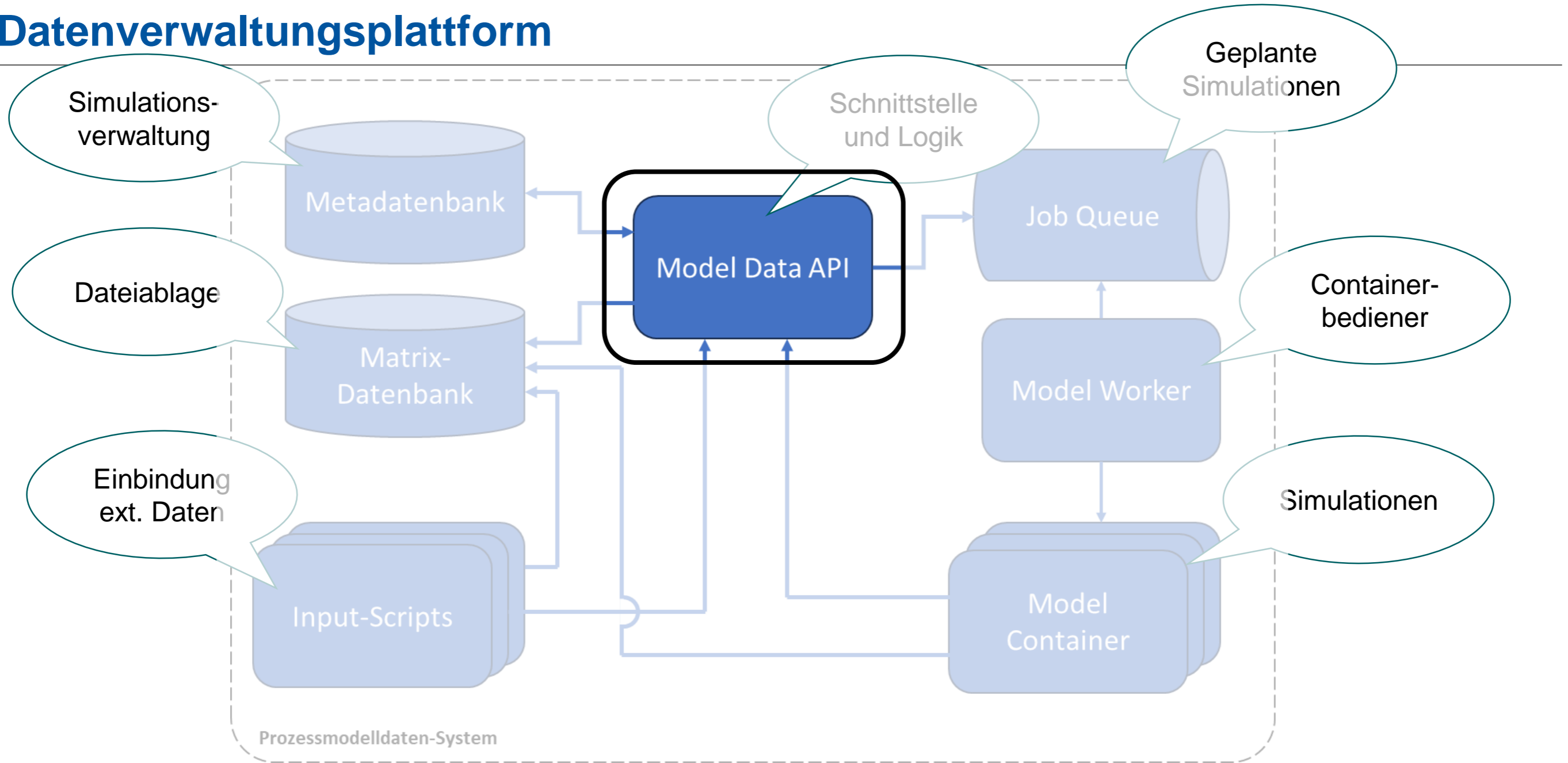
# Datenverwaltungs- und Simulationsplattform

- Eine **Simulation** besteht aus:
  - einem ausführbaren Programm
  - Eingang
  - Ausgang
- Keine einheitliche Struktur
  - Verschiedene Betriebssysteme
  - Datenformate
- Simulationskopplung
  - Simulationsergebnisse werden Inputs
  - Einbeziehung von Messungen
  - Verbesserung von Simulationsergebnisse durch genauere Inputs
  - Automatisierte Simulationsausführung



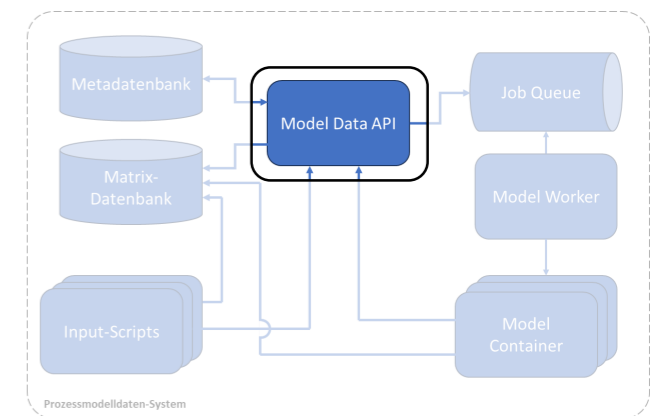


# Datenverwaltungsplattform



# Programmierschnittstelle

- Python Flask Applikation
- REST-API (Representational State Transfer - Application Programming Interface)
  - zustandslos
  - HTTP-Protokoll
  - JSON-Nachrichtenformat
- API-Endpoints
  - Zum Hinzufügen und Abrufen von Datensatzeinträgen
  - GET/POST Requests
    - `http://{base_url}/api/v1/{endpoint}/{querystring}`
      - `{base_url}`: localhost:5000
      - `{endpoint}`: *dataset, simulator, input, step*
      - `{querystring}`: optional, bietet erweiterte Funktionen z. B. Filterung, Zuordnung
  - HTTP-Responses: 200 OK, 409 Conflict, 422 ValidationError ...



- Keine Speicherung in der API

→ Verwendung von Datenbanktechnologien

## MongoDB

- NoSQL-Datenmanagementsystem
- Metadaten
- Speicherung einzelner Ergebnisse
- JSON-LD (JavaScript Object Notation for Linked Data)

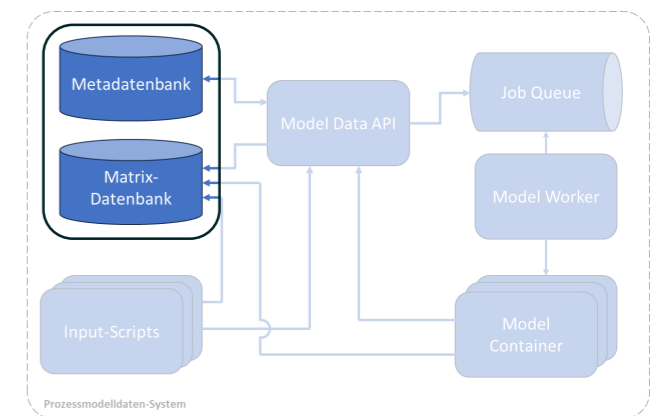
```
"sosa:observedProperty": {
  "@type": ["qudt:Quantity", "sosa:ObservableProperty"],
  "qudt:hasQuantityKind": [{"@id": "cpmProcess:atmosphereTemperature"}],
  "rdfs:label": "Temperature in the oven 04",
  "ssn:isPropertyOf": {
    "@id": "http://acplt.org/individual/gpkm/factory/productionUnits#oven04",
    "@type": "cpm:ProductionUnit"
  }
},
"cpm:relativeAccuracy": 0.01,
"sosa:hasResult": {
  "@type": ["sosa:Result", "qudt:QuantityValue"],
  "qudt:numericValue": 25.5,
  "qudt:unit": {"@id": "unit:DEG_C"}
},
```

Art der Messung

Wert

## MinIO

- Objektspeicher
- Beliebige Dateiformate
- Speicherung mehrdimensionaler Daten (Matrizen)
- HDF5 (Hierarchical Data Format)





# Simulationen

- Einheitliche Ausführungsumgebung: Docker (wenn möglich)

## Wrapper library



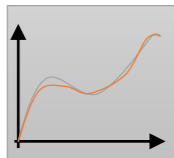
- Upload und Download von Prozess- und Simulationsdaten
- Generierung der Datenstruktur
- Automatische Einheitskonvertierung

## Wrapper scripts



- Konfigurieren und Starten der Simulationen
- Kommunikation mit der API der Plattform

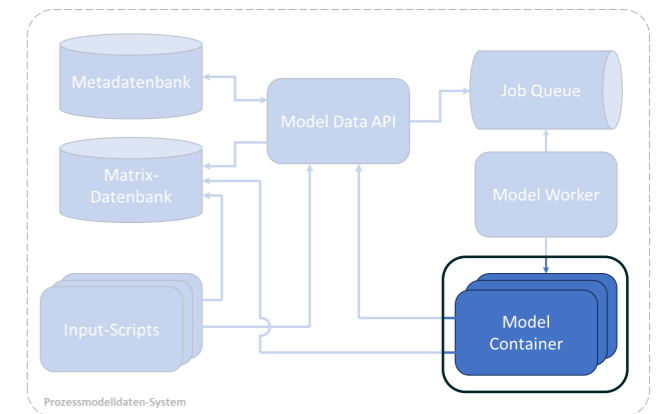
## Simulation



## Docker container



- Ausführbares Softwarepaket, das alle Elemente zur Durchführung einer Simulation enthält
- Mehrere Container können gleichzeitig laufen
- Kapselt die Inhalte

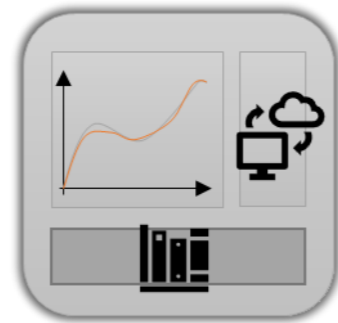


# Simulationen

---

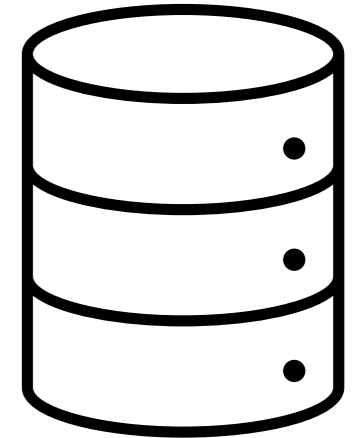
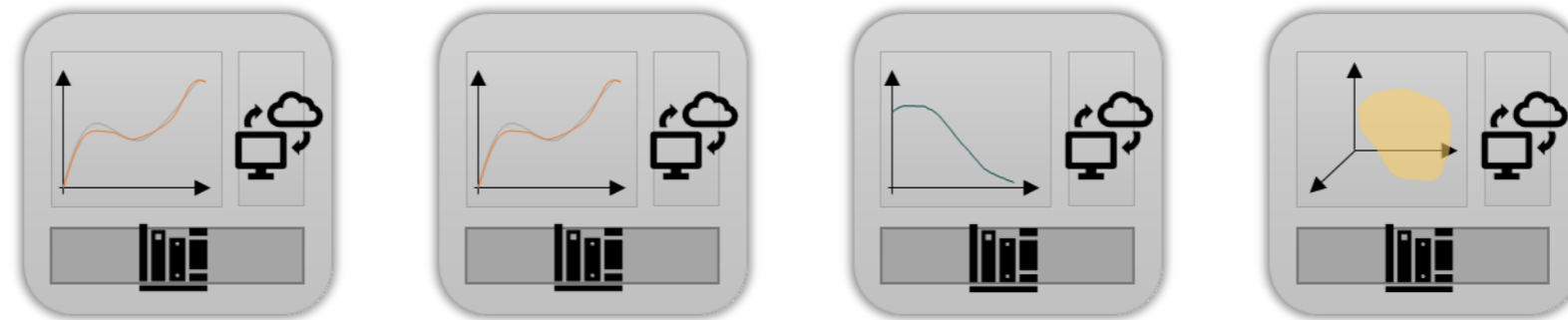
- Einheitliche Ausführungsumgebung: Docker (wenn möglich)

Simulationscontainer



# Simulationen

- Einheitliche Ausführungsumgebung: Docker (wenn möglich)

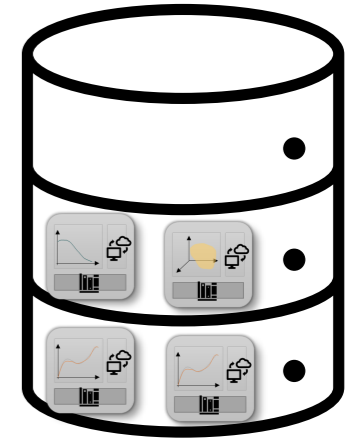


Docker Registry

# Simulationen

---

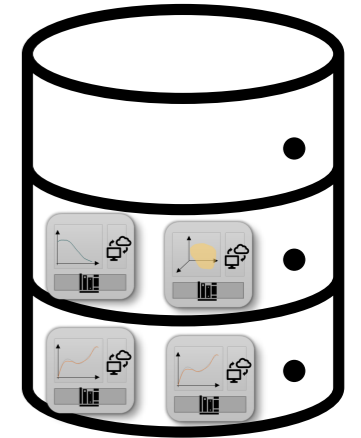
- Einheitliche Ausführungsumgebung: Docker (wenn möglich)
- Hochladen in lokales Docker Registry



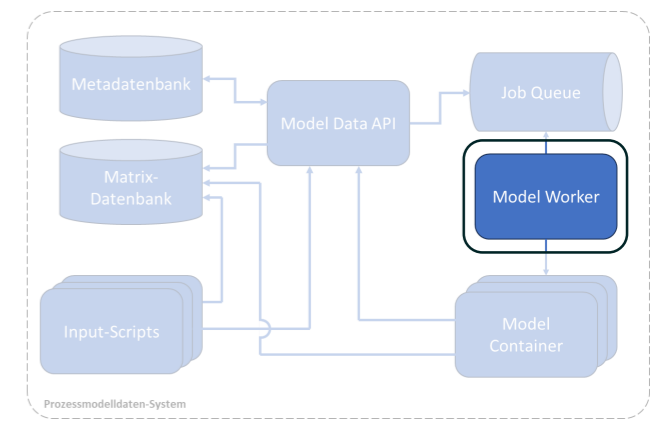
Docker Registry

# Simulationen

- Einheitliche Ausführungsumgebung: Docker (wenn möglich)
- Hochladen in lokales Docker Registry
- Die Container werden von dem RQ-Worker<sup>1</sup> ausgeführt



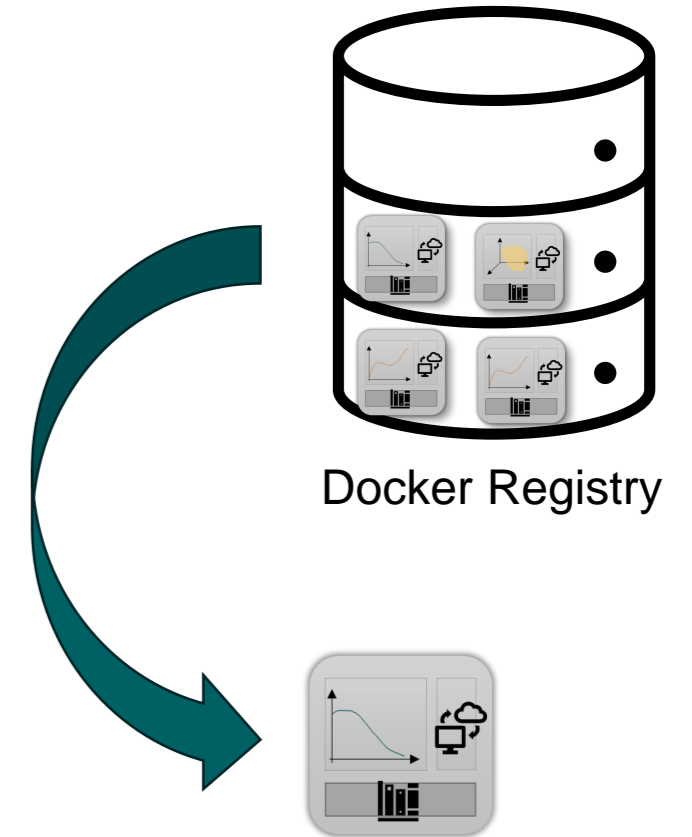
Docker Registry



<sup>1</sup> <https://python-rq.org/docs/workers/>

# Simulationen

- Einheitliche Ausführungsumgebung: Docker (wenn möglich)
- Hochladen in lokales Docker Registry
- Die Container werden von dem RQ-Worker<sup>1</sup> ausgeführt
  - Pull Container

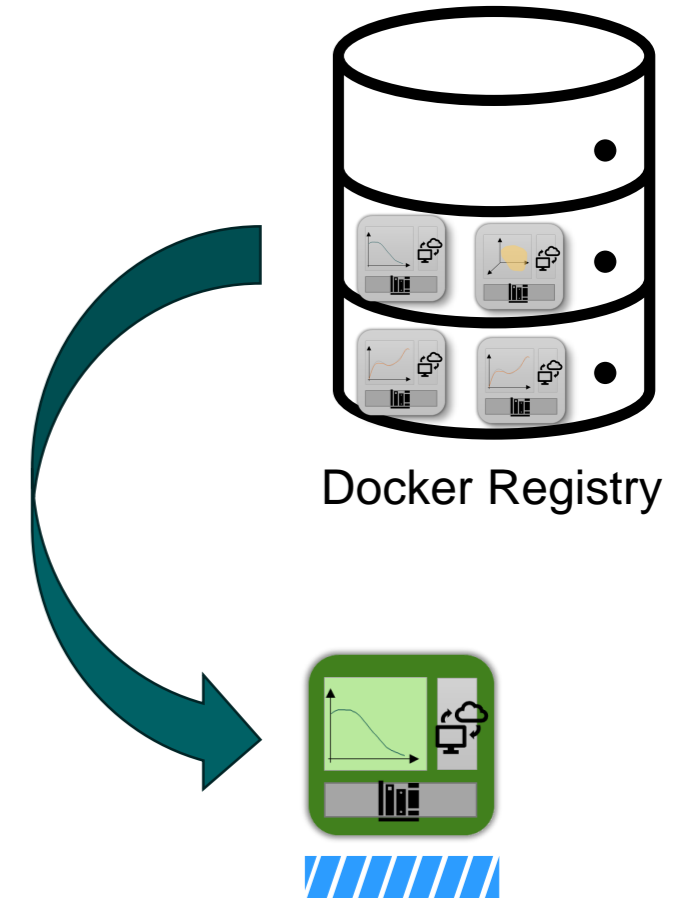


<sup>1</sup> <https://python-rq.org/docs/workers/>



# Simulationen

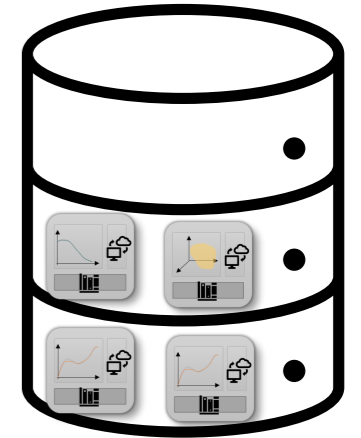
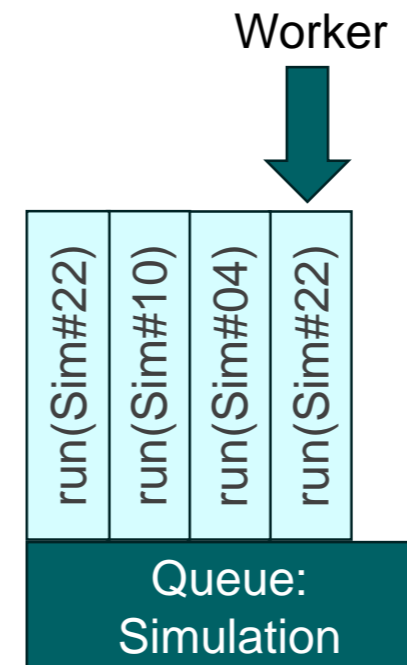
- Einheitliche Ausführungsumgebung: Docker (wenn möglich)
- Hochladen in lokales Docker Registry
- Die Container werden von dem RQ-Worker<sup>1</sup> ausgeführt
  - Pull Container
  - Nächste Container in Redis-Queue starten



<sup>1</sup> <https://python-rq.org/docs/workers/>

# Simulationen

- Einheitliche Ausführungsumgebung: Docker (wenn möglich)
- Hochladen in lokales Docker Registry
- Die Container werden von dem RQ-Worker<sup>1</sup> ausgeführt
  - Pull Container
  - Nächste Container in Redis-Queue starten
  - Wenn *depends\_on* Bedingung aktiv, warten auf erfolgreiche Containerausführung
- Upload und Download von Datensätze im Container

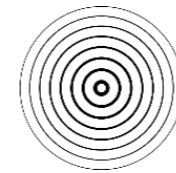


Docker Registry



<sup>1</sup> <https://python-rq.org/docs/workers/>

# Aluminiumdemonstrator - Speira & AluNorf



speira

**ALUNORF**

---

Qualitätsverbesserung durch Reduzierung der Zipfelbildung bei der Aluminiumdosenproduktion

**AF IGF**

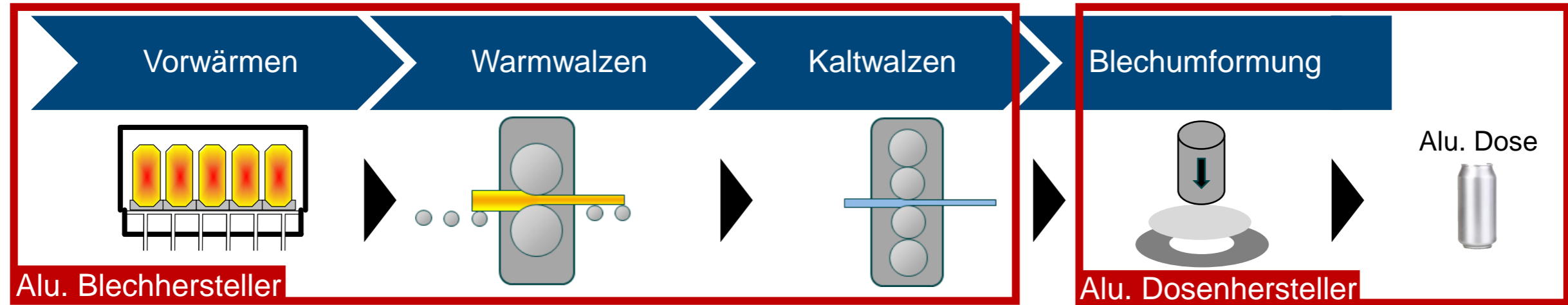
**IAT**

**ibf**

**IOB**

**RWTHAACHEN  
UNIVERSITY**

## Prozesskette der Aluminiumdosen Herstellung



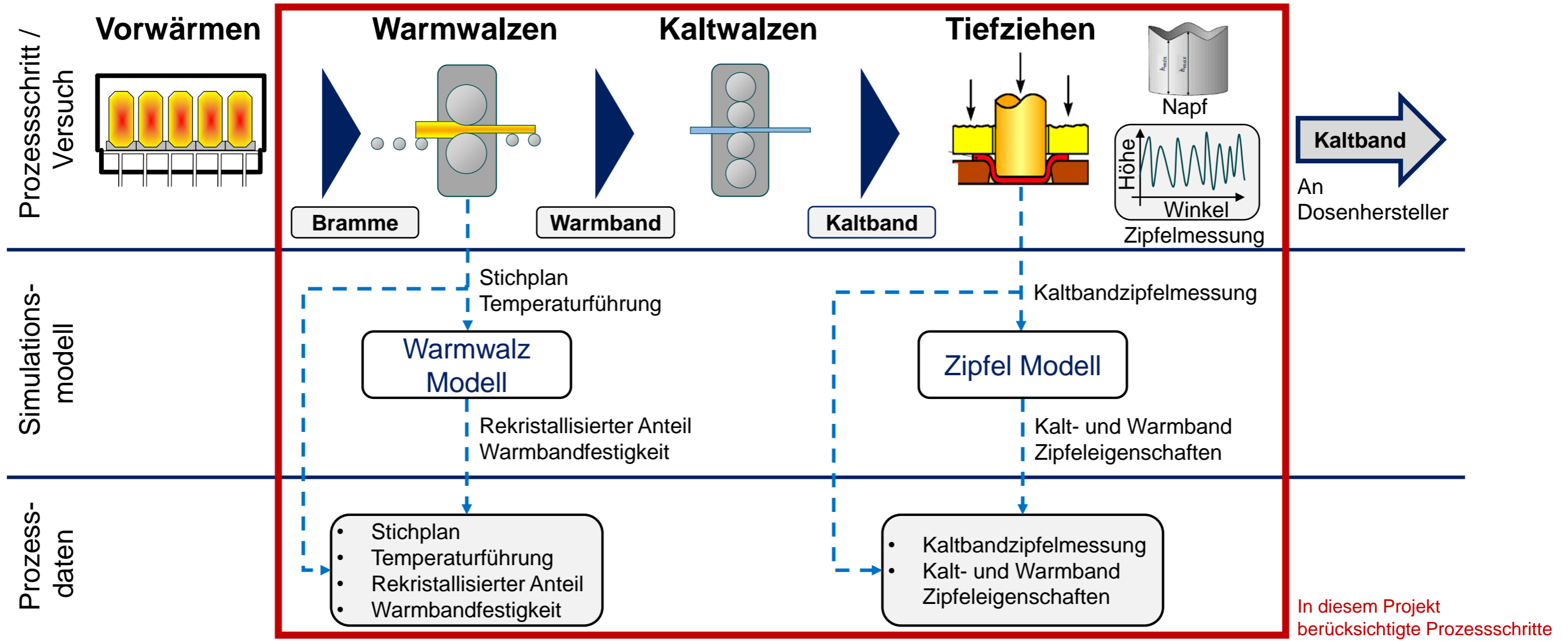
Prozesskette der Aluminiumdosenproduktion



Zipfelbildung bei der Aluminiumdosenproduktion [Speira]

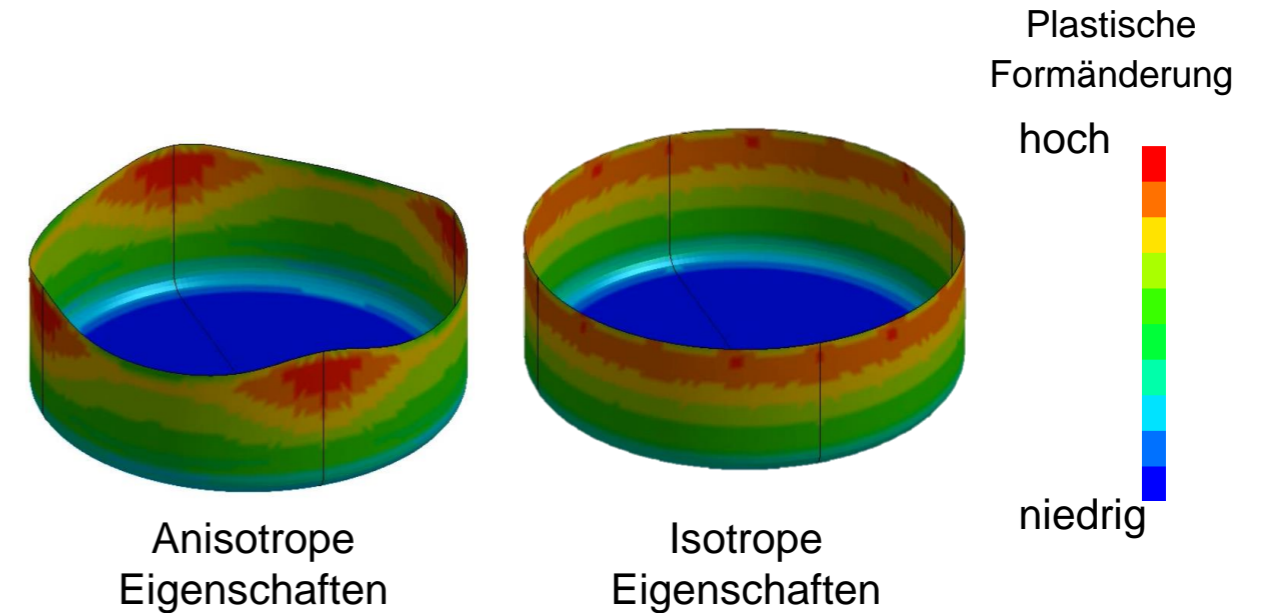
# Aluminiumdemonstrator – Speira & AluNorf

## Prozesskette der Aluminiumblech Herstellung



## Motivation:

- **Zipfelbildung** aufgrund des anisotropen Materialverhaltens
    - Mikrostruktur nicht korrekt eingestellt
    - Unvollständige Rekristallisation
    - Walztextur durch Warmwalzprozess
    - Ausscheidungen
  - **Entkoppelte Simulationsmodelle**
    - Unterschiedliche Datenformate
    - Simulation der kompletten Prozesskette zeit- und kostenaufwändig
- Erzeugung von Ausschuss
- Wirtschaftlichkeit der Prozesskette sinkt



Einfluss des anisotropen Materialverhaltens auf die Zipfelbildung



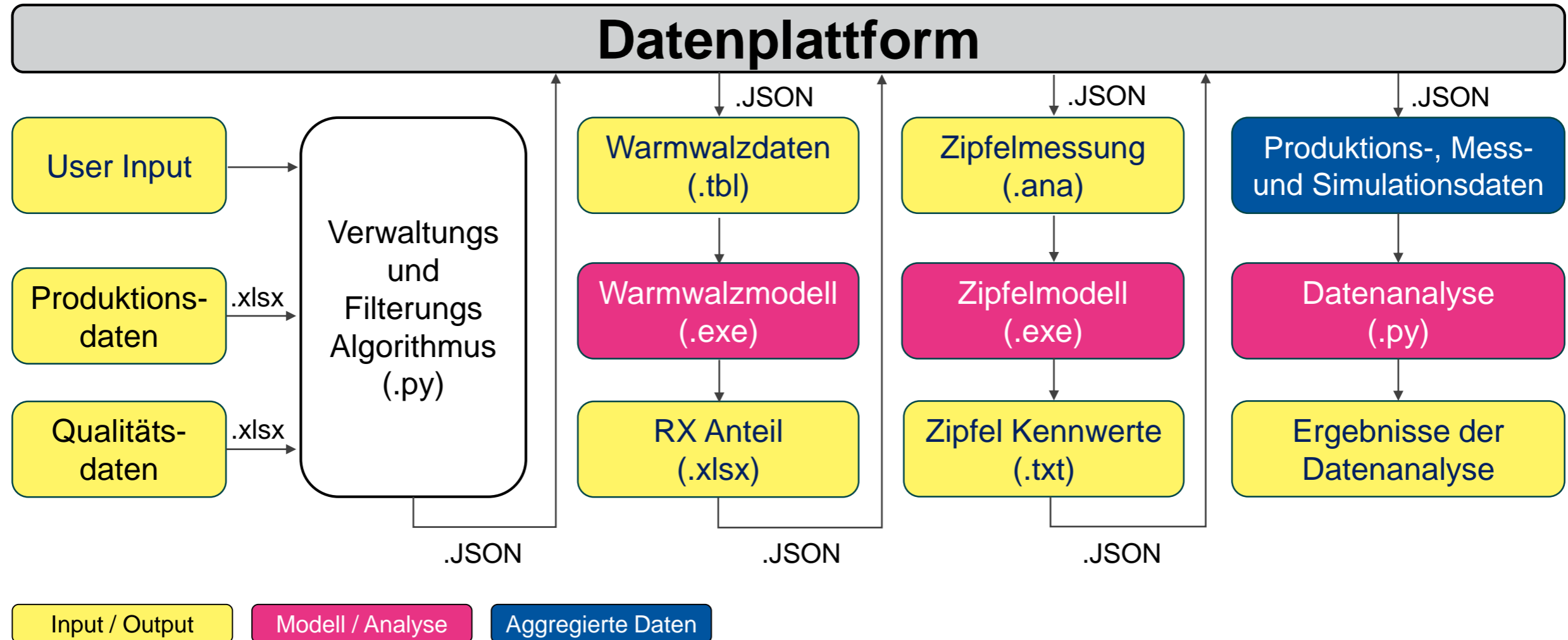
## Ziel:

- Minimierung der Zipfelbildung durch Optimierung des Warmwalzprozesses

## Vorgehen:

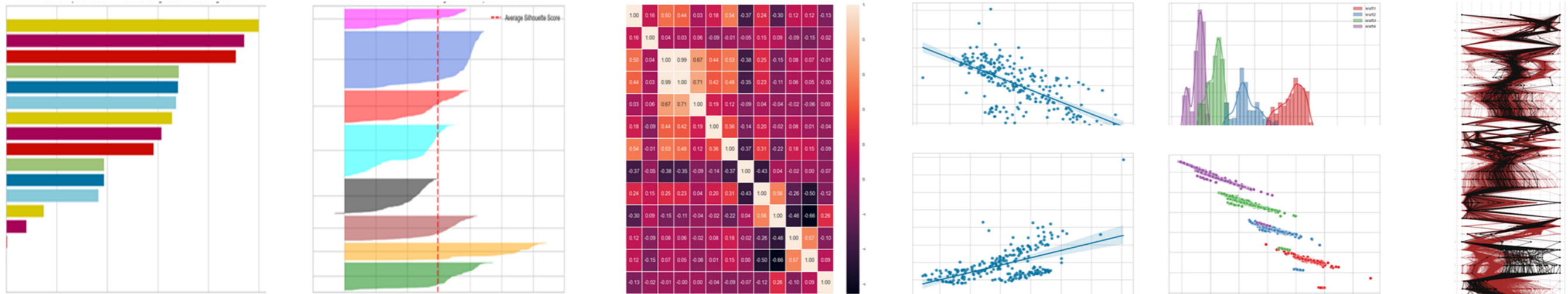
- **Kopplung der Simulationsmodelle über die Datenverwaltungsplattform**
  - Definition relevanter Eingabe- und Ausgabeparameter und Datenformat für die Modelle
  - Erweiterung der Warmwalz- und Zipfelmodelle
  - Validierung der Simulationsmodelle
- **Analyse der aggregierten Daten zur Optimierung der Warmwalzparameter**
  - Bereitstellung der aggregierten Simulations- und Prozessdaten über Datenplattform
  - Analyse der Korrelation zwischen Zipfelbildung und Warmwalzparameter
  - Bestimmung der optimalen Warmwalzparameter

## Kopplung von Prozessdaten und Simulationsmodellen



## Datenanalyse zur Optimierung der Warmwalzprozessparameter

- Vorschläge zur Warmwalzparameter Optimierung
  - Factor Importance Analysis
  - Cluster Analysis
  - Correlation Analysis
  - Histogramm und Scatter Plots



## Datenverwaltungsplattform:

### ■ Aufbau der Plattform

- API
- Datenbanken
- Simulationscontainer und Simulationsausführung

## Aluminiumdemonstrator:

### ■ Kopplung der Modelle über die Datenverwaltungsplattform

- Erweiterung und Validierung der Warmwalz- und Zipfelmodelle

### ■ Datenanalyse

- Verwaltungs- und Filterungs-Algorithmus
- Korrelation zwischen Warmwalzprozessparametern und Zipfelbildung unter Verwendung von aggregierten Prozess- und Simulationsdaten
- Optimierung des Warmwalzprozesses



Gefördert durch:



Bundesministerium  
für Wirtschaft  
und Klimaschutz

aufgrund eines Beschlusses  
des Deutschen Bundestages

Thinking the Future  
Zukunft denken

# Industrielle Gemeinschaftsforschung

## **Tamás Farkas, M.Sc.**

Lehrstuhl für Informations- und Automatisierungssysteme  
für die Prozess- und Werkstofftechnik  
*RWTH Aachen University*  
*Turmstr. 46*  
*52064 Aachen*

Tel.: +49 (0) 241 80-97747

E-Mail: [t.farkas@iat.rwth-aachen.de](mailto:t.farkas@iat.rwth-aachen.de)

## **Nilesh Thakare, M.Sc.**

Institut für Bildsamer Formgebung  
*RWTH Aachen University*  
*Intzestr. 10*  
*52072 Aachen*

Tel.: +49 (0) 241 80-95945

E-Mail: [nilesh.thakare@ibf.rwth-aachen.de](mailto:nilesh.thakare@ibf.rwth-aachen.de)

## **Jan Erik Menzler, M.Sc.**

Institut für Industrieofenbau  
*RWTH Aachen University*  
*Kopernikusstr. 10*  
*52074 Aachen*

Tel.: +49 (0) 241 80-25944

E-Mail: [menzler@iob.rwth-aachen.de](mailto:menzler@iob.rwth-aachen.de)