

# Smart Sensors: Softwaregetriebene Innovation in der Sensortechnik

INDUSTRIEKOLLOQUIUM DATENTECHNIK 2025

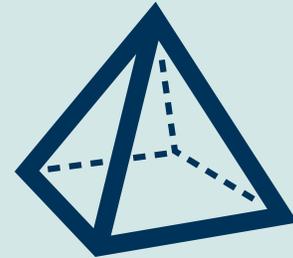
Dr.-Ing. Tobias Schwarz

# Agenda



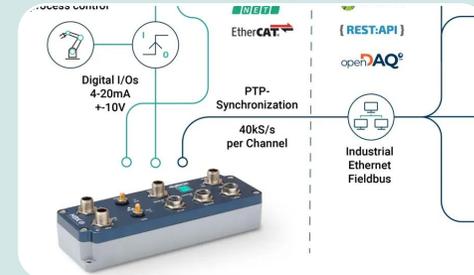
## Smart Sensors

- Messprinzip
- Prozessintegration
- Praxisbeispiel



## Entwicklungs- methodik

- Smart Sensor  
Plattform
- Testen
- CI/CD



## Deep Dive: Konnektivität

- IT/OT-  
Konnektivität
- WebAssembly
- Live Demo

# Hottinger Brüel & Kjær

- ▲ Hottinger Baldwin Messtechnik seit 1955 in Darmstadt
- ▲ 2019: Fusion mit Brüel & Kjær
- ▲ Fokusbereiche:
  - Prozesstechnologie
  - Statisches und dynamisches Monitoring
  - Prüfstandbau
- ▲ Fokusbereiche:
  - Mechanik (Kraft, Gewicht, Drehmoment)
  - Elektrik



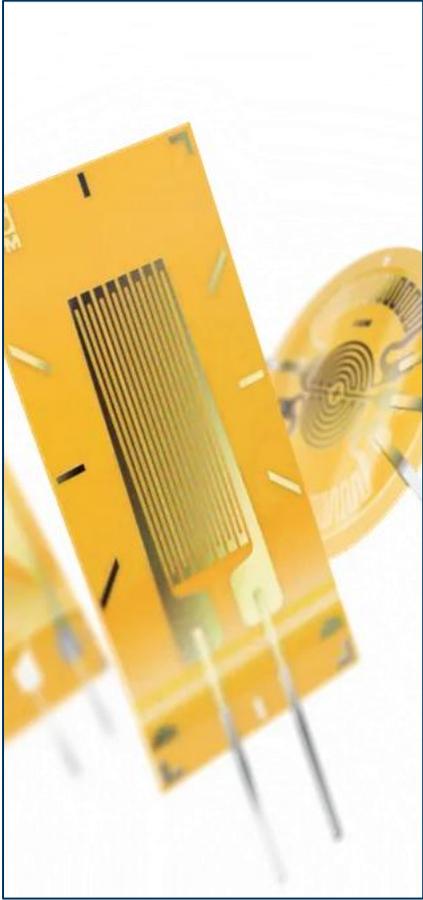
# Intro: Hottinger Brüel & Kjær



Prozesstechnologie

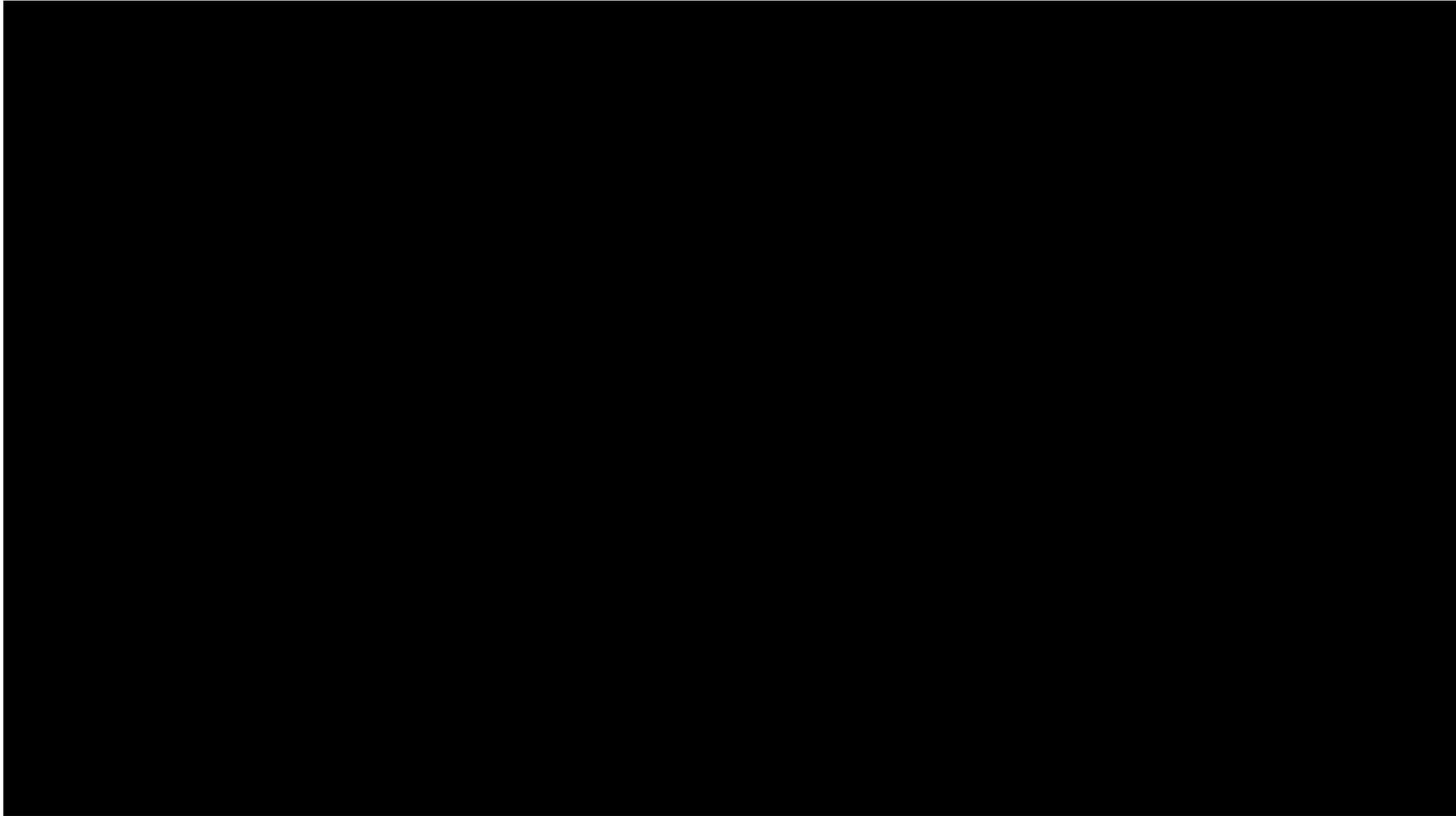


Prüfstände



Dehnungsmessstreifen

# Praxisbeispiel Prozesstechnologie



# Praxisbeispiel Prüfstände

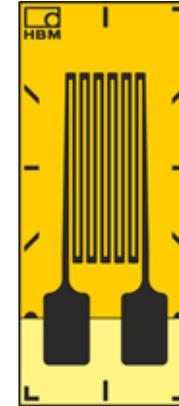
- ▲ DMP41 DMS-Messverstärker
  - Auflösung bis zur physikalischen Grenze
  - Langzeitstabilität  $\pm 5$  ppm über 40 Jahre
- ▲ Beispiel: Kalibrierung von Windkanalwaagen



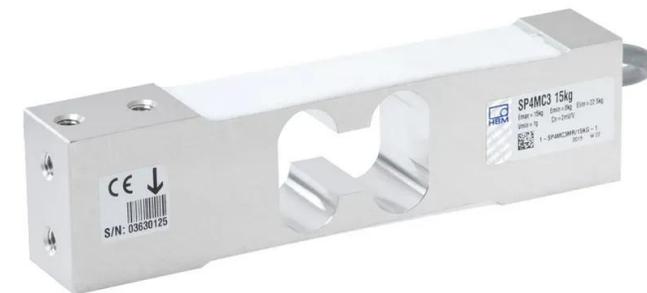
# Smart Sensors Basics

# Messprinzip: Dehnungsmessstreifen

- ▲ Dehnungsmessstreifen setzen Dehnung in Widerstandsänderung um
- ▲ Typisch  $10^{-4} - 10^{-2} \Omega/\Omega$  relative Änderung
- ▲ Applikation auf entsprechendem Messkörper erlaubt Messung von
  - Kraft
  - Gewicht
  - Drehmoment



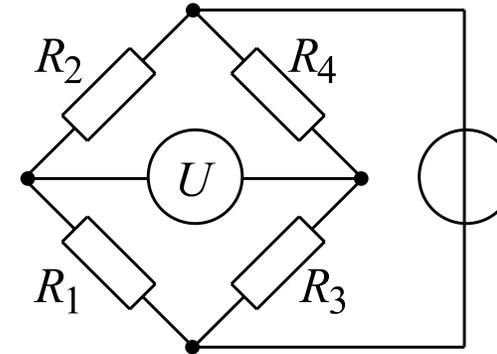
Linear-DMS



DMS-basierte Wägezelle

# Messprinzip: Wheatstone-Brücke

- ▲ Verschaltung von bis zu vier DMS (hier gezeigt: als Vollbrücke)
- ▲ Präzise Messung kleiner Widerstandsänderungen
- ▲ Weitere Vorteile
  - Temperaturkompensation
  - Gleichtaktunterdrückung



Wheatstone-Brücke

Beispiel Wägezelle:

- 2 DMS in Dehnungsrichtung ( $R_1$ ,  $R_4$ )
- 2 DMS in Stauchungsrichtung ( $R_2$ ,  $R_3$ )

Krafteinwirkung wird in Änderung der Brückendiagonalspannung  $U$  umgesetzt



# Analoge Prozessintegration von Sensoren

- ▲ Traditionelle (analoge) Einbindung von Aufnehmern in einen Prozess:



# Digitale Prozessintegration

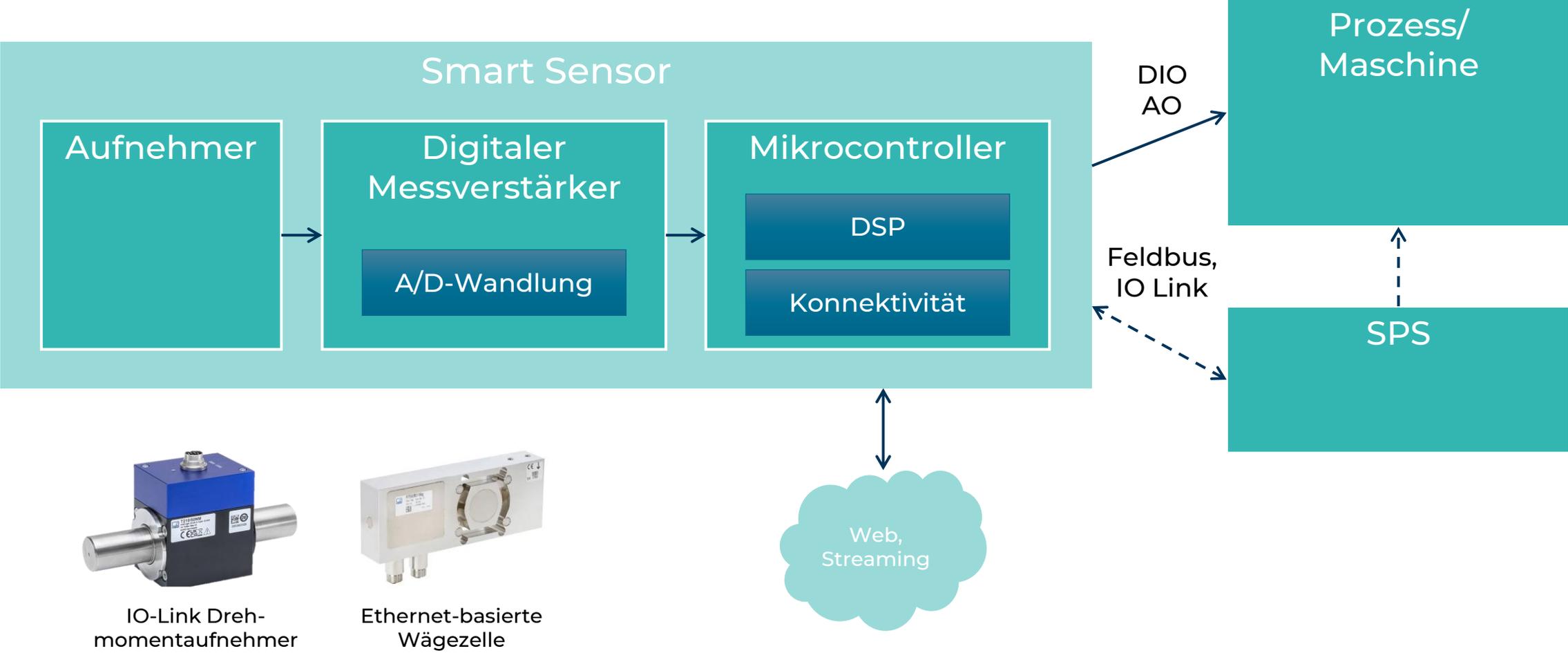
## ▲ Digitalisierung im Messverstärker:



- ▲ Digitale Signalverarbeitung im Verstärker ergänzt Kompensation im Aufnehmer
  - Digitale Filter, Skalierung, etc.
- ▲ Flexiblere Topologien durch Feldbus

# Smart Sensors

## ▲ Vollintegrierte Sensorlösung: Smart Sensor



# Smart Sensors in der Praxis

## ▲ Ketchup-Abfüllanlage revisited

- 80 Füllköpfe mit Wägezellen
- 2 kHz Datenrate pro Zelle → ~5 Mbit/s insgesamt
- Berechnung der Ventilsteuerung direkt im Sensor

### Reduzierte Datenmengen

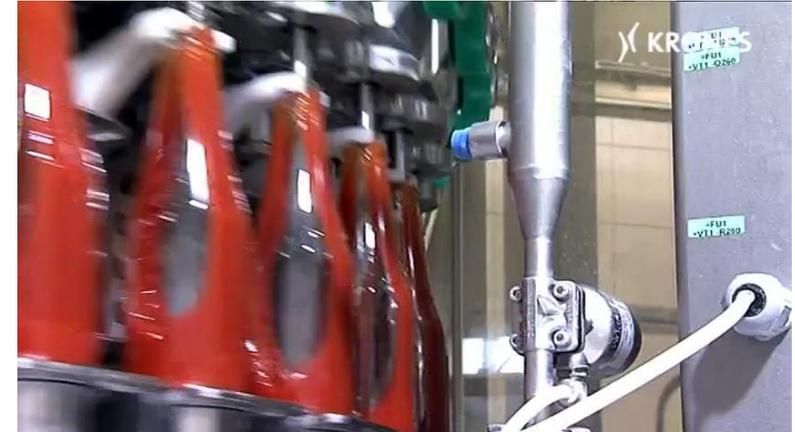
- 80 x 2 Bit Ventilsteuerung statt 5 Mbit/s

### Erhöhte Geschwindigkeit

- Direkte Steuerung ersetzt SPS mit Zykluszeit im ms-Bereich

### Geringerer Jitter

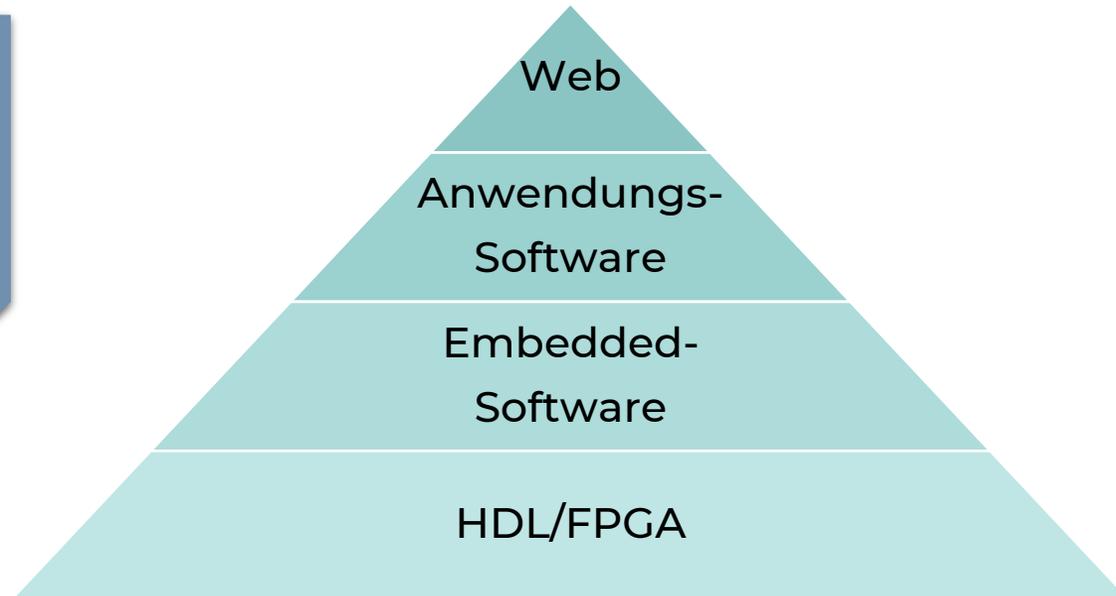
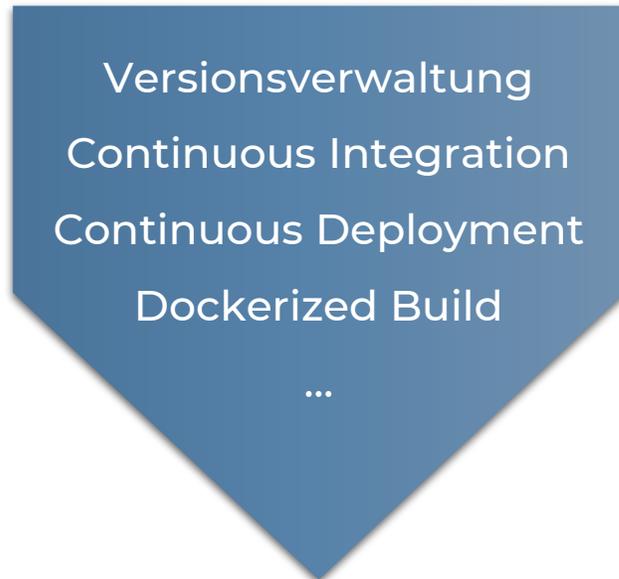
- Direkte Ansteuerung durch Digitalausgang statt Ethernet-basierten Feldbus



# Entwicklungsmethodik

# Entwicklungsmethodik: Motivation

## Entwicklungstechnologien



# Entwicklungsmethodik: Kernkomponenten

- ▲ Hardware- und Softwareentwicklung eng verzahnt
- ▲ Fokus hier: Software-/Firmware-Entwicklung

## Entwicklungsorganisation

- Scrum (SAFE)

## Technischer Entwicklungsprozess

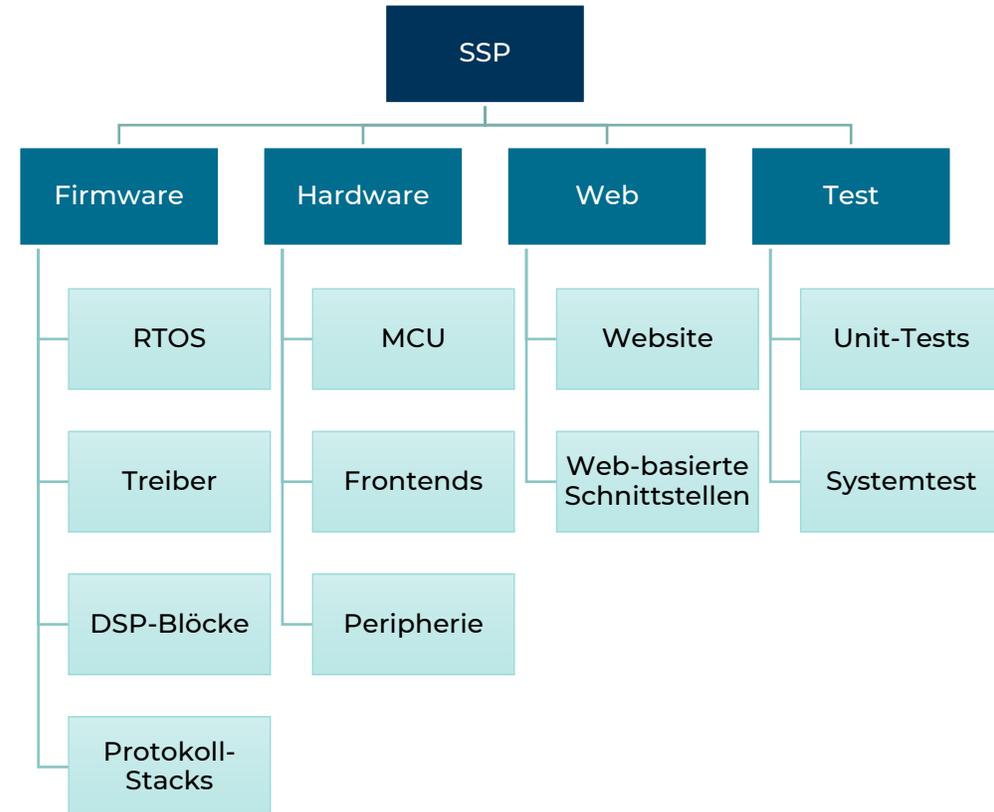
- Test Driven Development
- Continuous Integration

## Know-How

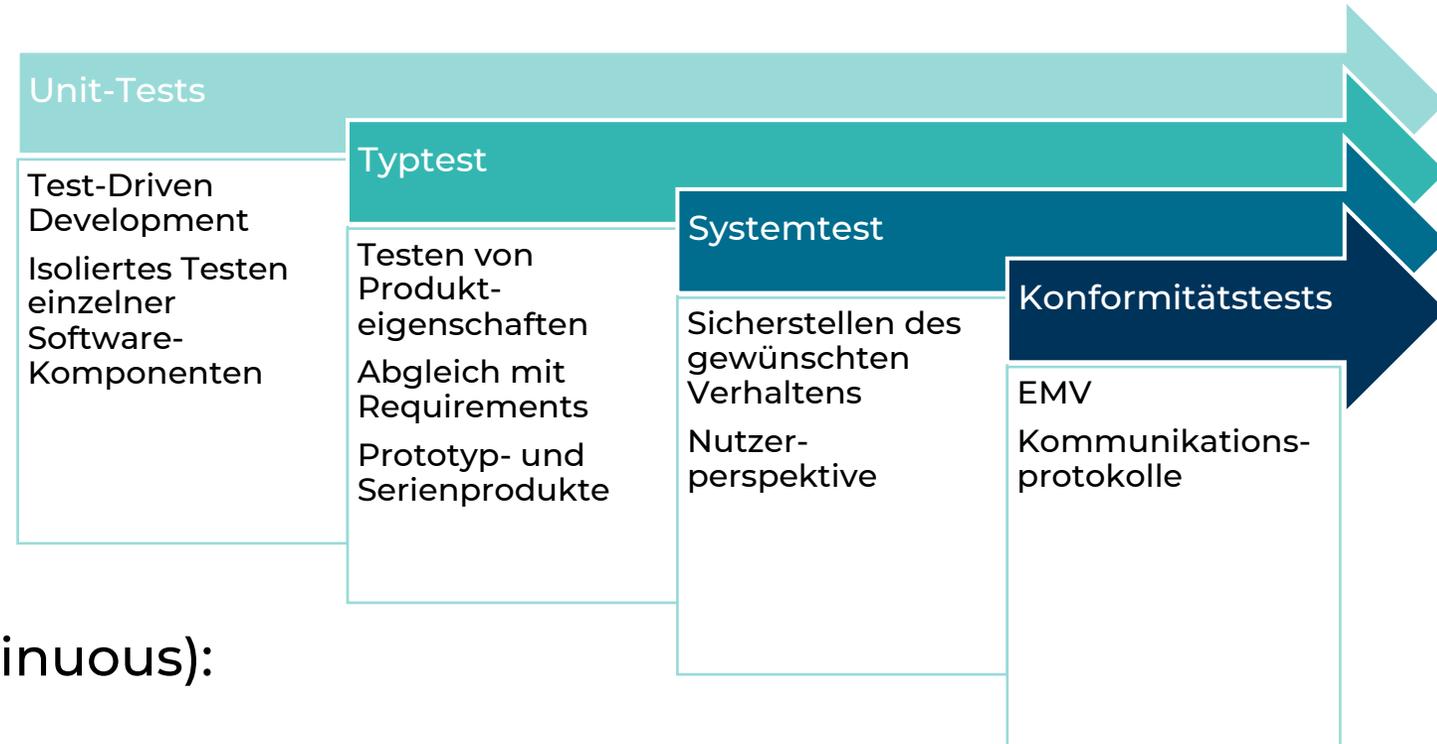
- Smart Sensor Platform

# Smart Sensor Platform

- ▲ Wiederverwendbarkeit von Hard- und Softwarekomponenten
  - über Produkte hinweg
  - über Domänen hinweg
- ▲ Beschleunigter Entwicklungsprozess außerdem durch
  - wachsendes Know-How
  - eingespielte Schnittstellen zwischen den einzelnen Teams



# Tests, Tests, Tests



- ▲ Wiederkehrend (Continuous):
  - Unit-Tests
  - Systemtest
- ▲ Punktuell im Entwicklungszyklus:
  - Typtest
  - Konformitätstest

# Test Driven Development

- ▲ Paradigma ursprünglich aus der Entwicklung von Anwendungssoftware

„Keine Zeile Code wird geschrieben, bevor ein fehlschlagender Unit-Test existiert.“

## Embedded-spezifische Herausforderung

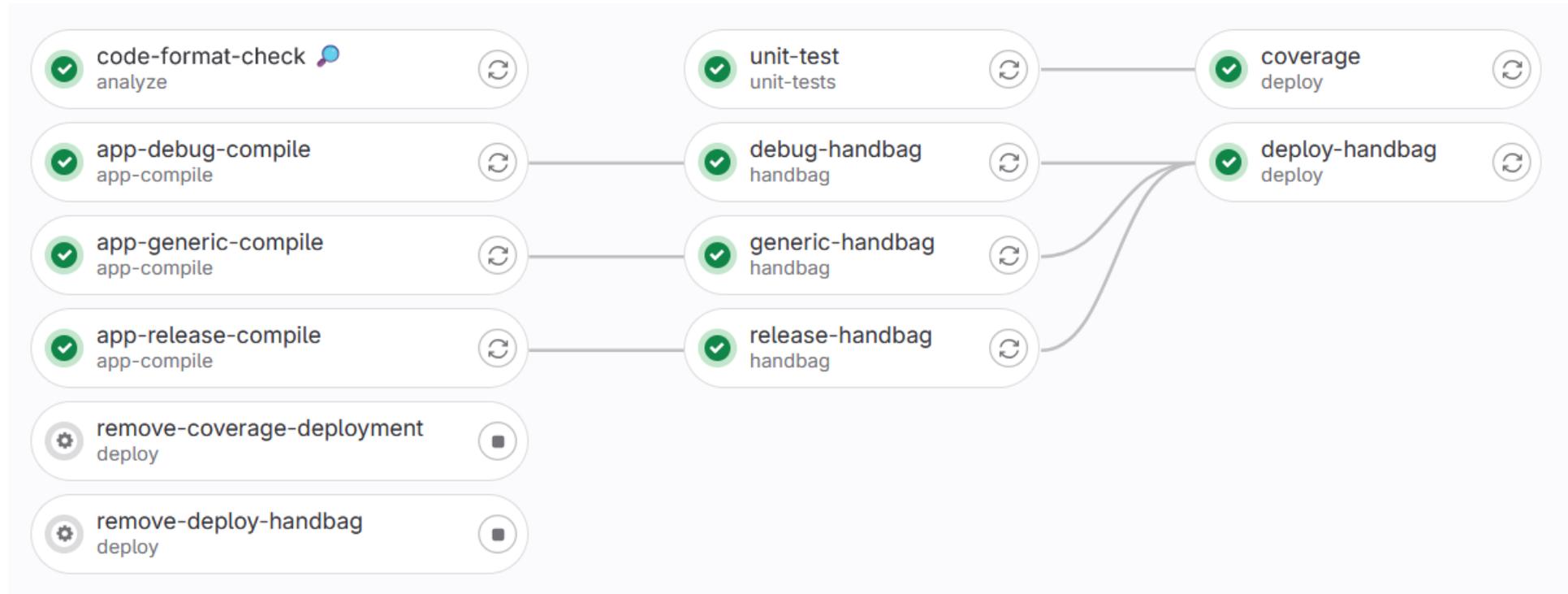
- Modellierung der Software/Hardware-Interaktion

## Embedded-spezifische Vorteile

- Code verifizierbar bevor Hardware verfügbar ist → reduziertes Risiko
- Debug-Zeit auf der Hardware verringert
- Software/Hardware-Interaktion wird durch Modellierung isoliert untersuchbar
- Testbarer Code ist modular → Decoupling von Modulen und der Hardware

# Continuous Integration

- ▲ Jede Firmware und Bibliothek besitzt eine GitLab CI-Pipeline
- ▲ Beispiel: digiBOX-Firmware
  - Hier nicht gezeigt: Systemtest-CI



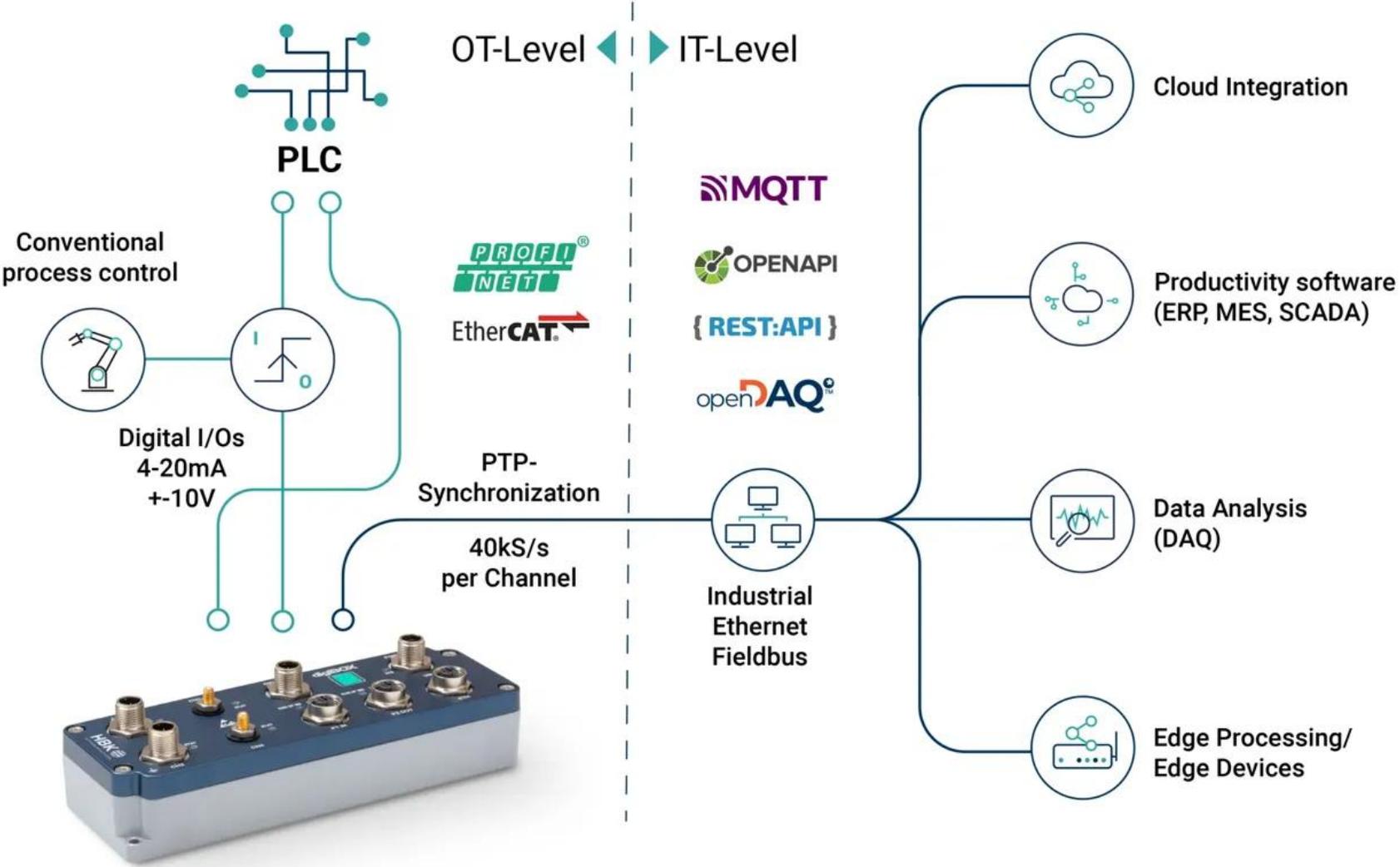
# Deep Dive: Konnektivität



# digiBOX Industrial

- ▲ Mehrkanalverstärker auf Basis der Smart Sensors Platform
- ▲ Frontends für
  - DMS-basierte Aufnehmer
  - Piezoelektrische Aufnehmer
- ▲ Umfangreiche Signalverarbeitung
- ▲ Fokus auf IT/OT-Konnektivität
  - IT: Informationstechnologie
  - OT: Betriebstechnologie

# digiBOX Konnektivität



# Edge computing vs. Cloud computing

## Edge computing

- Datenverarbeitung nah an der Quelle (hier: nah am Prozess/Aufnehmer)
- Niedriger Bandbreitenbedarf

### Digitale I/Os

- Anwendungsspezifische Steuerung

### Feldbusse

- Profinet
- Ethercat

MQTT

### Analogausgänge

- +/- 10V
- 4 – 20 mA

## Cloud computing

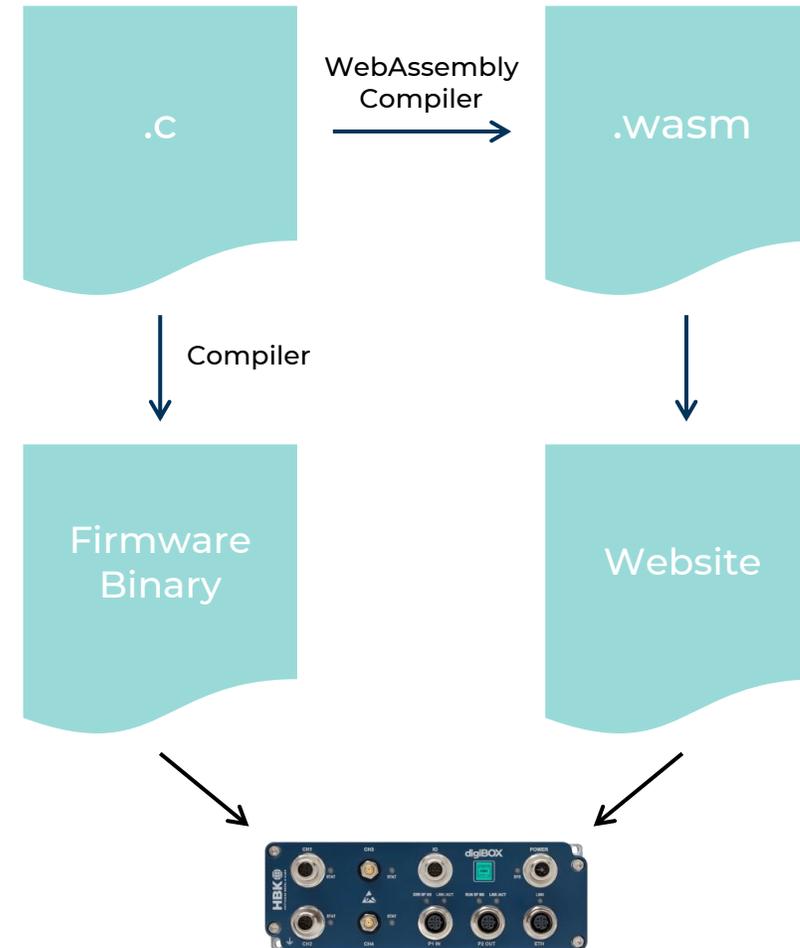
- Zentrale Datenverarbeitung
- Hoher Bandbreitenbedarf

### Ethernet-Streaming

- 4 Kanäle bei je 40 kSPS
- Zeitstempel und Statusinformationen

# WebAssembly

- ▲ Ermöglicht performante Ausführung von Code im Browser
- ▲ Beispiel: Digitale Filter
  - Firmware-Code übersetzt nach WebAssembly
  - Garantiert funktionsgleiche Simulation des Filterverhaltens auf Website



# Live Demo

# Werbung

- ▲ Wir suchen regelmäßig
  - Praktikanten
  - Werkstudierende
  - Absolventen

**Vielen Dank für Ihre  
Aufmerksamkeit!**