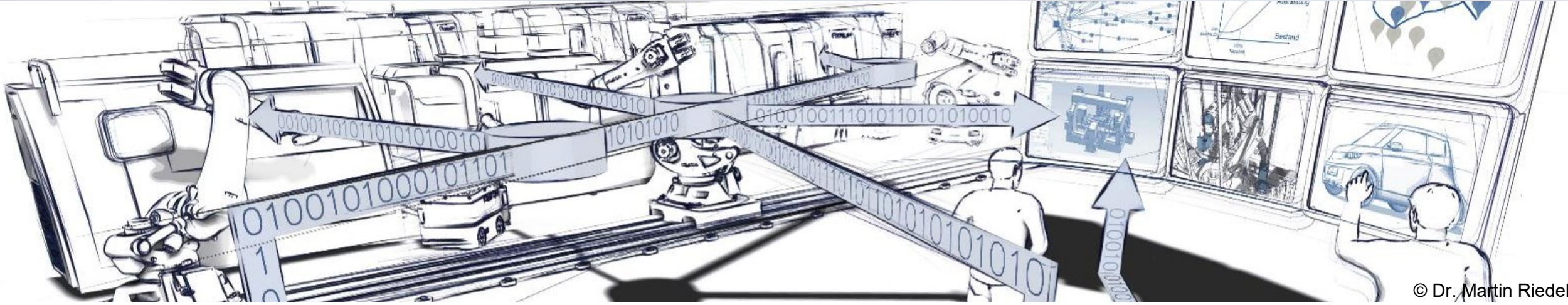


SOIL - SensOr Interfacing Language: Light-weight universal data model for FAIR Industrial Sensor Data

SensOr Interfacing Language

FAIR Sensor Data by Default

Matthias Bodenbenner

WZL | RWTH Aachen University
Chair of Production Metrology and Quality Management &
Institute for Information Management in Mechanical Engineering
Prof. Dr.-Ing. Robert Schmitt

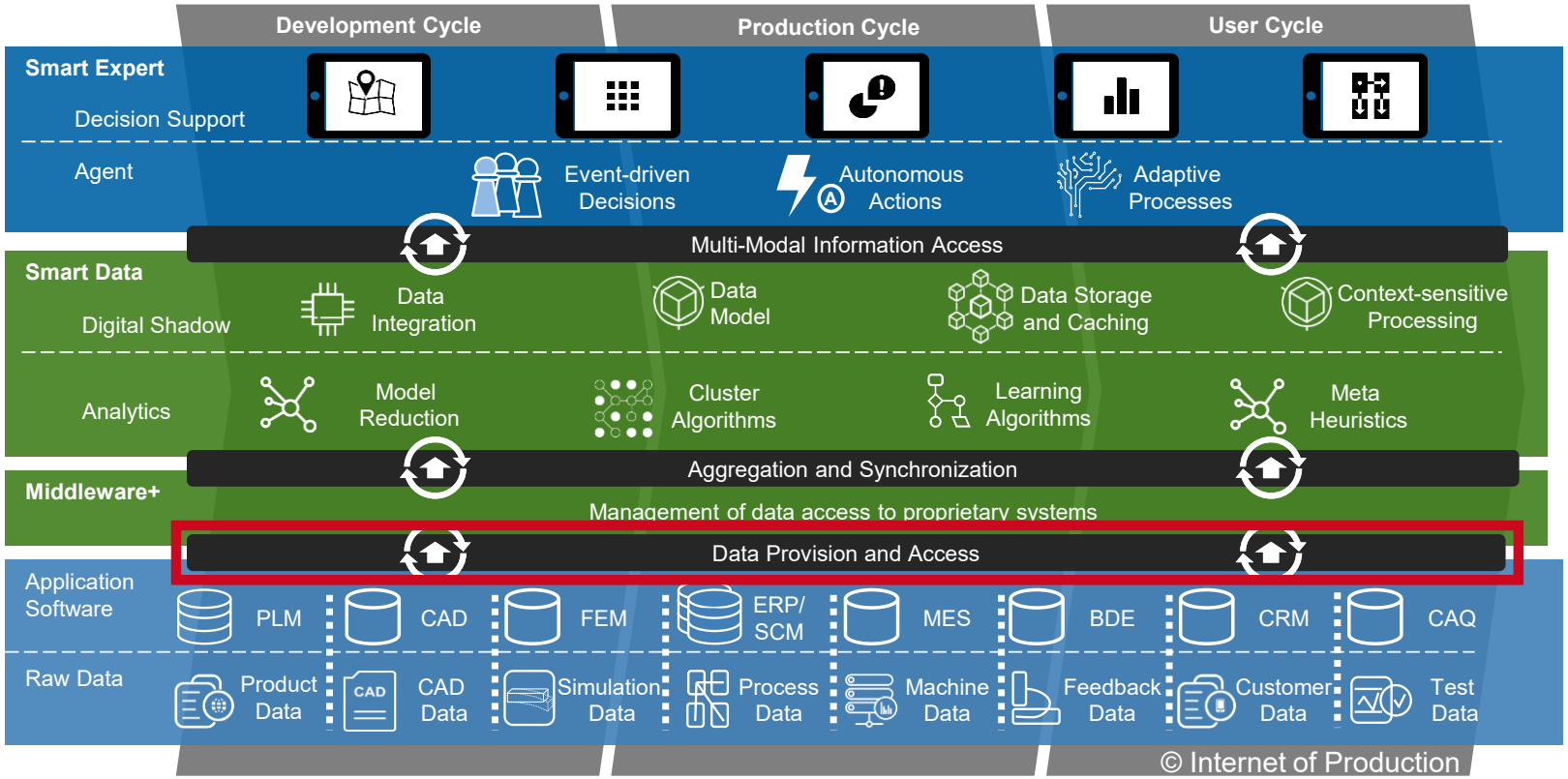
10.08.2023

NFDi4ing

AIMS

INTERNET OF PRODUCTION | WZL | RWTH AACHEN UNIVERSITY

Needs for Long-Term Reusability of Sensor Data



Making data-based decisions

Turning data in value

Deriving knowledge from data

Data aggregation

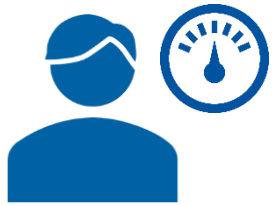
Heterogeneity of data sources

Very high amount of data

Industry demands for interoperable data models and interfaces

Content

Interface hierarchy



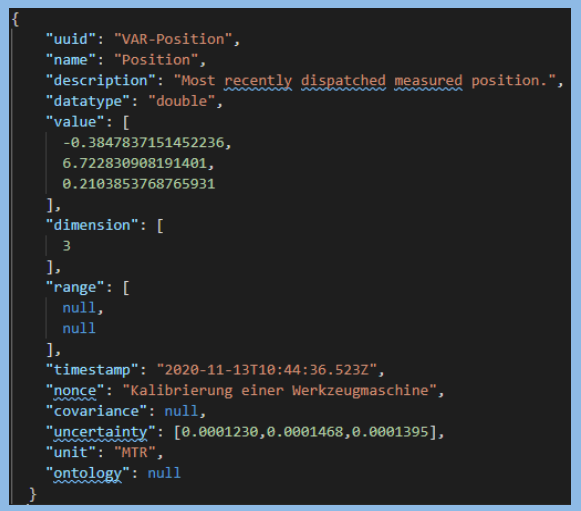
Sensor expert

Network expert

Data steward

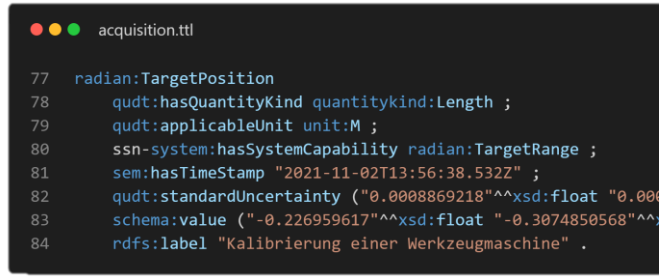
Proprietary data package
and protocol
…
216424524131765722
134765243134265724
316425727341642572
14324645
…

Manufacturer-specific interface



{ REST } MQTT.ORG

Technically interoperable interface



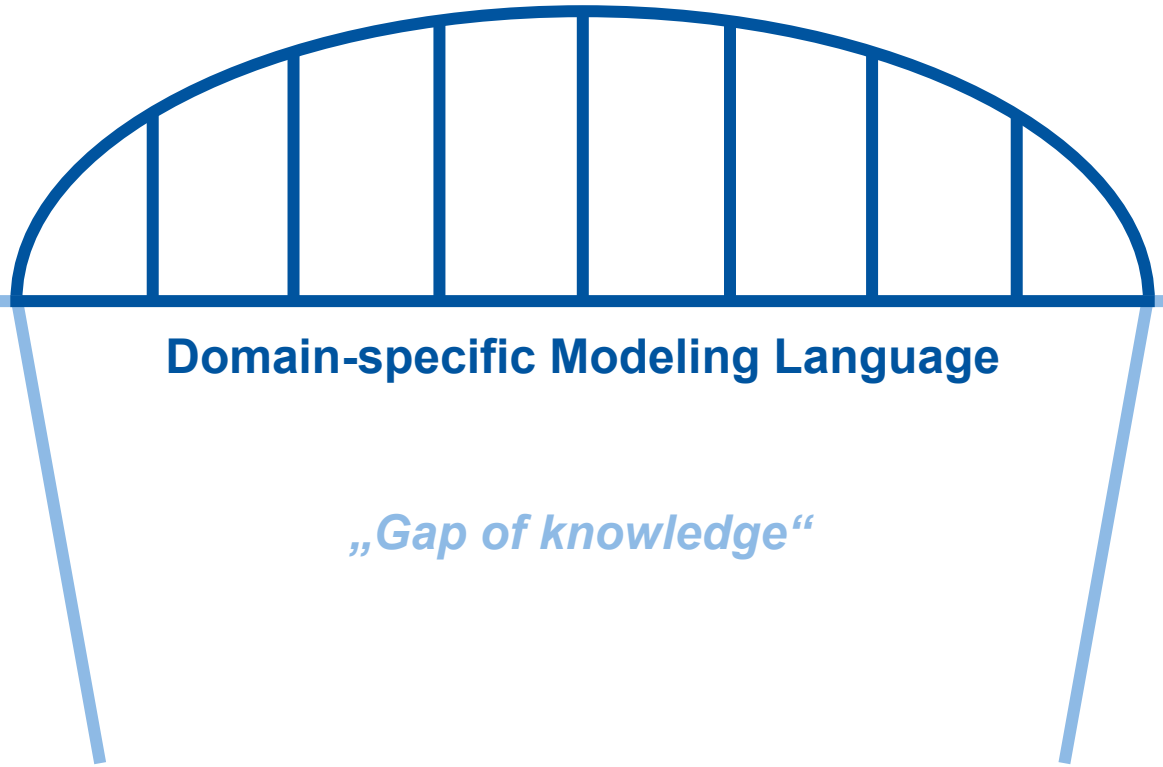
Semantically interoperable interface

FAIR Sensor Service

Approach

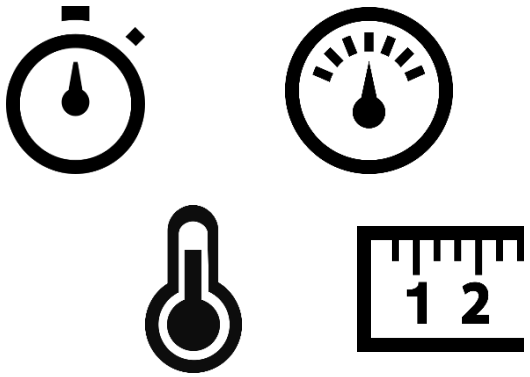
Production Engineer

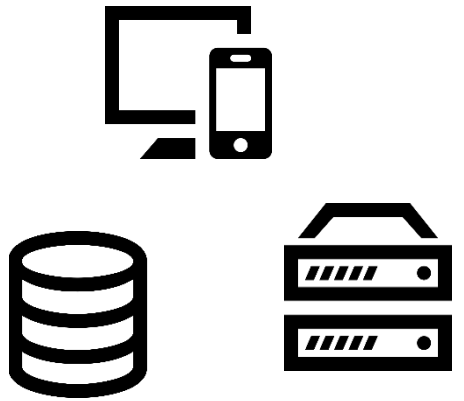
- Device knowledge
- Process knowledge



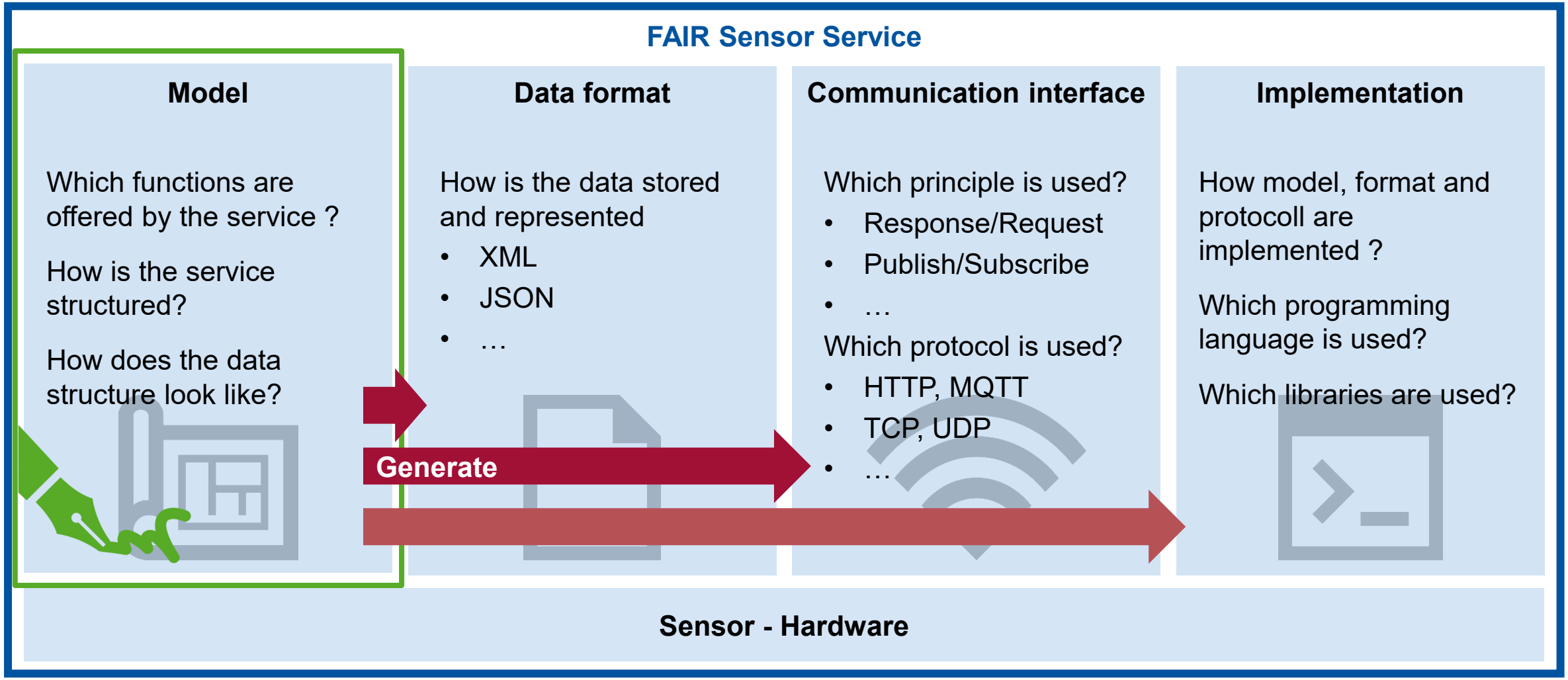
Computer Scientist

- Protocol knowledge
- Network knowledge





Horizontal Software Architecture



Content

Interoperable Measurement Data

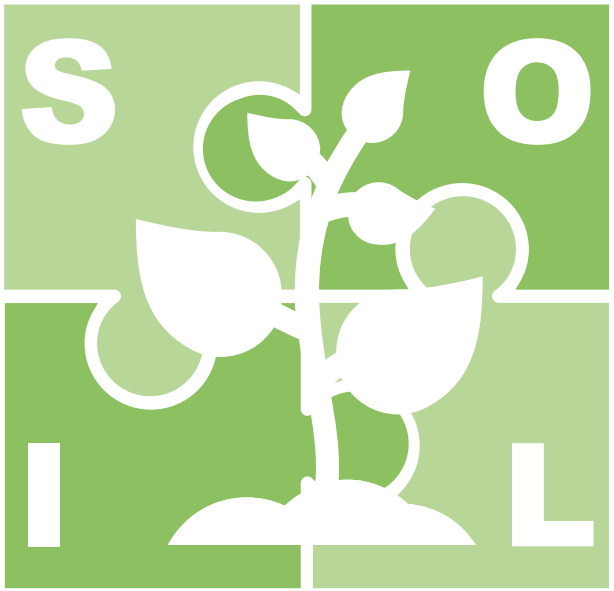
Definition of modern high-level data models

Data structure, format and communication protocol can be fully generated

Context and meta information are automatically inherited in data packages

Visual and textual web editor for (non-) programmers

SensOr Interfacing Language



FAIR Sensor Data by Default

Meta Model

Element

- ID
- Name
- Description

Parameter

- Value
- Dimension
- Range
- Datatype
- Physical Unit

Measurement

- *Value*
- *Uncertainty*
- *Timestamp*
- Physical Unit
  - Dimension, Range, Datatype

Function

- Arguments
- Returns

Component

- Components
- Functions
- Measurements
- Parameters

SensOr Interfacing Language
Minimal Example

```
measurement Temperature {
    name: "Ambient temperature"
    description: "The current ambient temperature."
    datatype: float
    range: (-20, 50)
    dimension: []
    unit: Celsius
}

component EnvironmentalSensor {
    name: "Environmental Sensor"
    description: "A single wireless environmental sensor, measuring temperature, pressure and humidity."
    measurements:
        Temperature temperature

    streams:
        temperature: fixed(5)
}

interface EnvironmentalSensor MySensor {}
```

M. Bodenbenner:
Description and Manual of the *SensOr Interfacing Language*
Zenodo - https://doi.org/10.5281/zenodo.7757249

WZL | RWTH AACHEN UNIVERSITY

HOME DOCS PUBLIC MODELS PRIVATE MODELS

https://iot.wzl-mq.rwth-aachen.de/soil/

Projects

- Robot
- Robot
- Monitoring
- Monitoring
- Lasertracker

+ Create New Project

SensOr Interfacing Language

FAIR Sensor Data by Default

PUBLISH DOWNLOAD VALIDATE GENERATE PYTHON ERROR LOG

LASERTRACKER | MOBILE_ENTITIES | BASE_STATIONS | UTILS | +

```
1  import utils;
2  import base_stations;
3  import mobile_entities;
4
5  parameter Time {
6     name: "Time"
7     description: "Current system time."
8     datatype: time
9     dimension: []
```

Errorlog

Textual representation

Lasertracker

Name: Lasertracker

Description: Active coordinate measurement device based on laser interferometry for Large-Scale metrology applications.

Components	Functions	Measurements	Parameters
baseStations	reset		state
mobileEntities	shutdown		manufacturer

Visual representation

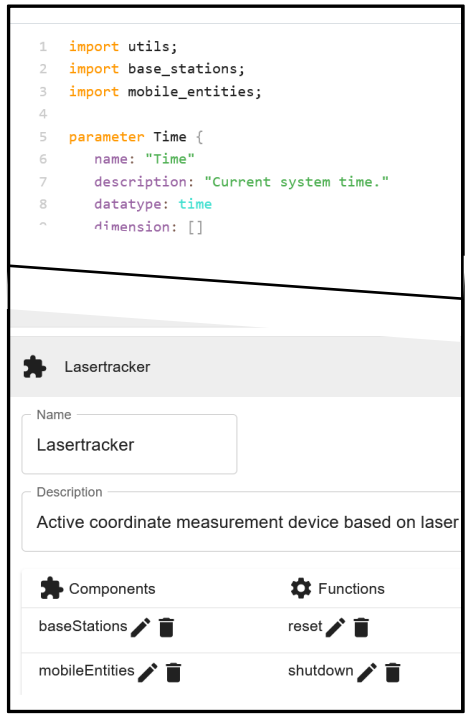
base_st

mobile_

 Matthias Bodenbenner

SensOr Interfacing Language
Automated generation of a *FAIR Sensor Services*

SOIL Editor



Server-Backend
- SOIL Parser
- SOIL Validator
- SOIL Generator

Template for implementation

```
class COMLasertrackerTOP(object):

    def __init__(self, device: Device, state: State = None,
                 time: datetime = None):
        self._device = device
        self._par_state = state
        self._par_manufacturer = manufacturer
        self._par_version = version
        self._par_time = time
        self._com_basestations = COMBaseStations(device)
        self._com_mobileentities = COMMobileEntities(device

    1 usage
    def get_par_state(self) -> State:
        return self._par_state

    def set_par_state(self, state: State):
        self._par_state = state

    def get_par_manufacturer(self) -> str:
        return self._par_manufacturer

    def get_par_version(self) -> int:
        return self._par_version

    def get_par_time(self) -> datetime:
        return self._par_time
```

Complete implementation

```
class COMLasertracker(COMLasertrackerTOP)

    def __init__(self, device: Device, st
                 time: datetime = None):
        COMLasertrackerTOP.__init__(self,
        self._com_basestations = COMBaseS
        self._com_mobileentities = COMMob

    def fun_reset(self):
        self._device.reset_tracker()

    def fun_shutdown(self):
        self._device.stop()

    def get_par_state(self):
        return str(self._device.state)

    def get_par_time(self):
        return da    e.datetime.now()
```

MQTT-Library

REST-Library

FAIR Sensor Service

➡ Manual implementation of the sensor logic
➡ Automated generation

SOIL-Datenpaket → JSON


```
{
   “covariance”: [
         [1.3276206530821578e-10, 1.0791437929694123e-11, 7.9915972948561065e-11],
         [1.079143792969412e-11, 5.5091565613752213e-11, 1.0896977314156854e-11],
         [7.9915972948561065e-11, 1.0896977314156854e-11, 5.9079887330941402e-11]
      ],
   “datatype”: “float”,
   “description”: “Position in Cartesian Coordinates in meter.”,
   “dimension”: [3],
   “hash”: “3a785573eecfc75434572906c31bae675f1c173612bc25579c996bddf4e51a85”,
   “label”: “Calibration of machine tool.”,
   “name”: “Position [m]”,
   “range”: [-40, 40],
   “timestamp”: “2022-06-13T11:27:15.046550Z”,
   “unit”: “MTR”,
   “uuid”: “MEA-Position”,
   “value”: [2.4184784967537318, 0.32977268925781256, -3.8299400458421968]
}
```

Content

Conclusion

Summary

- *FAIR Sensor Services* adopt the FAIR Guiding Principles for industrial sensor data
- *SensOr Interfacing Language* provides easy-to-use modeling framework
- Approach effectively decouples modeling and integration

Outlook

- Adopting existing terminology
  - e.g., Semantic Sensor Network Ontology
- Full semantic mapping of the SOIL meta model
- Mapping to widely applied models (OPC UA, AAS)

SensOr Interfacing Language

FAIR Sensor Data by Default

Thank you very much

Matthias Bodenbenner
Team Leader | Large-Scale Metrology
m.bodenbenner@wzl-mq.rwth-aachen.de
+49 241 80 20605 | +49 15204284307

Chair of Production Metrology and Quality Management &
Institute for Information Management in Mechanical Engineering
Prof. Robert Schmitt

Acknowledgements

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – EXC-2023 Internet of Production – 390621612.

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Project-ID 432233186 – AIMS.