DaPro

Data-driven Process Optimisation using Machine Learning in the Beverage Industry

Consortium:

Supported by:

Gefördert durch:

Bundesministerium für Wirtschaft und Energie

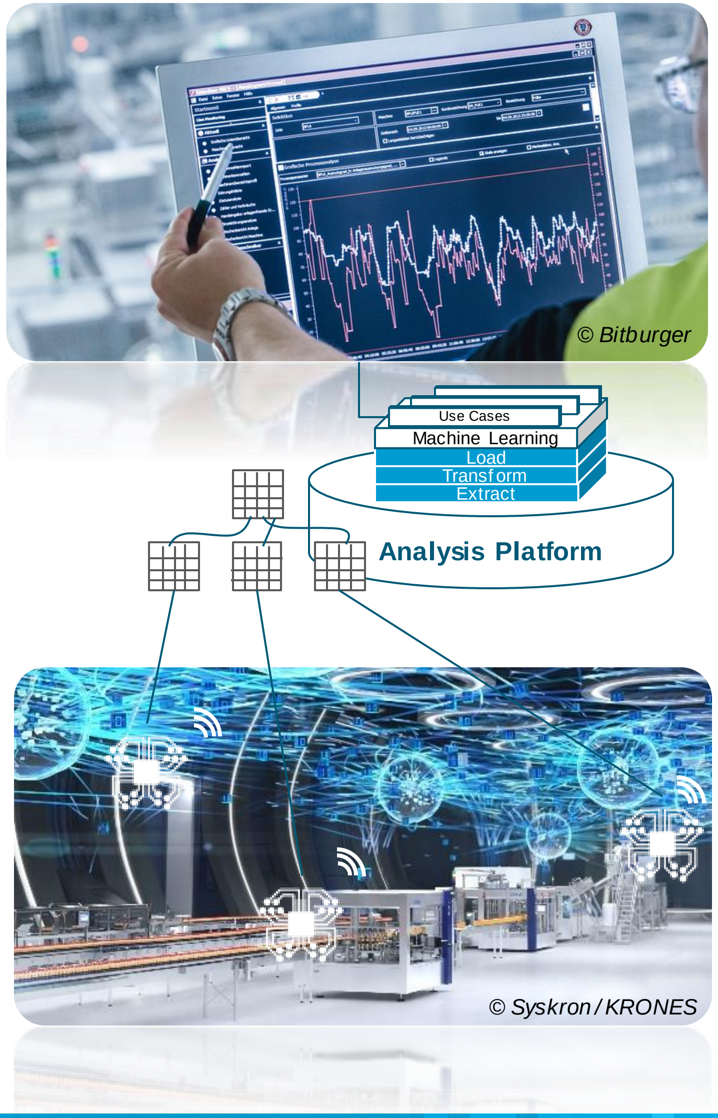
aufgrund eines Beschlusses des Deutschen Bundestages

DLR Projektträger

Reference Architecture (Compact)

www.dapro-projekt.de

Click on the section you want to navigate to

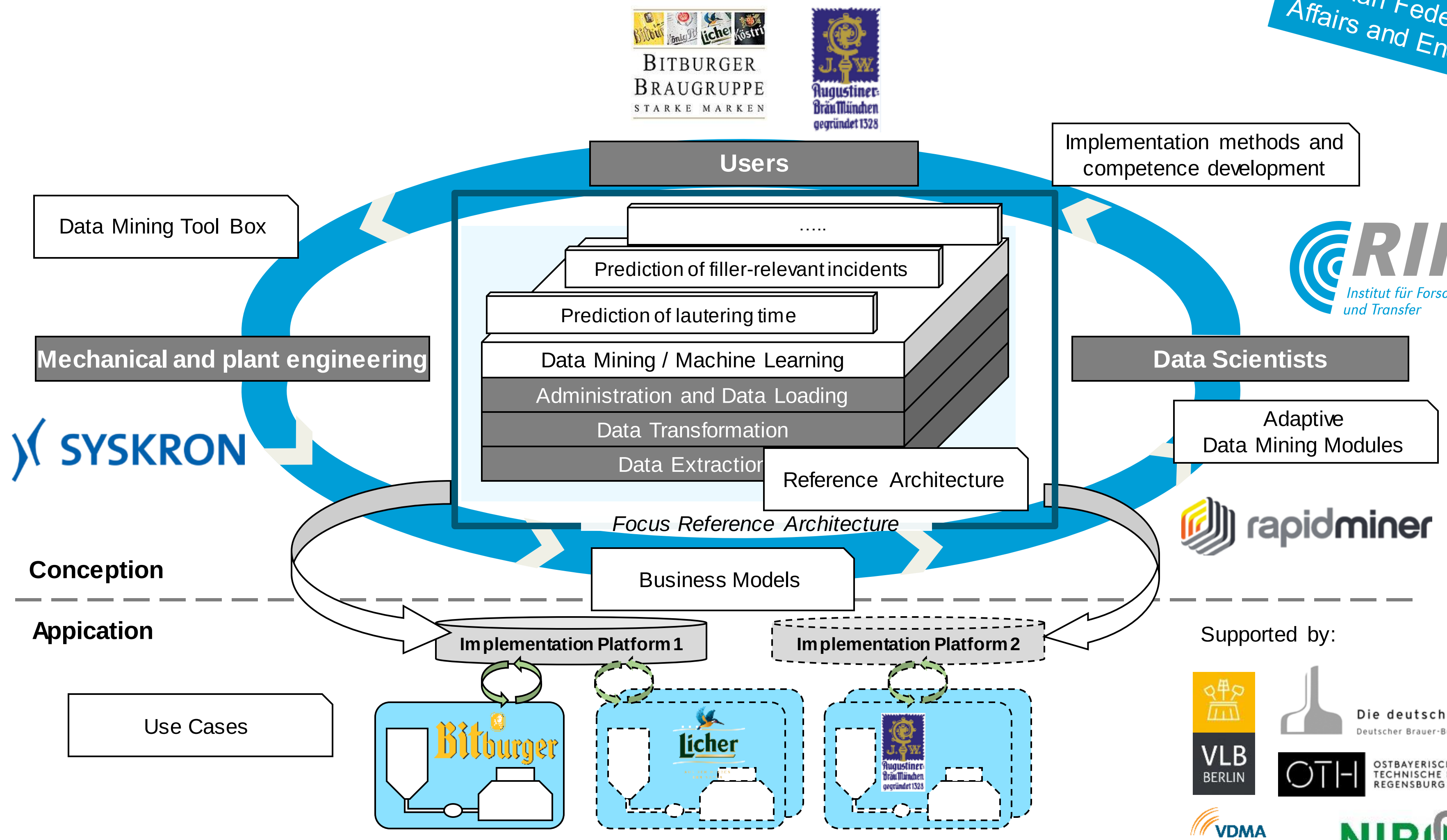


Overview

DaPro Project and Focus of the Architecture

Project Overview DaPro

Project duration:
01.01.2019 – 30.06.2022
Grant provider:
German Federal Ministry for Economic Affairs and Energy (BMWi)



Gefördert durch:
Bundesministerium für Wirtschaft und Energie
aufgrund eines Beschlusses des Deutschen Bundestages
Smarte Datenwirtschaft
DLR Projektträger

Supported by:
VLB Berlin
Die deutschen Brauer Deutscher Brauer-Bund e.V.
Swinkels family brewers
OTH Ostbayerische Technische Hochschule Regensburg
UTS University of Technology Sydney
VDMA Nahrungsmittelmaschinen und Verpackungsmaschinen
NIRO Netzwerk Industrie RuhrOst
prostep ivip

Why do we need a Reference Architecture?

- Relationships between cause(s) and effect(s) in Process Industry with complex in biochemical processes often cannot be explained with conventional methods (Lean, CIP, Six Sigma).

- The use of machine learning (ML) or artificial intelligence (AI) for product and process improvement offers solutions, but places new demands on hardware and software.

- This requires the introduction of new IT systems and underlying architectures, as well as the harmonisation of ML environments with existing IT architectures in the company.

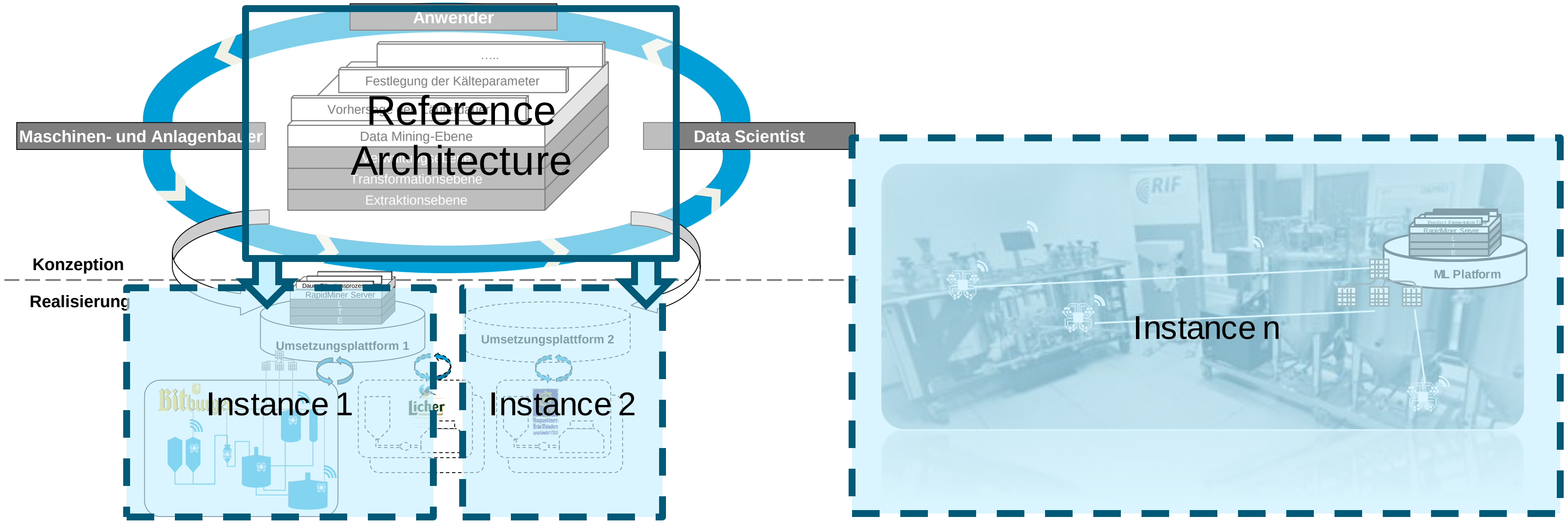
- The increasing digitalisation of plants and their connectivity ("Industry 4.0") are drivers of high data availability, but also increase the complexity that needs to be mastered.

- Existing reference models are either too abstract, manufacturer-specific and do not sufficiently combine existing IT landscapes with required new technologies.

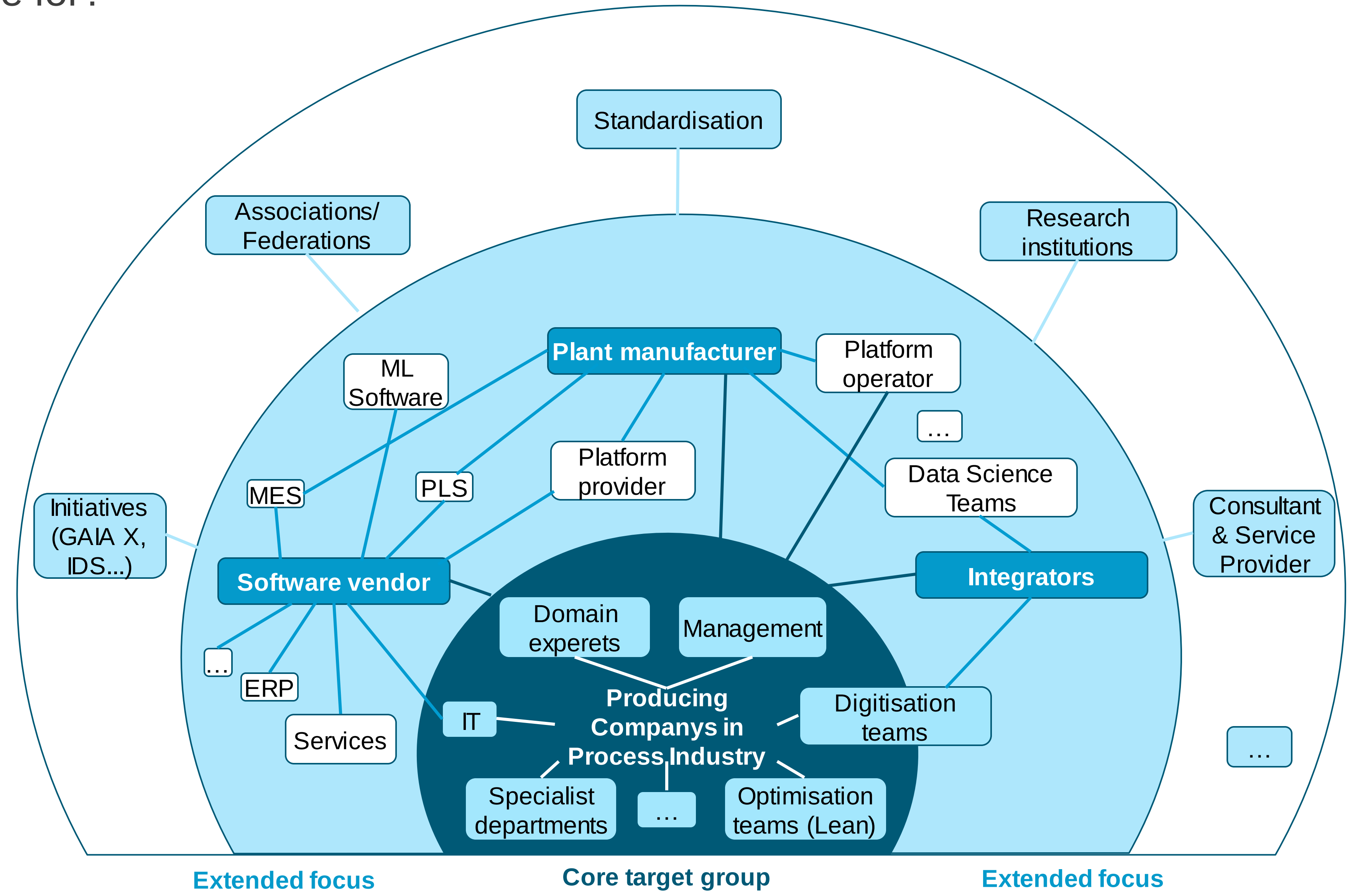
→ Methodical support and moderation for the implementation of ML architectures in your own company!

What is a Reference Architecture?

- „Abstract architecture that is intended to **make it easier for people to develop systems, solutions and applications** by **providing knowledge and a framework for development**. The relationship between reference architecture and concrete architecture is characterised by the fact that the subject matter or content of the reference architecture is (re)used in the construction of the concrete architecture of the particular system to be developed. The reference architecture **has a technical focus**, but **combines this with the associated expertise of the respective domain**. It forms a common framework through its expression and content, **around which detailed discussions of all stakeholders involved in the development can be conducted**." (Reidt et al. 2018)



Who is it made for?



What are the goals of the Architecture?

- Transparency about required elements, data flows and solution approaches for the implementation of ML in the Process Industry.
- Increasing the quality and suitability of real ML architectures while reducing risks through the use of proven solutions and good practices
- Saving time in building ML architectures by reusing or deriving elements from the reference architecture
- Enabling a structure for data-driven process optimisation and related KPI-oriented improvements
- Creation of an innovative solution space
- Use of the reference architecture as a communication element

How was the reference architecture developed?

Practical Input

Published Good Practices:
- Existing published ML implementation examples in the process industry with focus on ML architectures

DaPro project:
- Workshops with departments
  - Prodction, Logistics, Filling,…
  - IT
  - Process optimisation
  - …
- Interview study
- IT structures and requirements in practice
- Implementation of use cases and ML architectures

Beyond DaPro:
- Practical experience through numerous implementation projects at RIF and IPS
- Implementation expertise of the DaPro project partners Syskron and RapidMiner
- Discussions with associated partners, e.g. VDMA, and other companies in the process industry

Reference Architecture

Research Input

Literature:
- Research-based identification of use cases and required architecture components
- (I)IoT (reference) architectures
- ML environments and architectures
- IT structures and requirements in technology terms

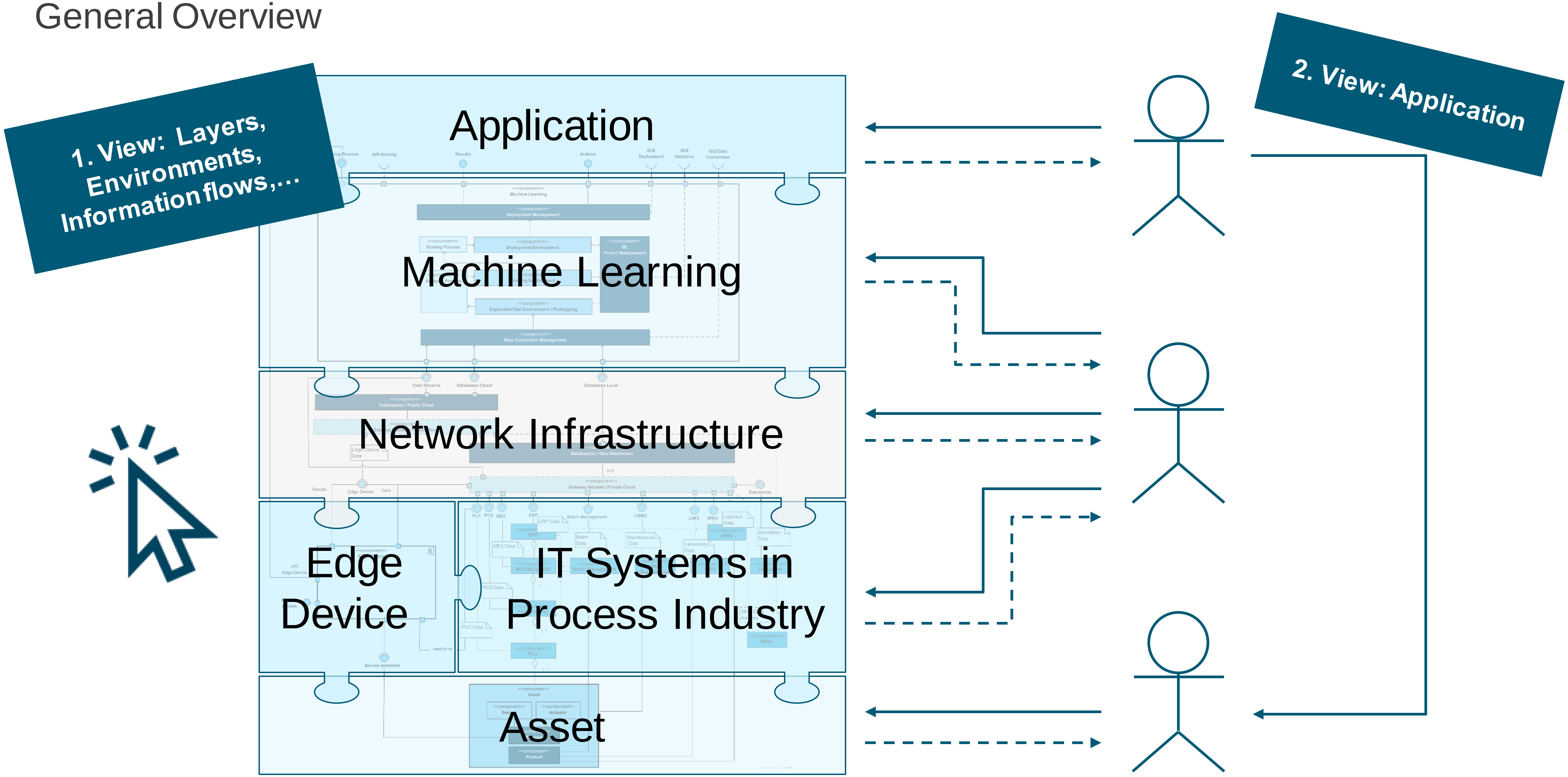
DaPro project:
- Both use cases and ML architectures are developed in a research-driven manner and tested in practice.
- Discussions with Data Science and ReadyKit team Syskron Xand research department of RapidMiner
- Research institute(s) RIF and IPS

Beyond DaPro:
- Research activities of RIF, RapidMiner and Syskron in the fields of process optimisation, ML and IoT
- Discussions with external research institutions

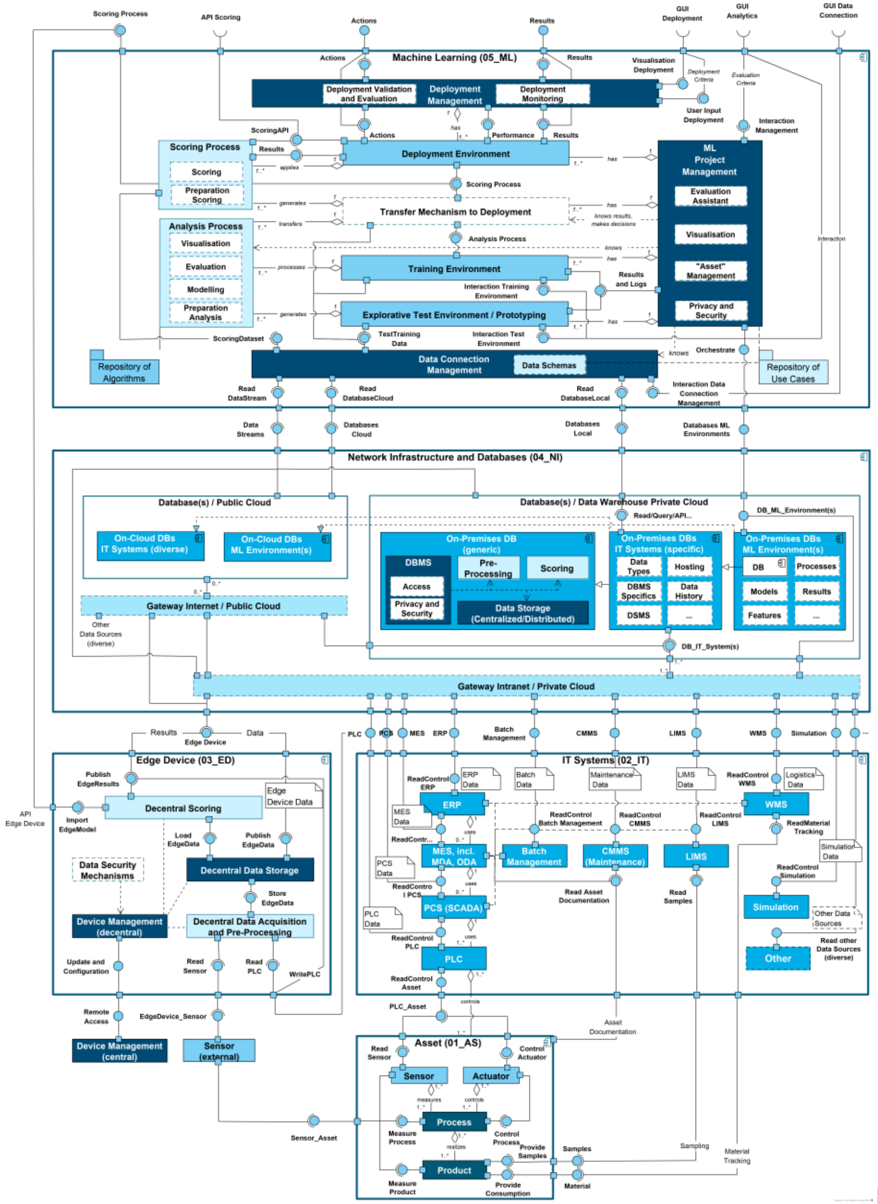
Overview of the Reference Architecture

Layers and Viewpoints

General Overview



General Overview



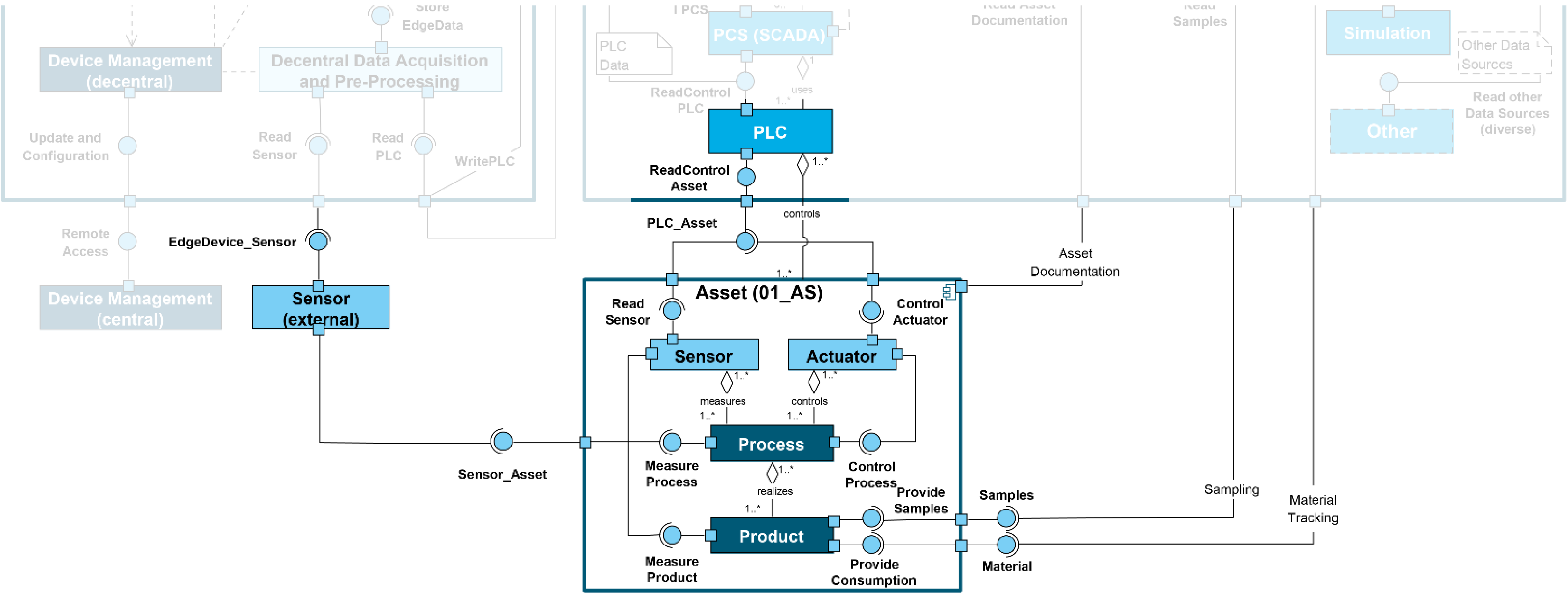
Details Online

http://dapro-projekt.de/wp-content/uploads/2021/07/Referenzarchitektur_Draft.svg

Details

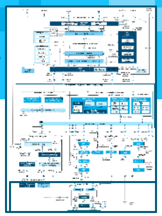
Asset

Asset Layer



Store
EdgeData
Device Management (decentral)
Decentral Data Acquisition and Pre-Processing
Update and Configuration
Read Sensor
Read PLC
WritePLC
Remote Access
Device Management (central)
EdgeDevice_Sensor
Sensor (external)
PLC Data
PCS (SCADA)
uses
ReadControl PLC
PLC
ReadControl Asset
PLC_Asset
controls
Documentation
Samples
Simulation
Other Data Sources
Read other Data Sources (diverse)
Other
Asset Documentation
Asset (01_AS)
Read Sensor
Control Actuator
Sensor
Actuator
measures
controls
Process
Measure Process
Control Process
realizes
Product
Measure Product
Provide Consumption
Provide Samples
Samples
Material
Sampling
Material Tracking
Sensor_Asset
1..*

Reference Architecture

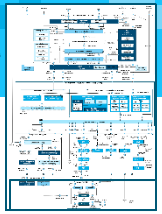


Asset Layer - Summary

- The basic layer of the architecture includes the representation of processes and/or products to be optimised, which are realised with the help of an asset.
- The level represents a combination of the established ISA 88 and ISA 95 structures with common modelling forms of cyber-physical production systems (CPS) or “Industrie 4.0” components.
- An asset corresponds to a plant according to ISA 88, in which a control recipe of a product is transferred into a procedure and realised by one or more processes.
- It is possible to divide the asset into sub-assets, technical equipment and individual control units as well as processes into process sections, operations and steps.
- The reference architecture does not prescribe a mandatory structure for this, in order to be able to address a variety of real scenarios. Links between plants, processes and products are relevant, e.g. via unique batch IDs.
- A specification is made at the higher level of the IT systems, where data is generated and interventions are made.
- Measuring points realised by sensors and possible process interventions by actuators, which can be well analysed and visualised for example by PFD or P&ID diagrams, are of particular importance.
- There is an optional interface to edge devices that can upgrade existing systems to CPS through additional sensors, connection to network structures or intervention options on the PLC.

Asset Layer - Notes

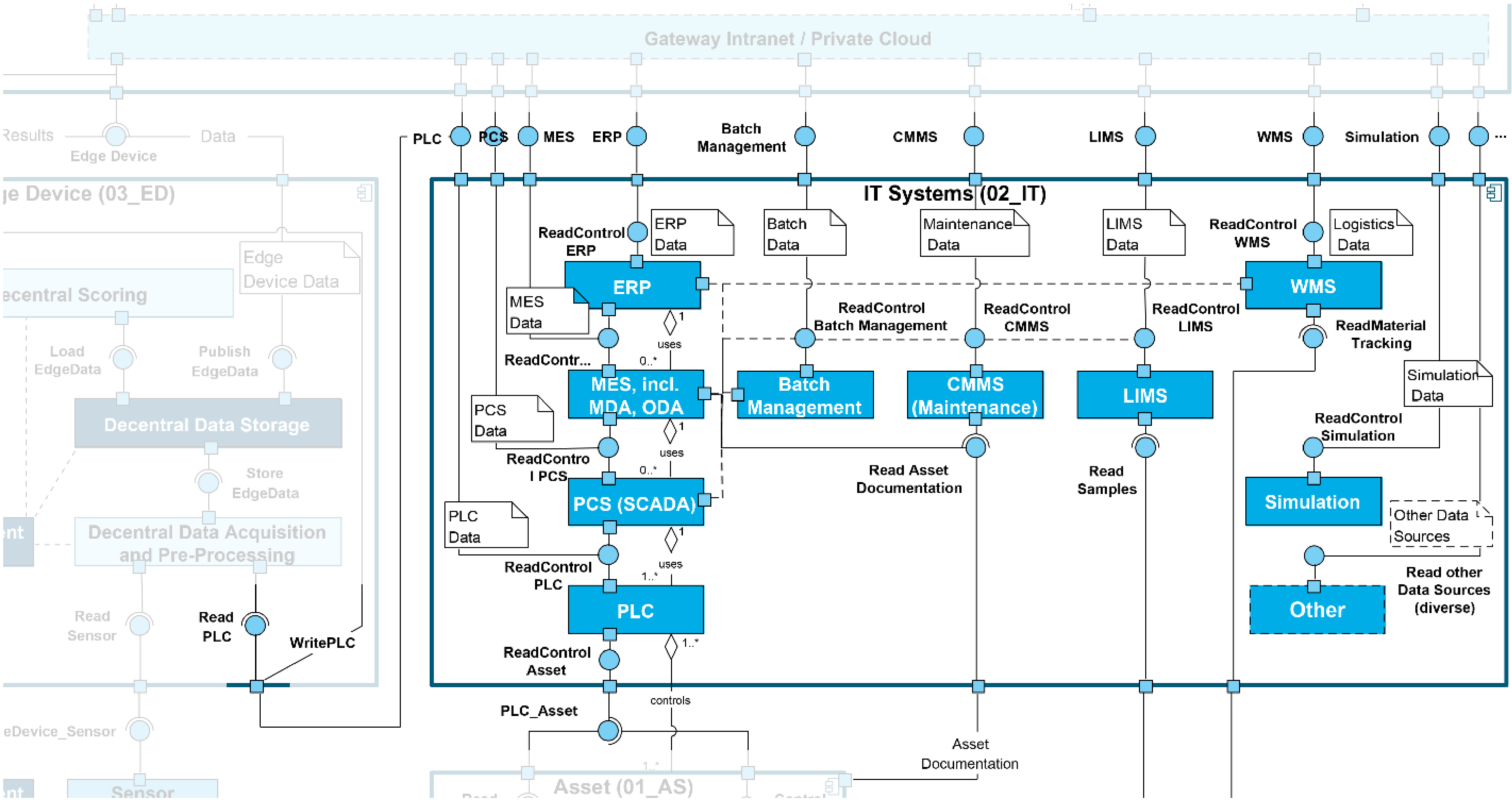
- DIN EN 61512-1:2000-01 defines the fundamental structures of general, site, master and control recipes. In addition to the products, recipes also describe their production process. The design options of products, processes and underlying plant technologies are manifold, thus there is no specification.
- The physical model of DIN EN 61512-1:2000-01 as well as PFD and P&ID diagrams are established as standards for structuring and visualising plants in the process industry, especially the underlying sensors and actuators.
- Furthermore, there are initiatives in the DEXPI project or the ISO 15926 series for the harmonised data exchange of flow diagrams and associated plant structures as well as metadata.
- DIN EN ISO 10628-2:2012 introduces a list of common plant components and technical equipment via vessels and tanks, centrifuges, heat exchangers, dryers, comminution machines, pumps, valves, etc.. They represent the actuators of the asset.
- There are no limits to the diversity of real designs, so that products, processes, actuators and sensors take on individual characteristics and are to be analysed according to the objectives.



Asset - Solution Options

		Exemplary design options									
Asset	Product	Reactants (Raw materials)	Main product	Coupling product	By-product	Inert substances (solvents/carriers)	Catalysts	Contaminants			
	Process	Mixing	Separating	Heating	Cooling	Chemical Reactions	Forming	Transporting	Filling	Packing	
	Sensor	Temperature	Pressure	Flow rate	Density	Viscosity	pH value	Fill level (radar, pressure, ultrasonic...)	Spectroscopy (NIR, UV, laser,...)	Laboratory analytics (various)	…
	Actuator	Pump	Valve	Centrifuge	Heat exchanger	Dryer	Steam generator	Seperator	Filter	Grinder	
		Vessels and Tanks	Pipe	Mixer	Shaping machine	Compressor	Lifting, conveying, transport equipment	Agitator	Fittings	Blowers	

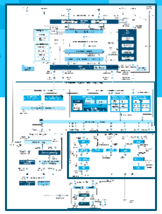
IT Systems



Gateway Intranet / Private Cloud
PLC
PCS
MES
ERP
Batch Management
CMMS
LIMS
WMS
Simulation
...
IT Systems (02_IT)
ReadControl ERP
ERP Data
Batch Data
Maintenance Data
LIMS Data
ReadControl WMS
Logistics Data
ERP
WMS
MES Data
ReadControl Batch Management
ReadControl CMMS
ReadControl LIMS
ReadMaterial Tracking
ReadContr...
uses
MES, incl. MDA, ODA
Batch Management
CMMS (Maintenance)
LIMS
Simulation Data
PCS Data
ReadControl Simulation
ReadContro l PCS
Read Asset Documentation
Read Samples
PCS (SCADA)
Simulation
Other Data Sources
PLC Data
ReadControl PLC
Read other Data Sources (diverse)
PLC
Other
ReadControl Asset
controls
PLC_Asset
Asset Documentation
Asset (01_AS)
Edge Device
Results
Data
Edge Device (03_ED)
Edge Device Data
Decentral Scoring
Load EdgeData
Publish EdgeData
Decentral Data Storage
Store EdgeData
Decentral Data Acquisition and Pre-Processing
Read Sensor
Read PLC
WritePLC
eDevice_Sensor
Sensor

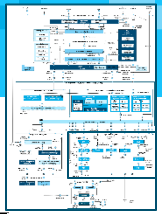
IT Systems - Summary

- IT systems provide the main data sources for the analyses. Based on the asset level, it expands the digital representations of the product, process and the plant itself to also include cross-sectional systems such as LIMS, WMS, maintenance or simulation software.
- The basic structure follows the automation pyramid according to ISA 88 and 95 or the real shop-floor IT.
- PLCs communicate with one or more assets at field level and transfer recipes to the sequence control, whose execution is controlled by actuators and progress is documented via sensor values and machine states.
- As part of their higher-level control function, PCSs address the actual and target processing of a recipe across several (sub-)plants and can be linked to batch information.
- MES transfer customer orders from ERP systems into concrete production orders to the PCS level; on the other hand, a higher-aggregated KPI measurement takes place. They therefore represent an important level of productivity measurement based on KPIs. MDA and ODA systems are also relevant.
- Other systems can be stand-alone or integrated into comprehensive MES and ERP systems. Batch management is important for the allocation of recipes, batches and raw materials to product, process and plant data, LIMS for the label generation of quality parameters, simulation software can be used for the artificial generation of data sets for ML.
- Common to the systems is an integration layer via the network infrastructure, through which data sets are integrated in batch or stream form in local or cloud-based databases.



IT Systems - Notes

- The IT system layer is characterised by application scenarios and IT landscapes in the company and can be expanded as needed. The following illustration presents exemplary design options; detailed market overviews are available online.
- The ERP market is dominated by SAP, other providers are Microsoft Navision or Oracle. Special tasks such as WMS, maintenance (CMMS) or LIMS are either mapped by individual software or increasingly integrated as modules in widespread MES or ERP systems.
- From the MES level upwards, the designs are more individually adapted to the plant and the production area. MES and PCS solutions are also frequently integrated.
- If no historical data is available, simulation systems can also be important data sources for generating test and training data. The diversity ranges from discrete-event process simulation (e.g. Plant Simulation) to chemical process simulation (e.g. ProSIm BatchReacotor) or free modelling (e.g. MATLAB).
- The concrete design of the IT systems is company-specific. For example, not all systems need to be available for all application scenarios. However, clear reference values (e.g. batch IDs or time stamps) are important, as the IT systems provide the data basis for later analysis. In addition to data scaling, attention must also be paid to data history.
- A ML-oriented data quality assessment method is provided by (Eickelmann et al. 2019).

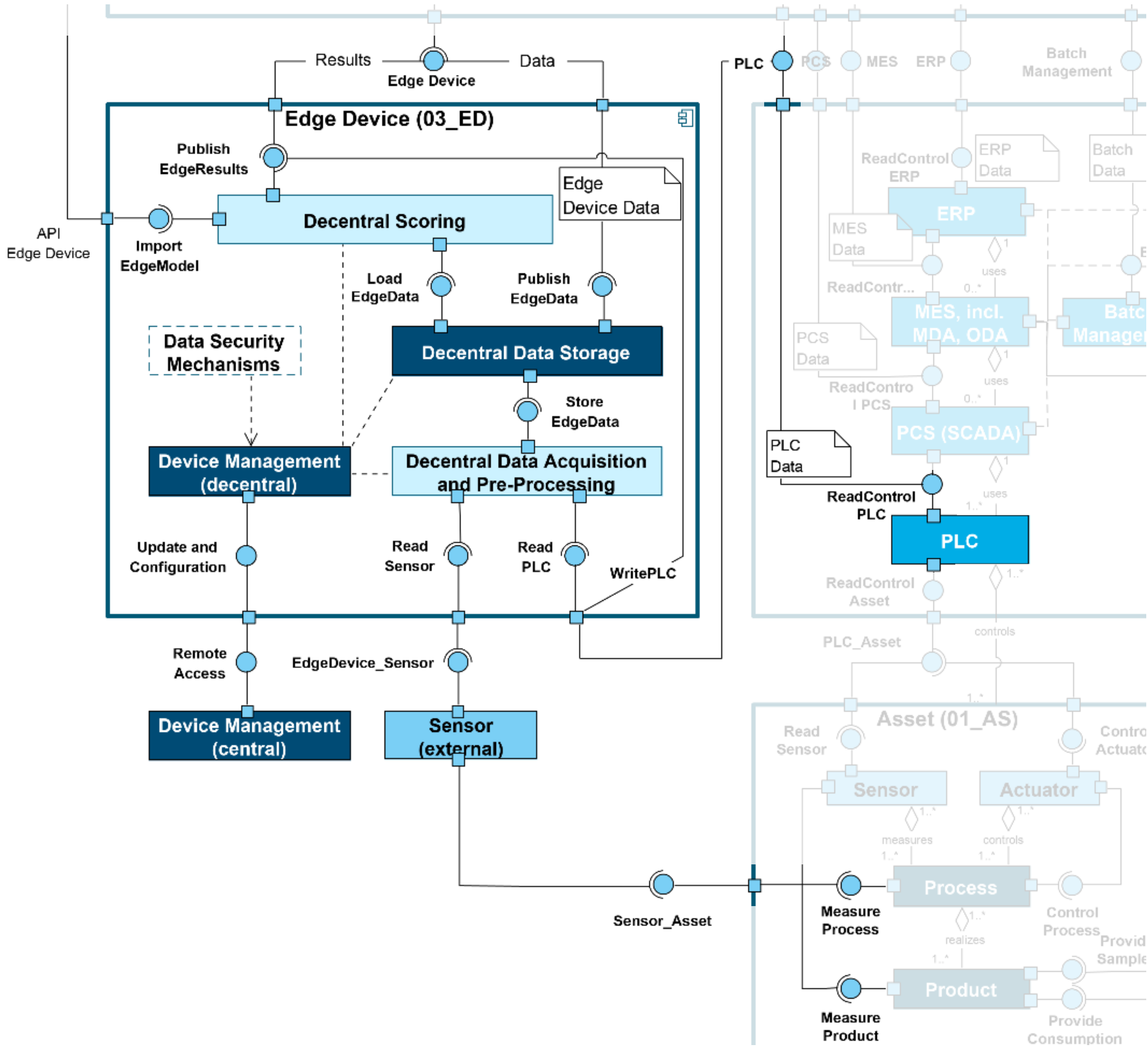


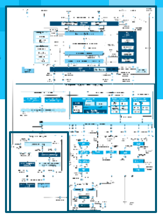
IT Systems - Solution Options

<table>
<tr><th></th><th></th><th colspan="9">Exemplary design options</th></tr>
<tr><td rowspan="9">IT Systems</td><td>ERP</td><td rowspan="5">SAP</td><td rowspan="2">Microsoft Dynamics AX/ Navision</td><td>Oracle Cloud ERP</td><td>Infor LN/COM</td><td rowspan="4">ORDAT FOSS</td><td rowspan="4">GUS-OS</td><td>3S Process</td><td>Comach ERP Enterprise</td><td rowspan="8">…</td></tr>
<tr><td>WMS</td><td>PROLAG</td><td>MoTIS</td><td>KiSoft</td><td>PSIwms</td></tr>
<tr><td>LIMS</td><td>blomesystem</td><td>Unilab</td><td>WinLMS</td><td>Infomodul LAB</td><td>Labordatenbank</td></tr>
<tr><td>CMMS</td><td>Boom Maintenance Manager</td><td>IBM Maximo</td><td>GE Predix APM</td><td>Siemens COMOS</td><td>IAS MEXIS DIVA</td></tr>
<tr><td>Batch Management</td><td>Rockwell Logix Batch and Sequence</td><td>Brewmaxx Batch Cockpit</td><td rowspan="3">ProLeiT Plant Batch iT</td><td rowspan="3">ZiPTECH ZIPMATIC</td><td rowspan="3">Siemens BRAUMAT</td><td>AVEVA Batch Management</td><td>ABB System 800xA</td></tr>
<tr><td>MES</td><td>PAS-X</td><td>MPDV Hydra</td><td>Syskron SitePilot</td><td>KHS Innoline</td><td>DEA2/sql</td></tr>
<tr><td>PCS (SCADA)</td><td>SIMATIC PCS 7</td><td>ABB Ability</td><td>Syskron Botec</td><td>B&R APROL</td><td>PSA PCS</td></tr>
<tr><td>PLC</td><td>SIMATIC S7</td><td>Beckhoff TwinCAT</td><td>PhoenixContact PLCnext</td><td>Mitsubishi MELSEC</td><td>B&R X20</td><td>Industrial Shields PLC Arduino/ RPi</td><td>Bosch Rexroth ILC / ctrlX</td><td>WAGO 750 /PFC/…</td></tr>
<tr><td>Simulation</td><td>Siemens PLM Plant Simulation</td><td>Dassault Systemès DELMIA</td><td>MathWorks MATLAB</td><td>Fortum Apros</td><td>GTT ChemSheet</td><td>CAPE-OPEN DWSIM</td><td>ProSim BatchReactor</td><td>Wolfram System Modeler</td><td></td></tr>
<tr><td></td><td>Others</td><td colspan="9">…</td></tr>
</table>

Reference Architecture

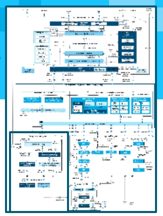
Edge Device





Edge Device - Summary

- The edge device is an optional component of the reference architecture for the decentralised execution of analysis processes to enable short control loops and low data traffic.
- Edge devices are particularly suitable for analysing data streams. This is supported by developments in AI-optimised hardware such as GPUs (Graphical Processing Unit) or NPUs (Neural Processing Unit), which are increasingly being used in a decentralised manner.
- The focus is less on training models and more on data preprocessing (e.g. feature extraction) and application (deployment or scoring).
- First of all, an interface to sensor and system data is required to provide the basis for decentralised data collection, preprocessing and storage. Existing sensors and machine states can be read out via the PLC, and edge devices also enable the integration of additional sensors via so-called retrofitting if the PLC cannot be accessed or new measuring points are required.
- Furthermore, an interface for importing models and device management must be provided.
- The edge device acts parallel to the "IT systems" layer and is integrated via the network infrastructure.
- The output of the edge device consists of raw or pre-processed data streams as well as analysis results that can be made available either to employees or to the plant or a control system.



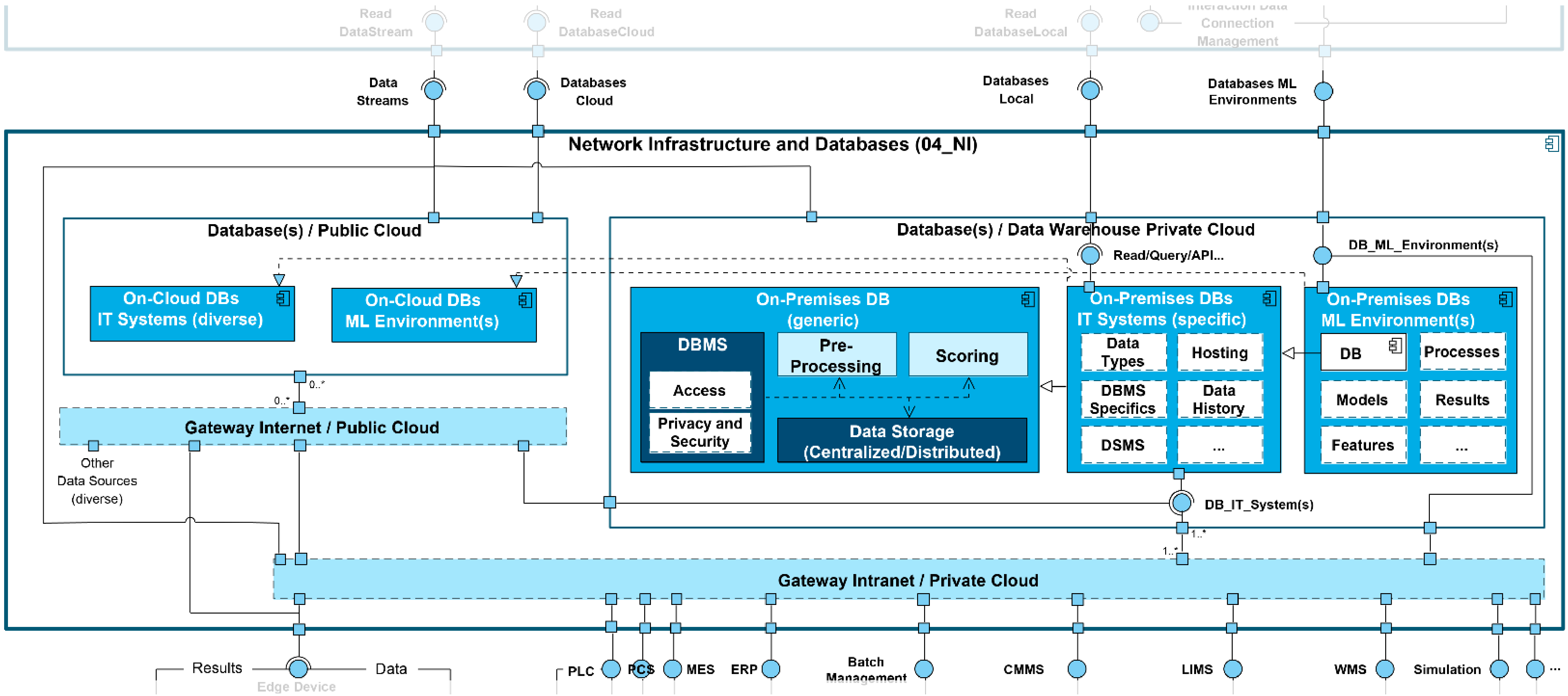
Edge Device - Notes

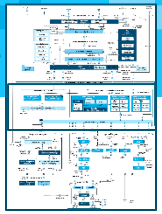
- There are numerous commercially available solutions for the design of hardware, software and interfaces of edge devices.
  - One trend is open low-cost solutions on Singe Board PCs (e.g. Raspberry Pi or Arduino) and the use of free software such as Node-RED. All common network interfaces such as WLAN, LAN, BLE or 5G as well as application protocols such as MQTT, UPC-UA or HTTPS can be used as network connection standards.
  - In a second trend, edge devices are increasingly becoming part of the solution portfolios of machine and plant manufacturers, for example Syskron ReadyKit, Bosch XDK or the SIMATIC 2000 series. The software functions either follow the provider-specific platforms (e.g. Share2Act), or hyperscalers such as AWS or Microsoft Azure increasingly offer comprehensive and user-friendly software functions for the deployment and management of edge devices and ML-based processes.
  - NVIDIA Jetson also offer comprehensive solutions for specific image data processing requirements. The following figure shows an excerpt of possible designs of edge devices and their components. (Zietsch et al. 2019) also present a selection support.
- The technology is useful when existing PLCs cannot be integrated into network structures and when scoring close to or inside an Asset is useful. Application areas are, for example, condition monitoring and predictive maintenance, which include the evaluation of higher-frequency sensor data (e.g. vibration or oscillation data or current curves), as well as video and image data processing.
- The transitions to the cloud are becoming increasingly fluid as a result of the Fog Computig, as shown by developments such as the Edge Data Center and ONCITE (German Edge Cloud) from Rittal.

Edge Device - Solution Options

		Exemplary design options							
Edge Device	Interface Edge Device to Network Infrastructrure (OSI: Application)	MQTT	AMQP	OPC-UA	HTTPS	CoAP	MTConnect	XMPP	
	Interface Edge Device to Network Infrastrucutre (OSI: Network)	WLAN	LAN	BLE	5G	6LoWPAN	ZigBee	LTE	
	Data Security Mechanisms	SSL/TSL	MQTT Server Certificate	Azure IoT Edge Security	AWS IoT Device Defender	TPM 2.0	PKCS #11	X.509	
	Device Management (central)	balena	AWS IoT Device Management	Azure IoT Device Agent	Coiote IoT Device Management Platform	Dell Edge Device Manager	SageMaker Edge Manager	Cisco IoT Edge Device Manager	
	Device Management (decentral)	div. Raspberry Pi OS	AWS IoT Greengrass Core	Azure IoT Edge Runtime	div. Container Engines	Windows IoT	FreeRTOS	Contiki-NG	
	Decentral Scoring	Node-RED	AWS Lambda Functions	Azure IoT Edge-Modules	RapidMiner Real Time Scoring Agents	TensorFlow Lite/JS	IBM Edge Application Manager	CPU/GPU/TPU SDKs	
	Decentral Data Storage (Software)	SQLite	MongoDB	Aure SQL Edge	influxDB	EdgeDB	Oracle Berkeley DB	Badger	…
	Decentral Data Storage (Hardware)	HDD	SSD	(Micro) SD	External (USB)	In-Memory/RAM	NAND-Flash		
	Decentral Data Acquisition and Pre-Processing (Software)	Node-RED	AWS IoT Greengrass Solution Accelerator	Azure IoT Edge	Siemens SIMATIC Software	Python	ROS	div. Pipeline-Tools (Kubeflow, Airflow,…)	
	Edge Device Hardware	Raspberry/ Banana Pi	NVIDIA Jetson	Harting Mica	Siemens SIMATIC IoT 2050	PLC (e.g. S7-1500)	Bosch XDK	WAGO Edge Computer	
	Interface Edge Device to PLC	OPC-UA	Modbus	COM3	RS-232	Profibus	IO-Link	HTTPS	
	Interface Edge Device to Sensor	RS-485	I²C	CAN-Bus	SPI-Bus	BLE	IO-Link	USB	
	Sensor (external)	Raspberry Sense Hat	MEMS-Sensors	Current Clamp	Light Curtain	Hydrometer	div. IO-Link-Sensors	NIR-Sensors	

Network Infrastructure and Databases



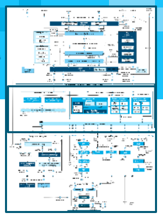


Network Infrastructure and Databases - Summary

- The network infrastructure and database layer represents an integration view of IT systems, edge devices and underlying data sources.
- The first sub-component is the gateway to the intranet and local (private) cloud environments, which combines network technologies and structures for the integration of IT systems and edge devices in internal network structures.
- Underlying databases conventionally consist of a database management system (DBMS) and the data. In addition, many database systems offer pre-processing and scoring options. Specific IT systems are often based on individual databases, which differ in DBMS, data types, data history and servers, among other things. Furthermore, databases that manage artefacts such as processes, models, features or results must also be provided for ML environments.
- High computing power is required for the application of ML. Cloud-based platforms offer advantages such as scalability and easy maintainability. The internet and public cloud gateway is therefore a module that builds on this. Challenges exist in data privacy and security as well as potential dependencies on platform providers.
- In addition to the analysis of bundled data packages (batch data) from local and cloud databases, data streams for the deployment of ML models are also of interest, depending on the application scenario.

Network Infrastructure and Databases - Notes

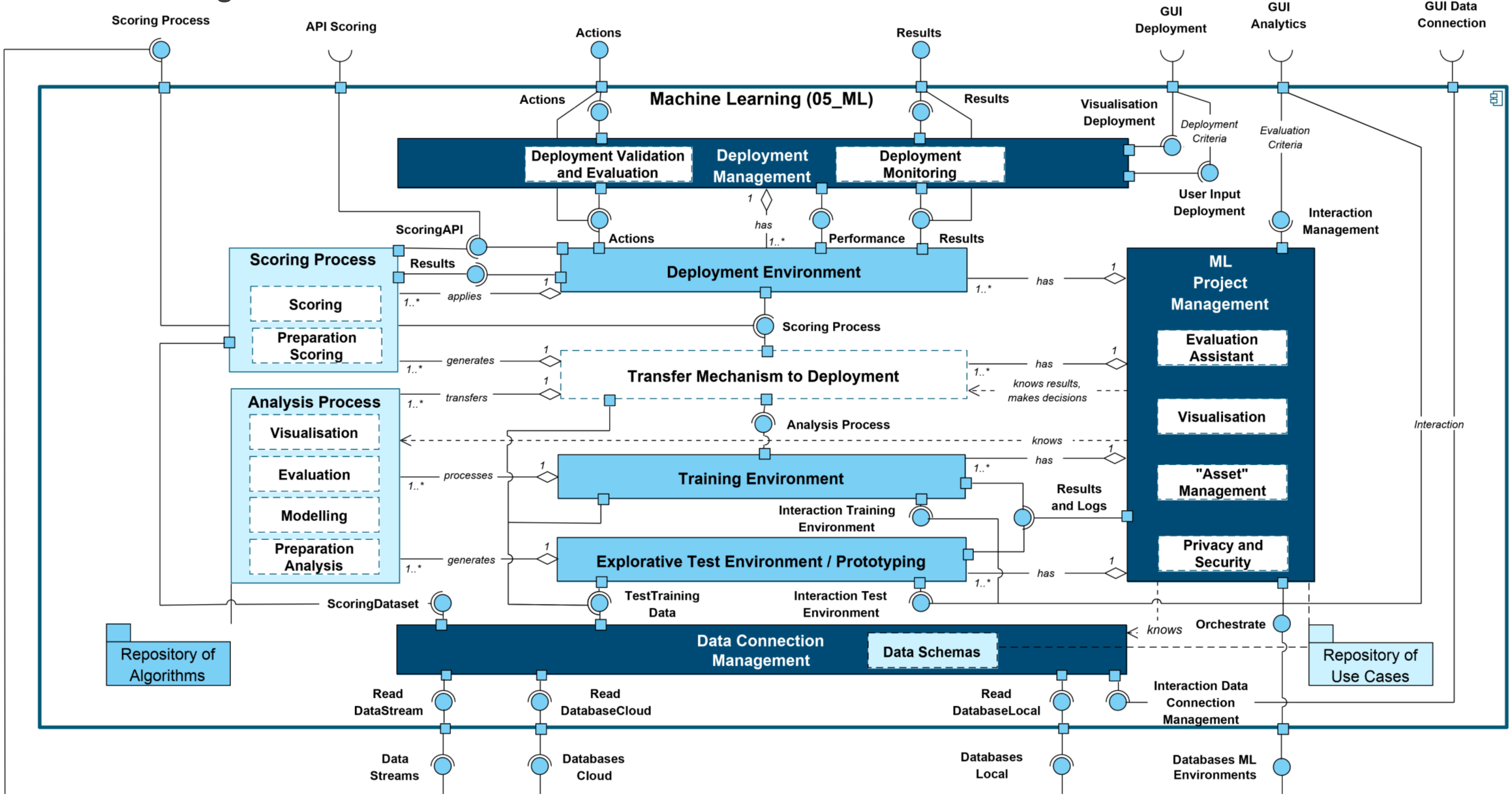
- Due to the heterogeneity of IT and network structures, there is no generally applicable solution for the selection of all components.
- The architecture does not aim to provide a selection guide as to which routers, switches, gateways, VPN tunnels, etc. should be selected in a specific case, but rather to create an awareness that precisely these paths should be analysed when setting up data pipelines, as they are not only required for training models, but must be available in a stable and near-real-time manner during deployment at the latest.
- In general, all seven levels of OSI architecture (ISO/IEC 7498-1, ITU-T X.200, X.207, ...) must be defined for each gateway connection and suitable database technologies must be selected for data storage.
- Increasingly, machine and plant manufacturers, cloud and IIoT platform operators are providing alternative solutions that need to be harmonised with company-specific IT security concepts.
- A central challenge in the design of the gateways is to ensure data security throughout, e.g. via encryption protocols such as TLL and SSL.
- The following illustration provides an overview of the possible designs. One challenge is the design of the ML backbone, as large amounts of data not only have to be connected, but also read and processed.

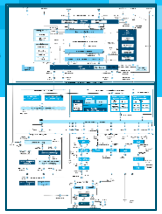


Network Infrastructure and Databases - Design Options

		Exemplary design options							
Network Infrastrucutre and Databases	Databases Public Cloud	All DBMS listed below (IaaS)	SQL Azure Server	BLOB Storages	Cloud Firestore	Amazon RDS	AWS S3	Google Cloud Databases	
	Gateway Internet / Public Cloud	Ethernet/LAN/DSL	WAN / MAN	5G	LTE	Satellite	WLAN	VPN	
	Databases/Data Warehouses Private Cloud (DBMS)	MySQL	MSSQL	MongoDB	Oracle Database	influxDB	SQLite	PostgreSQL	
		SAP HANA	Amazon Dynamo	Apache Cassandra	Hadoop HDFS (Hbase/Hive)	Kafka Streams/ ksqlDB	Elasticsearch	proprietary	
	Databases/Data Warehouses Private Cloud (Data Storage - Phyiscal)	HDD		SSD		In-Memory/RAM			…
	Databases/Data Warehouses Private Cloud (Data Storage - Network)	Single Server Architecture	Storage Attached Network (SAN)	Distributed Servers	Data Warehouse	Parallel Data Warehouse (PDW)	Massveley parallel processing (MPP)	Hadoop-Cluster	
	In-Database Pre-Processing	Data Definition Language (DDL)	Data Manipulation Language (DML)	Data Control Language (DCL)	Transaction Control Language (TCL)	Embedded SQL Statements	RapidMiner In-Database Processing Extension	Python/R integration in Databases	
	In-Database Scoring	PL/Python or R Integration	MSSQL Machine Learning Services / PREDICT T-SQL	Oracle Functions DBMS_DATA_MINING	Transfer Model in SQL Statement	IBM In-Database Machine Learning	SAS Scoring Accelerator	Google BigQuery ML	
	Gateway Intranet / Private Cloud	WLAN	LAN	BLE	NFC	6LoWPAN	ZigBee	VPN	

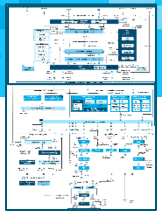
Machine Learning





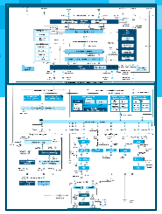
Machine Learning - Summary

- A central element is the ML layer. The basis is data connection management, which integrates local and cloud-based databases as well as data streams.
- The core is formed by three independently operating environments (test, training and deployment environment), which are to be orchestrated by an ML project management.
  - Initial analysis processes are created in the explorative test environment. The analysis process consists of the steps of data preparation, modelling, evaluation and visualisation.
  - If data connections and promising analysis processes exist, training and optimisation of models take place in the training environment. A separate environment is recommended because the training effort can be very time-consuming with large amounts of data and many influencing variables.
  - Selected models are then transferred to deployment with the help of a transfer mechanism. For this purpose, scoring processes are generated that are part of the deployment environment. A scoring process consists of a pre-processing path, the scoring itself and a required data connection.
  - The deployment environment applies scoring processes and outputs performance measures in addition to results and actions. It can be operated separately, but deployment management that orchestrates several deployment environments and thus scoring processes makes sense and is therefore part of the architecture.
- Depending on the phase of an implementation project and the environments involved, numerous interactions take place for which different GUIs exist. They start with the selection and set-up of the environments, continue with the generation and evaluation of analysis processes and end with the generation and management of scoring processes.



Machine Learning - Notes

- The design of the ML layer is a strategically relevant decision. Platform solutions are suitable for integrating the multitude of possible IT systems and data sources over long and complex production processes in the process industry. The question of local or cloud-based hosting must be answered, taking into account the effort, convenience and security of cloud-based solutions versus the effort, security and data sovereignty of local solutions - always taking into account the expected costs as well.
- Python and R represent cost-effective and lightweight solutions, but they require ML skills. An easier entry is offered by graphical software tools such as RapidMiner, which are more recommended for users in the manufacturing industry without previous ML experience. Mathematical expert systems appear less promising, but can be used if they are already being applied, for example, for process simulation. In addition, large platform solutions such as MS Azure or AWS could also provide a basis for the integration of Python, R or other software environments such as RapidMiner in the sense of PaaS or IaaS.
- Especially with regard to model management and maintenance, it is important to implement various validation and evaluation steps and to continuously monitor deployments. Especially in the process industry with constantly changing raw material characteristics and thus unknown framework conditions, ensemble strategies often pave the way for robust algorithms in deployment (Kadlec and Gabrys 2009).



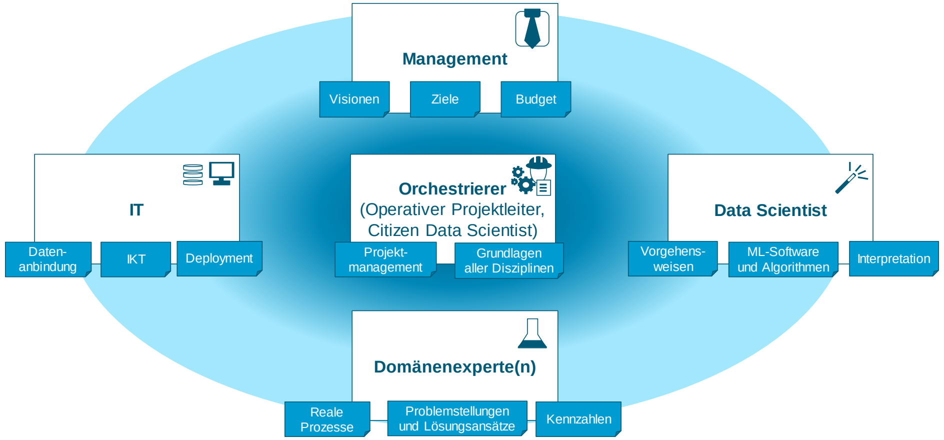
Machine Learning - Design Options

Machine Learning Environments and Modules

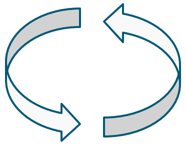
	Microsoft Azure	Google Cloud AI Platform	Amazon SageMaker	SAP Data Intelligence	IBM Watson/ Cloud Pak for Data	RapidMiner	MLflow	KNIME	TensorFlow Extened (TFX)	Dataiku DSS	DataRobot	Kubeflow	Apache Spark	Others: Open Frameworks, Tools & Librarys	
Deployment Management	Machine Learning Operations (MLOps) and Azure DevOps	MLOps resp. AI Platform Pipelines, Beta-Version, Kubeflow Pipelines and TFX based	SageMaker Model Monitor, SageMaker Pipelines	Data Intelligence Monitoring	Watson Knowledge Catalog	Diverse Deployment Management option based on AI Hub: Dashboards, Monitoring, Model Ops	MLflow Model Registry for Model Management, external Deployment Environment required	KNIME Model Factory (Pre-configured KNIME-Workflow)	Usage of management services of the piplenine administrating environment (Kubeflow Pipelines, Apache Airflow, Apache Beam)	Automated node or API execution, API Deployer for management, add. container-based services and Kubernetses support	MLOps (extensive functions)	Kubeflow Pipelines: Management of deployments on Kubernetes clusters Kubeflow Feast (Alpha-Version)	-	Django, Pyramid, Flask, Bottle, Bitbucket, Jenkins, web2py, Zope, Jetpack SDK, DeepStream SDK, NVIDIA Triton…	
Deployment Environment	Diverse deployment options, Azure Machine Learning Pipelines up to external model exports	AI Platform Prediction, hosting on Google Cloud, REST API to platform, web-based API requestst for scoring	Deployment of endpoints in S3-Cloud, AWS, AWS Lambda Functions for requesting and scoring, also external deployment possible	SAP Cloud Environment	IBM Cloud Pak for Data, deployments as web services, batch predictions or realtime streaming predictions	Hosting on RapidMiner AI Hub, Scoring Agents, Web Services, also dezentrale deployments possible	MLflow Models for export and deployment of models Model: Input and output schema, support of all common model formats, External (concrete) Deployment Environment required	Hosted on KNIME Server, scheduled or REST-triggered	TensorFlow Serving for server-based deployment, TF Lite for mobile deployment or IoT applications, TFJS as light-weight JavaScript	Diverse deployment hosting options, batch execution of workflows or API nodes for external deployment			ML Pipeines APIs as well as Spark Streaming		
Scoring Process	Diverse model export functions, e.g. ONNX, also deployment options with Azure Container, Kubernetes Services, REST-based HTTP(S) endpoint possible	Models (e.g. pickle-Files) in versioned containers	Python-based models and model groups Neo Compilation Jobs for SageMaker endpoint or AWS IoT Greengrass device	Proprietary models	Proprietary models, export with CoreML possible	Proprietary models, limited export options as PMML	Integration of all common model formats, add. establishing new model format (Mlflow Models)	Proprietary models	TensorFlow Model dormat, exportable in TensorFlow Serving and as JavaScript Object (TFJS), Docker and YAML export of pipelines	Scoring Recipes (proprietary models), external API nodes, support of Phyton, Spark and SQL engines	Proprietary models, export in Hadoop as well as JAR-based scoring code possible, Import of models (e.g. pkl) for deployment	YAML-storage of pipelines, support of the environments of KFServing, Seldon Core, BentoML, NVIDIA Triton, TensorFlow Serving, TensorFlow Batch Prediction	PMML export for a small amount of models in MLlib, further depending on programming environemnt	Export of models in pickle files (.pkl), add. package specific export formats possible (e.g. Joblib oder XGBoost)	
ML Project Management	Azure Machine Learning Workspace	Google Cloud Project	SageMaker Studio	ML Scenario Manager (Notebooks, Pipelines, Executions and Models)	Watson Studio projects	RapidMiner AI Hub, RapidMiner Projects	MLflow Projects and MLflow Tracking for managing projects and experiments Platform for integration of Python, R, Java and Scala languages as well as common ML librarys and frameworks like scikit-learn, TensorFlow, PyTorch, Keras, Spark as well as environments like Azure, SageMaker, H2O, …	KNIME Server	Construction and management of pipelines, artifacts and experiments in Airflow, Kubeflow or Beam	Dataiku Projects, platform-based	DataRobot Project on platform	Kubeflow UI for management of pipelines and notebooks	-	JupyterLab, Jupyter Hub, Git-basierte Rstudio-oder Python-Projekte, Coment, Neptune, …	
Analysis Process	Proprietary format in Machine Learning Studio, Integration of Python und R code possible	Python Virtual Environment, code-based	SageMaker Notebooks (Jupyter Notebooks), code-based	Graphical programming in Modeler, integration of Python notebooks	Graphical programming in SPSS Modeler or code-based in Jupyter Notebooks	Graphical programming in RapidMiner Studio integration of Python and R code possible	Code-based	Graphical programming in KNIME Workbench resp. Workflow Editor	Jupyter Notebooks, code-based construction of analysis pipelines, TFX Python packages	Graphical programming of Analysis Processes, web-based GUI	Automated, configurable construction of Analysis Processes	Code-based (Python, R)	Code-based in programming environments of Scala, Java, Python, R by using the APIs of MLlib	Code-based (Python, R,...)	…
Repository of Algorithms	Limited own repository, integration of common Python and R frameworks, languages and extensions	Support of TensorFlow, PyTorch, R, scikit-learn, XGBoost, content of Deep Learning Containern in diverse NVIDIA Packages (CUDA, CuDNN, NCCL)	Integration of common Python and R frameworks and extensions, add. SageMaker JumpStart as Reopository of Use Cases	Integration of common Python and R frameworks and extensions	Support of frameworks like PyTorch, TensorFlow and scikit-learn as well as R, Phython and Scala languages	Extensive Repository of Algorithms integrated, diverse integration options of extensions as well as Python ad R-codes	Python and R incl. diverse repositories, add. Java API	Integrated Node Repository in KNIME Workbench	Integration of all common Python librarys and extensions	Visual Rrecipes (Integrated repositores), Code Recipes (Python and R integration)	Diverse integrated models (Model Blueprints), but only few extension or import options and low degree of freedom	Integration of all common Python librarys and extensions	Machine Learning Library (MLlib) and diverse APIs to common frameworks	SciPy, Pandas, IPython, Anaconda, TensorFlow, Keras, Scikit-learn, PyTorch, cuDNN, DALI, TLT…	
Training Environment	Azure Machine Learning Studio/Designer, Learning Environments, R and Python SDK integration, distributed training with PyTorch an TensorFlow integration options	AI Platform Training, AI Platform Deep Learning Containers, AI Platform Deep Learning VM Image	SageMaker Experiments, Distributed Training, Debugger SageMaker Neo as transfer mechanism to desired hard- and software environment	Data Intelligence Modeler	Watson Machine Learning Deployment Space for configuration, training, testing and transfering models into deployment	RapidMiner Studio, local or server-based instances possible, Radoop extension for executing jobs in Hadoop clusters	No native functions, environments are loceted locally or on platforms	KNIME Analytics Platform	TensorFlow, hosting of TFX Pipelines on Apache Airflow, Apache Beam or Kubeflow Pipelines, Keras integration for Deep Learning	Data Science Studio (DSS), Usage of ML engines of scikit-learn/XG-Boost, MLLib, H2O or Vertica on hosted environments	Gudided, automated creation of Analysis Processes on DataRobot platform, API for accessing the functions in own Python clients possible	Kubeflow Fairing to train (local) Python codces in cloud environments on remote, support of the Chainter, MPI, MXNet, PyTorch and TensorFlow (TFJob) frameworks	Spark background Engine for cluster computing in Java, Scala, Python- or R-based environments Spark framework on Hadoop YARN, Apache Mesos, Amazon EC2, Kubernetes , standalone usable	see below, hosted on more powerful computing environments	
Explorative Test Environment / Prototyping	Azure Machine Learning Studio, Learning Environments, Experiments	AI Platform Notebooks: scalable hostet VM instance with pre-configured JupyterLab environment	SageMaker Studio Notebooks, scalable hosted Jupyter Notebooks	Data Intelligence Modeler	SPSS Modeler or Python notebooks in Watson Studio	RapidMiner Studio		KNIME Analytics Platform	TensorFlow Transform, add. Model Validator and Evaluator functions as well as visualisation	DSS, functions for accessing and exploring data		Jupyter notebooks, management of experiments and pipelines in centra UI		Jupyter Notebooks, RStudio, SAS Modeler, MATLAB, JMP …	
Data Connection Management (Batch)	Multiple modules possible, first Azure Machine Learning Datasets, also Azure Data Catalog, Resource Manager, Resource Provider	Only rudimental management functions, currently beta version of MLOps (AI Platform Pipelines)	SageMaker Data Wrangler, SageMaker Feature Store	Data Intelligence Connection Management	Watson Knowledge Catalog	Data Connection Management, definition of Data Schematas in DaPro Extension		Nodes resp. operators for accessing local and server/cloud databases, Big Data extension	TensorFlow Data Validation, add. Services of the resp. pipeline einvironment	Extensive interfaces to common data sources, definition of Data Schematas possible	Paxata Data Prep	Tendentially rudimentary, depends of the Kubernetes environment as well as underlying platform (Google Cloud, AWS, Azure, IBM, On-Premises, Local) Metadata-Module	Spark SQL and Data Frames for connections in Scala, Java, Python und R		
Data Connection Management (Stream)	Azure Stream Analytics (Jobs)	Only rudimental management functions JSON-Strings as request on web-based API	SageMaker Pipelines, add. diverse Edge Device deployment options	Data Intelligence Connection Management	Steaming Analytics environment (Beta) in IBM Streams, add. Streams Flow in Watson Studio	No native functions, Streaming extension (Alpha) for Flink and Spark-based stream analyses as wenn als Kafka connector	Depending on Deployment Environment (not native), recommended locally, Azure ML, SageMaker and Spark	No native functions, KNIME streaming execution extension (Beta)	Apache Beam as fundament of stream pipelines	-	No native functions, focus on web-based endpoints and APIs	Kubeflow Feast (Alpha-Version) for Feature Storage, Management and Serving (Batch und Stream-Daten)	Spark Streaming		

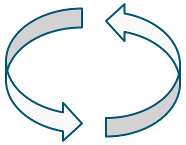
Application - Using the Architecture

Roles in implementation projects

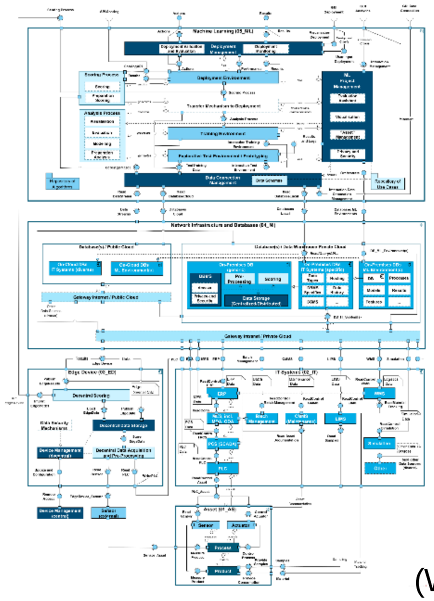


(Deuse et al. 2021)

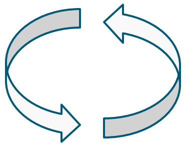




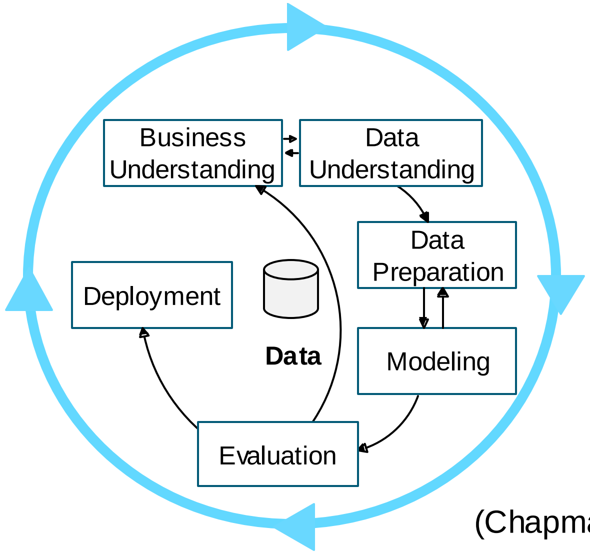
Reference Architecture



(Wöstmann et al. 2020)



Phases in implementation projects

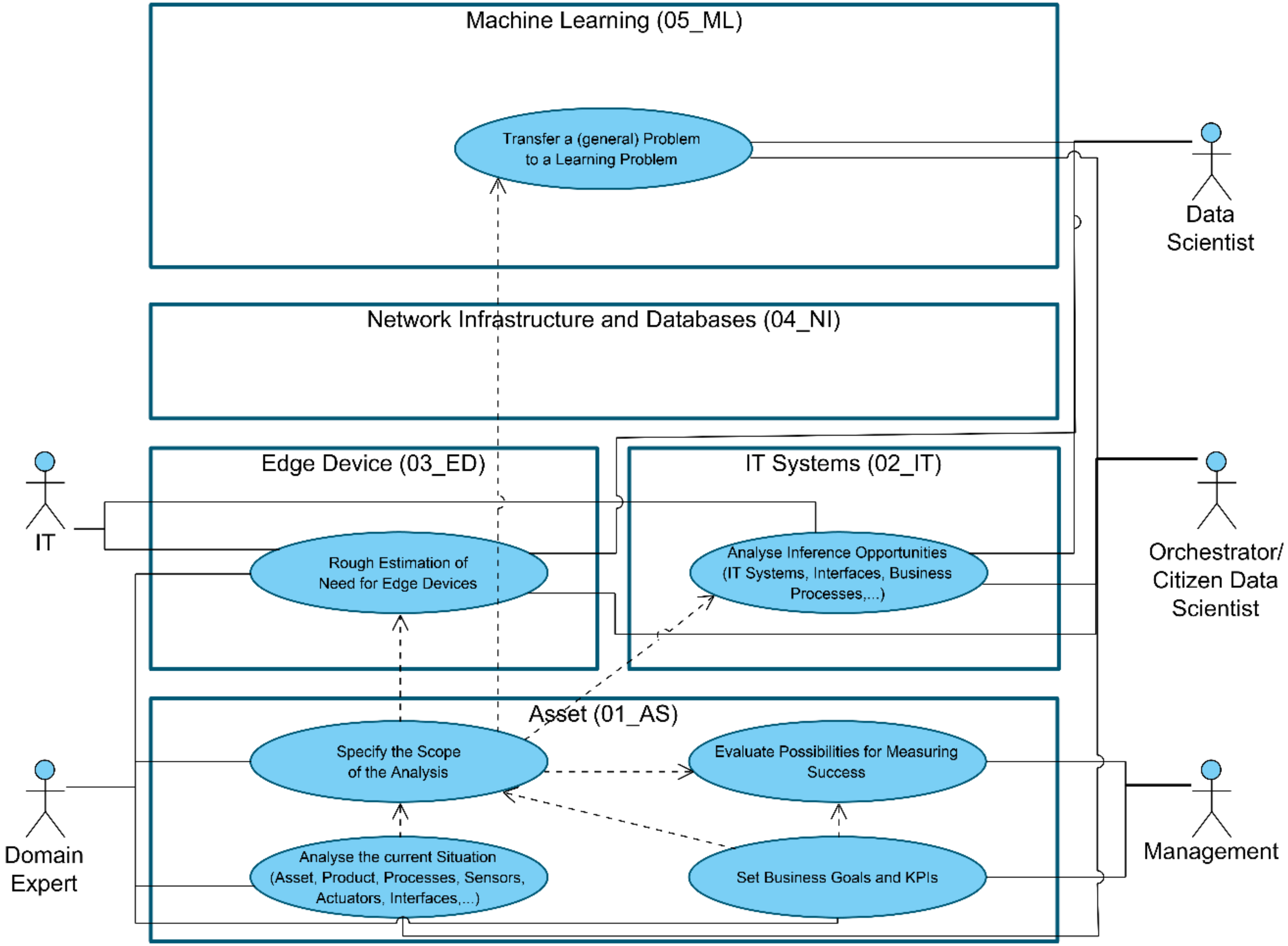


(Chapman 2000)

Application - Summary

- The application layer reflects the interaction of the architecture with the users and the integration of the results and actions into business processes and IT systems.
- The starting point for ML-based product and/or process optimisation is an operational challenge for which specific KPIs and goals must be defined in Business Understanding.
- The use of the architecture starts with an interface to the data connection management, which enables the initial connection to existing IT systems and underlying databases in the following phases Data Understanding and Data Preparation.
- In the subsequent phase of modelling and evaluation, analysts or (citizen) data scientists can access the exploratory test environment, the training environment and ML project management via an interface.
- Through interfaces to the deployment environments, model results and, if necessary, actions are made available and can then be integrated into IT systems and/or business processes.
- Separate access to the deployment management should ensure that created scoring processes are executed in a separate environment and that access options exist for further user groups such as specialist departments or management, but also IT systems and facilities themselves. In addition to user interaction, scoring processes can be triggered by a separate API, and there is also the option of exporting scoring processes to enable application-oriented (e.g. edge device-supported) deployments.

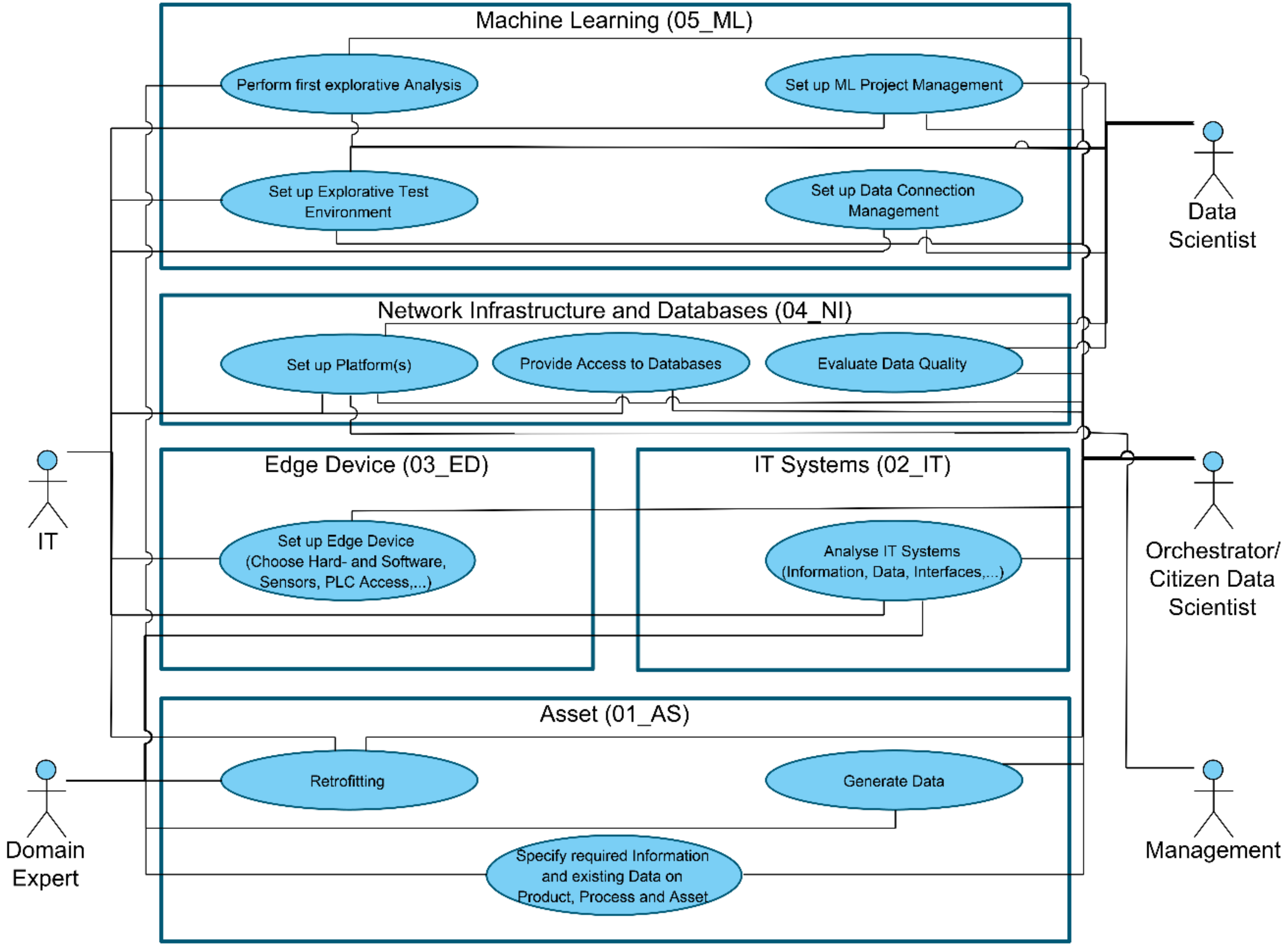
Application - Business Understanding



Application - Business Understanding

- The basic activities of business understanding address the creation of an understanding of products, processes, systems, goals and problems.
- The reference architecture supports as-is analyses, the definition of the focus as well as the derivation and discussion of potential machine learning problems and tasks.
- At the asset layer, problems are analysed and goals and KPIs are defined.
- The management is responsible for a strategic perspective in the constitution of a common vision, the composition of the project team and the provision of resources (e.g. budget and time).
- Domain experts play a decisive role in the analysis of the current situation, the derivation of goals and the delimitation of problems. In this context, a rough analysis of the IT systems is also carried out with the involvement of the IT departments in order to address potential intervention possibilities of ML and their requirements for the solutions to be developed at an early stage.
- At the edge device layer, domain experts and IT discuss whether the use of edge devices is an option or a requirement, or whether it should be neglected.
- The role of the data scientist supports the translation of general problems and goals into potential machine learning problems in order to obtain an early assessment of the possibilities and limitations of the application of ML. Citizen Data Scientists act as a coordinator and orchestrator of the actors and take on important project management functions.

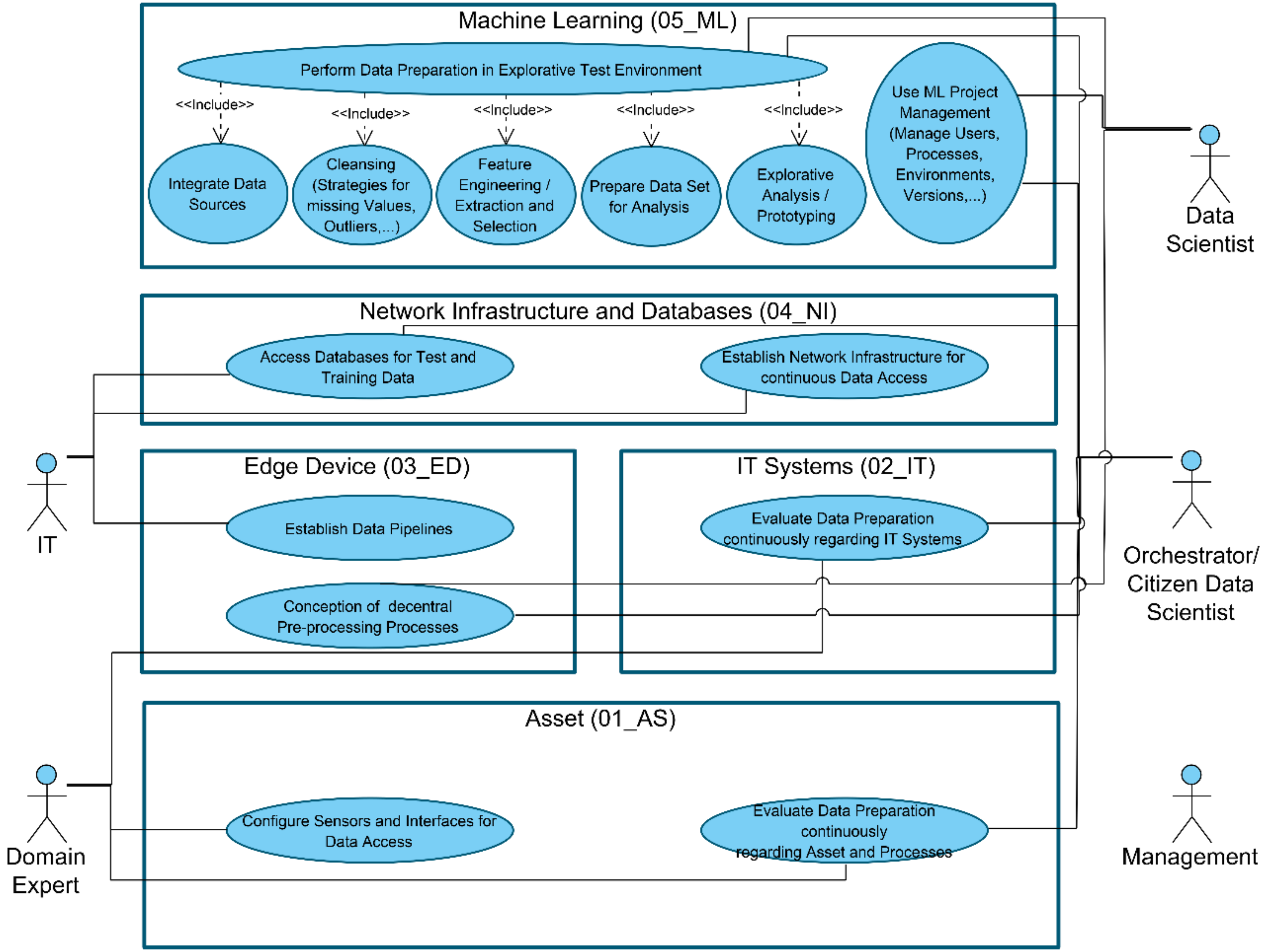
Application - Data Understanding



Application - Data Understanding

- The tasks of data understanding consist of building up an understanding of the relevant data sources for (possible) influencing and target variables of an identified problem as well as a more detailed assessment of data quality and quantity.
- In order to specify the information and underlying data on product, process and plant (components), existing IT systems must be analysed in detail and the necessary database access must be established.
- Important players are domain experts who know asset and IT systems from a user perspective, as well as IT departments that help with the implementation of data connections and ML environments.
- An initial exploratory assessment of data quality requires an environment of (exploratory) data analysis. Therefore, it is recommended to implement an exploratory test environment, data connection management and ML project management already in the data understanding phase.
- The reference architecture provides an overview of the general solution space. Since working with data extracts is recommended, high computing power is not necessarily required. While management may be involved in the selection of platform solutions, the Citizen Data Scientist orchestrates the essential activities, such as the assessment of data quality.
- The data scientist supports the selection and installation of the environments as well as the explorative analyses and visualisations - which in turn have to be interpreted together with domain experts.

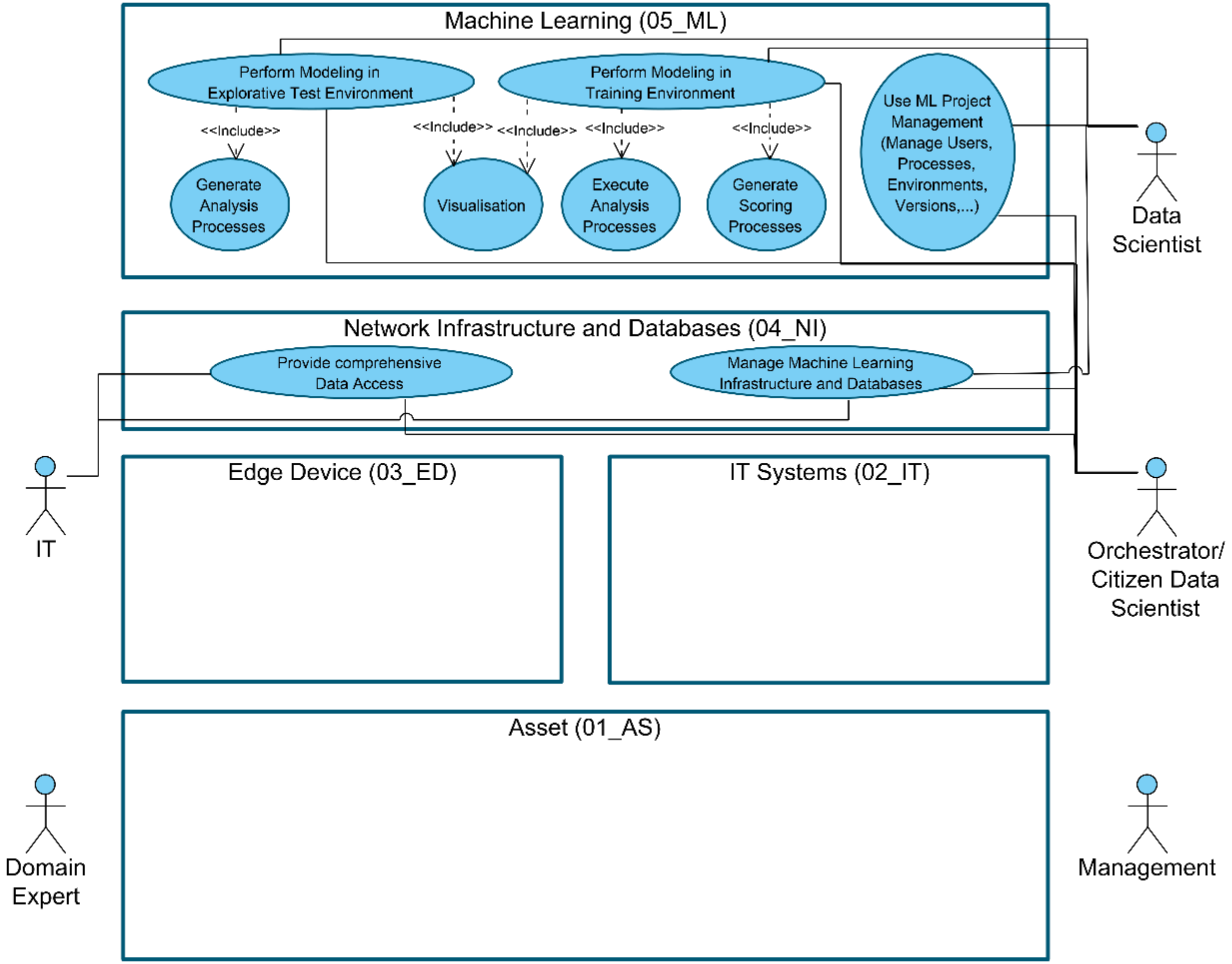
Application - Data Preparation



Application - Data Preparation

- Data preparation addresses all relevant activities for preparing data sets and connections for training models.
- The development of data connections begins at the asset level, where access to system and sensor data as well as interfaces for data access must be prepared.
- Data connections must also be set up at the edge device level, if planned, with the involvement of the IT departments.
- The core tasks at the network infrastructure level consist of establishing access to data sources selected in the data understanding and at the ML level in carrying out the data preparation steps. For this purpose, the exploratory test environment must be set up as a working environment for setting up analysis processes, as well as a suitable ML project management for managing users, processes, versions and environments.
- The core tasks are performed by (Citizen) Data Scientists and consist of merging heterogeneous data sources, data cleansing (e.g. removal of incomplete values, handling of outliers, estimation of missing values, ...), feature engineering as well as providing a suitable data set and pre-processing processes for the training environment.
- Domain experts are involved in evaluating the steps, and iterations are common practice, especially with the modelling and evaluation phases.

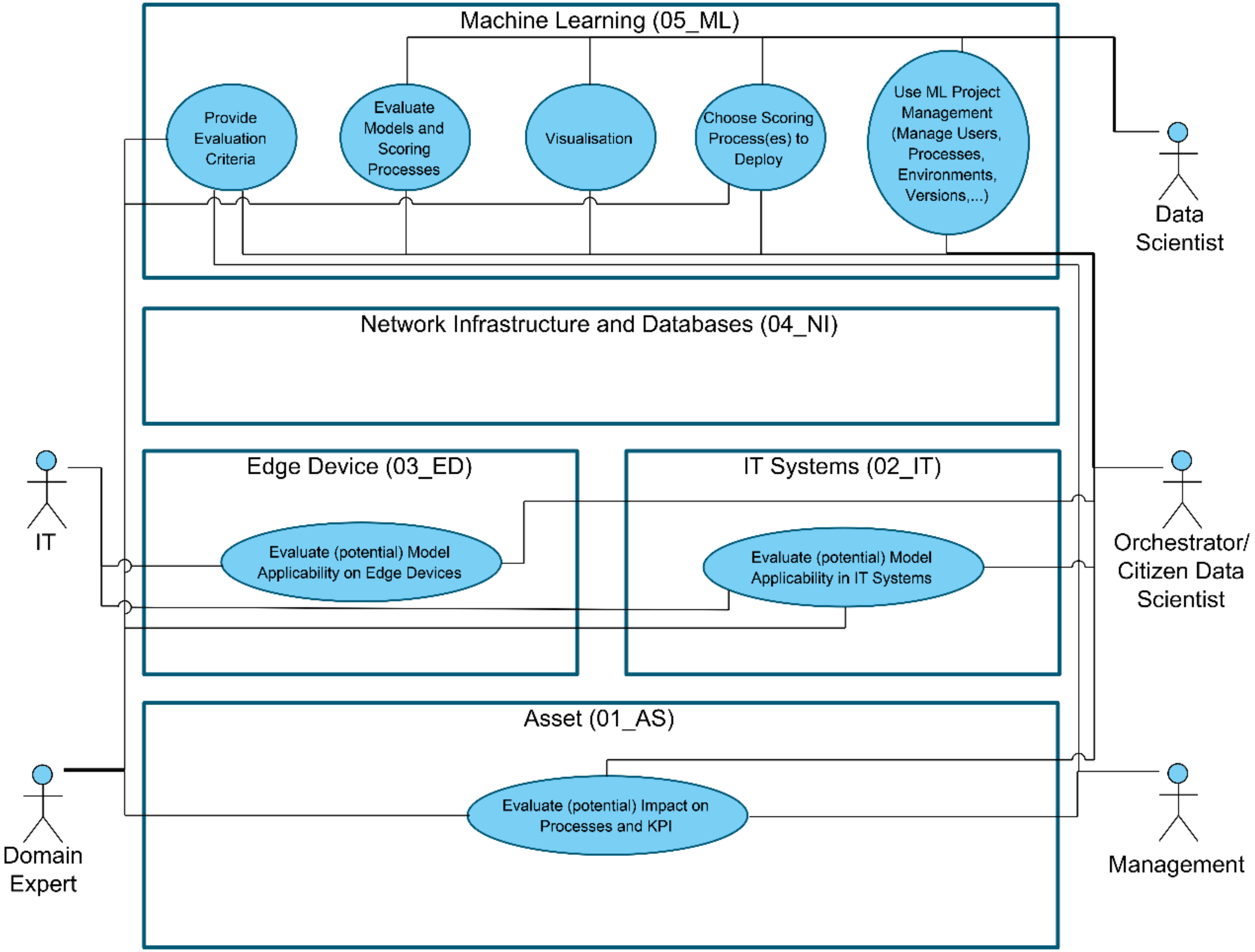
Application - Modeling



Application - Modeling

- In the modelling phase, different modelling approaches are selected for the learning problems discussed in Business Understanding and transferred into tasks for training models.
- In this process, analysis processes are set up and executed using the largest possible database.
- On the one hand, the explorative test environment is used, in which the first analysis steps were already created in the data preparation phase. On the other hand, the modelling phase requires greater computing power, since the test and training data are to be included in the training as extensively as possible, parallel analysis paths are created and tasks such as hyperparameter optimisation will also arise.
- At this point, at the latest, a training environment must be set up and used that can be hosted either on-premises or on the cloud. The reference architecture introduces an overview of alternative solutions for this, which can, however, be individually developed or extended.
- Furthermore, the use of an ML project management for the administration of experiments, processes, environments and users is recommended and should be set up accordingly.
- The essential modelling tasks are carried out by (Citizen) Data Scientists, supported by IT departments in providing the environments and data connections.

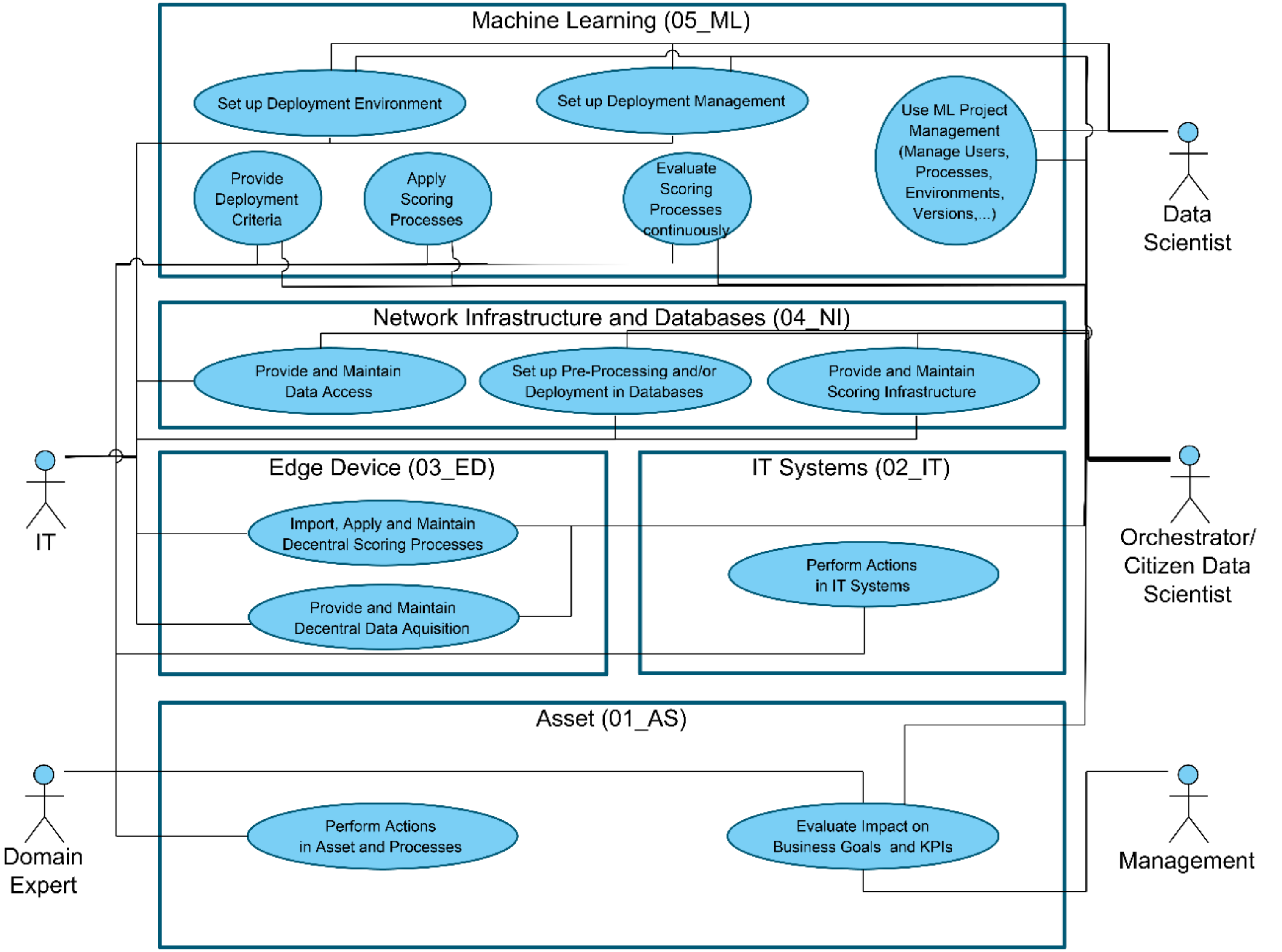
Application - Evaluation



Application - Evaluation

- The evaluation phase serves to assess the trained models with regard to performance and quality criteria. These (summarised as evaluation criteria) are dependent on the requirements of the application and IT systems and are therefore largely provided by domain experts and IT.
- Both the models and scoring processes are evaluated with regard to the requirements of the application. In addition to mathematical target values, it is also necessary to evaluate the meaningfulness and plausibility of the results obtained, taking into account the domain knowledge.
- In this step, the analysis steps and possible deployment scenarios must also be evaluated, which in turn can result in new requirements for both data collection and the application of the models.
- Visualisation also plays an important role in creating transparency and understanding.
- On a strategic level, this is followed by an assessment of the results in terms of their usefulness for the originally set goals and KPIs. Here, special focus must be placed on the applicability of the models and their integration into the IT systems and business processes.
- In addition to domain experts, management must also be involved. The goal of the phase is to select models and scoring processes that are then deployed and put into operation.

Application - Deployment



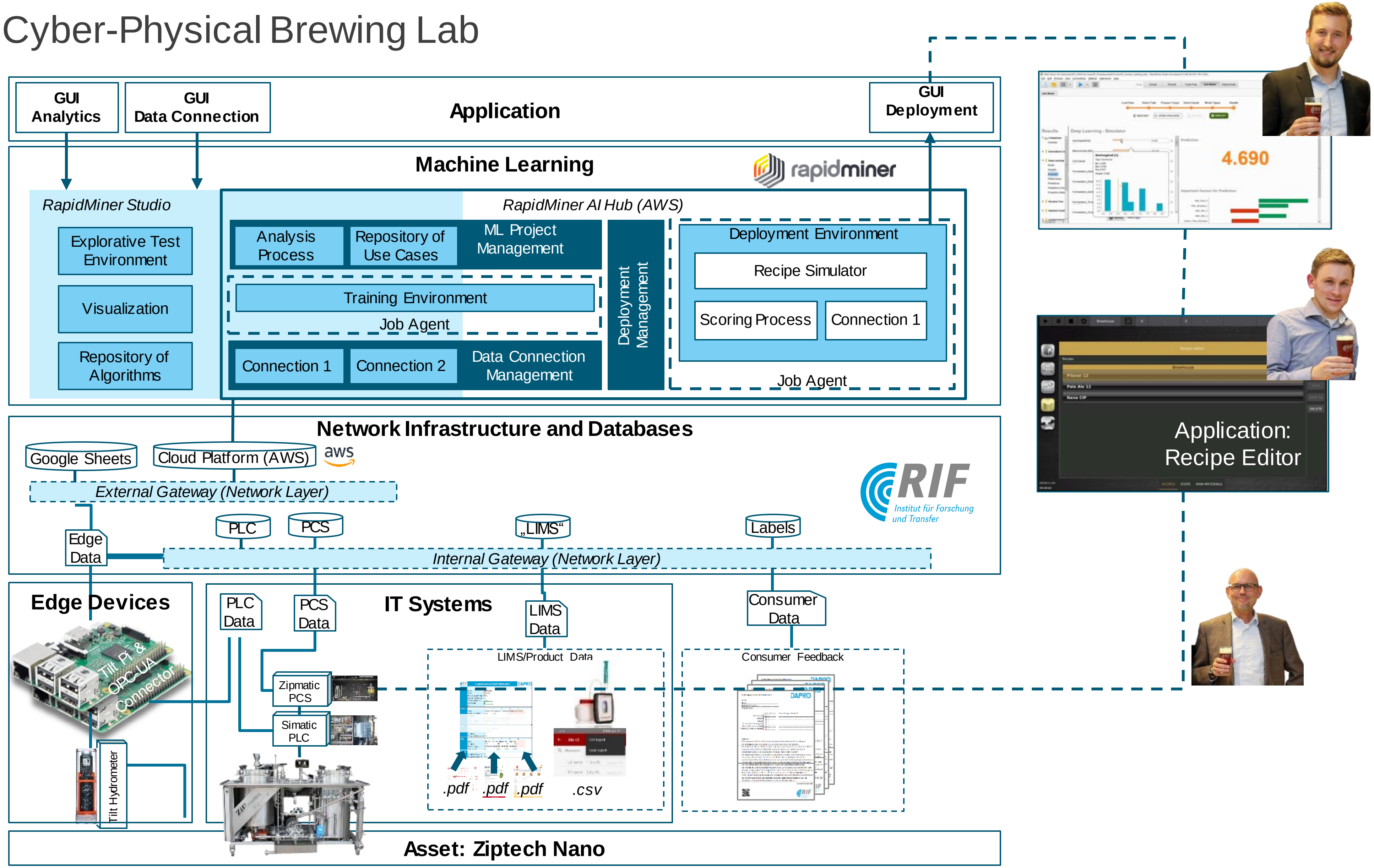
Application - Deployment

- The final phase includes the implementation of deployments and their management.
- For this purpose, the selected models and analysis processes must be transferred into scoring processes and the type of deployment must be selected. Scoring processes can run server-based, in databases, in IT systems or on edge devices and must be implemented accordingly.
- The reference architecture offers assistance in the design and solution selection of scoring processes, deployment environments and deployment management.
- In addition to the one-time setup of the environments with the involvement of IT departments and data scientists, the goal is the long-term delivery and maintenance of deployments by the users (e.g. specialist departments), who must also be enabled to carry out and maintain scoring processes in the long term. Deployment criteria must therefore be defined for continuous evaluation.
- Furthermore, strategic tasks consist of constituting and distributing the knowledge gained in the organisation, developing a strategy for deriving concrete measures based on model outputs, and integrating them into the structural and process organisation.
- For individual solution patterns, the Citizen Data Scientist takes on a central role as a company-internal service provider.
- Depending on the strategy, these tasks can also be carried out by machine and plant manufacturers or providers of data-based services.

Use Case Examples

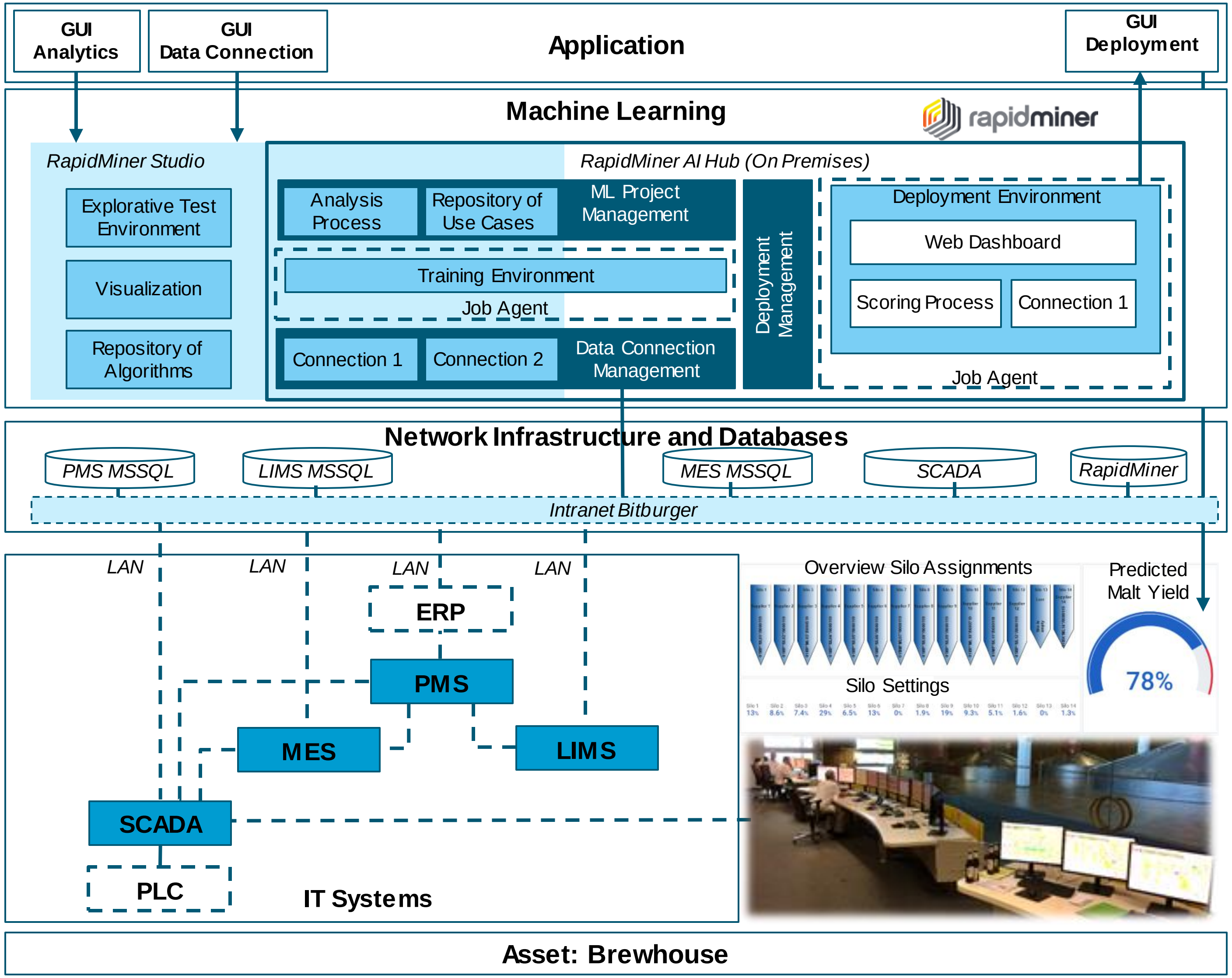
Illustrative Use Cases

Recipe Optimisation in Cyber-Physical Brewing Lab



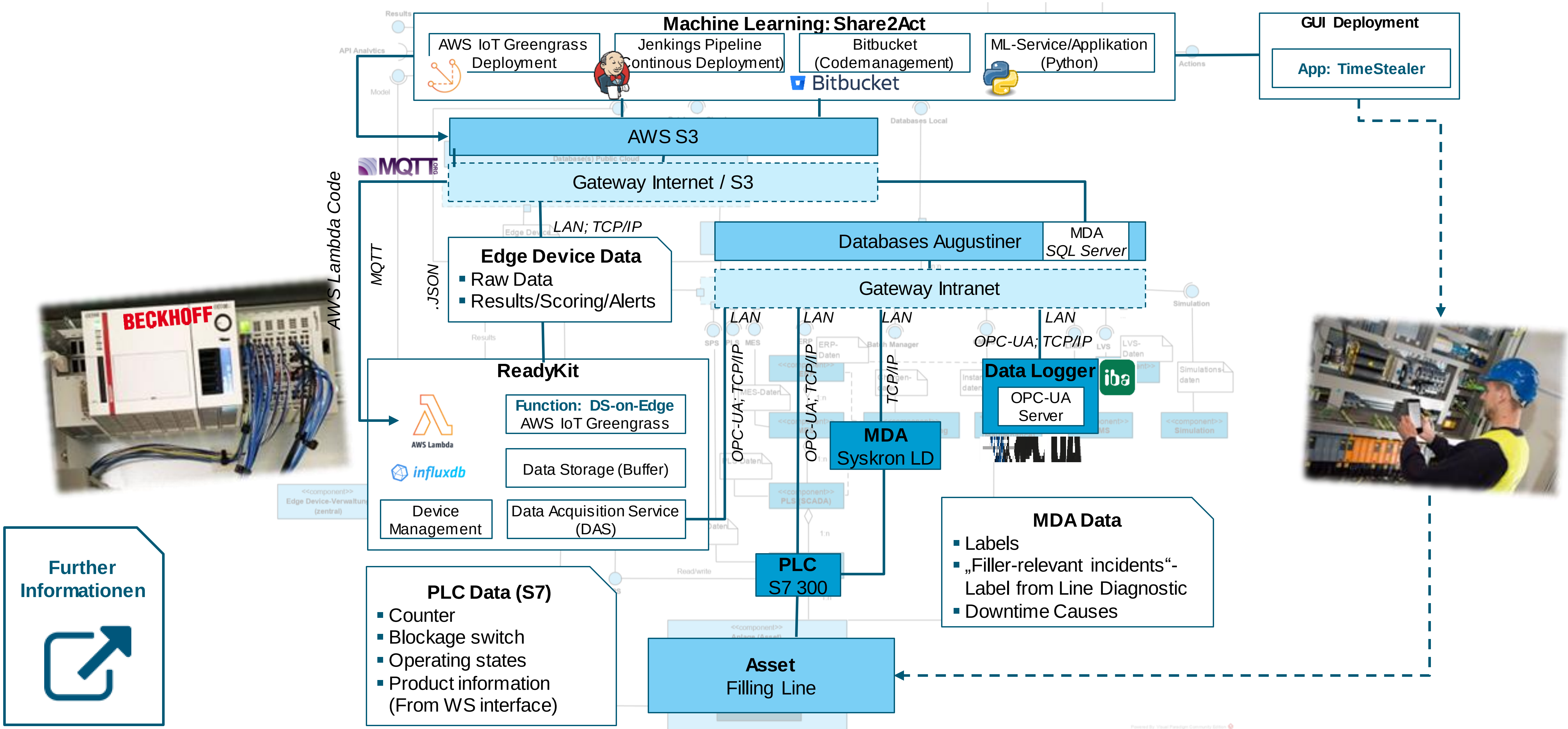
Further Informationen

Malt Yield Prediction and Optimisation in the Beverage Industry

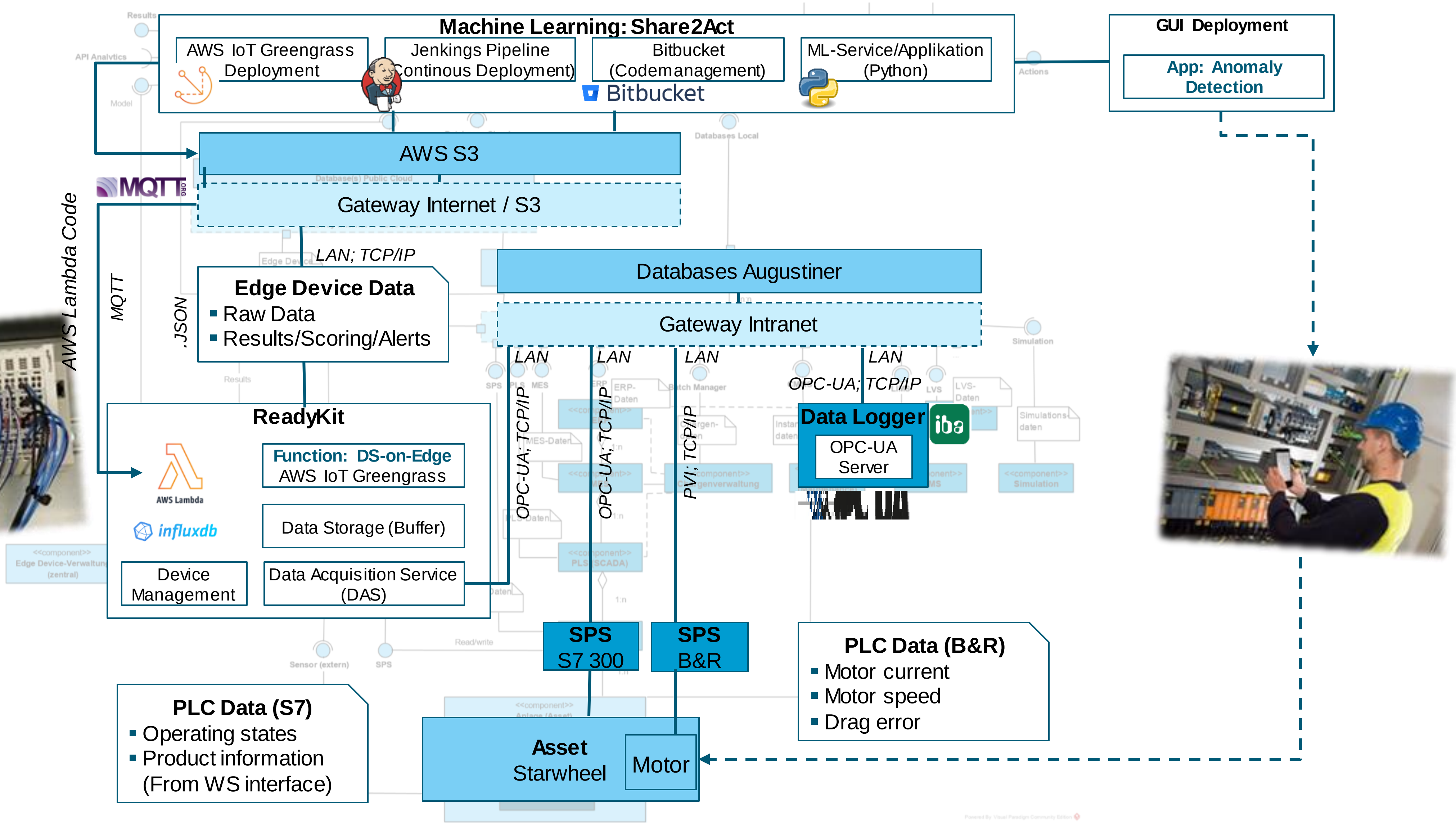


Further Informationen

Detection of bottlenecks in filling and packaging lines



Prevent downtimes via monitoring of machine mechanics



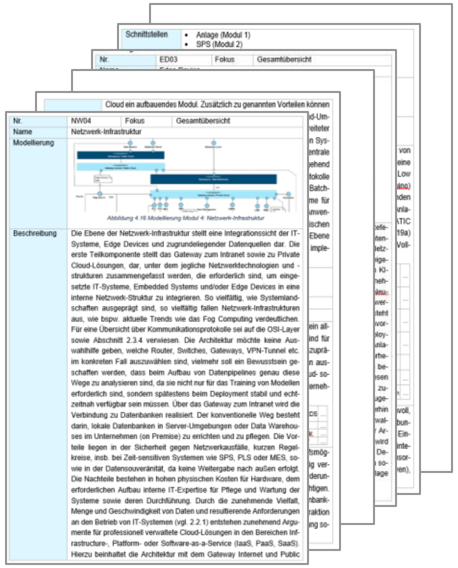
Further Informationen

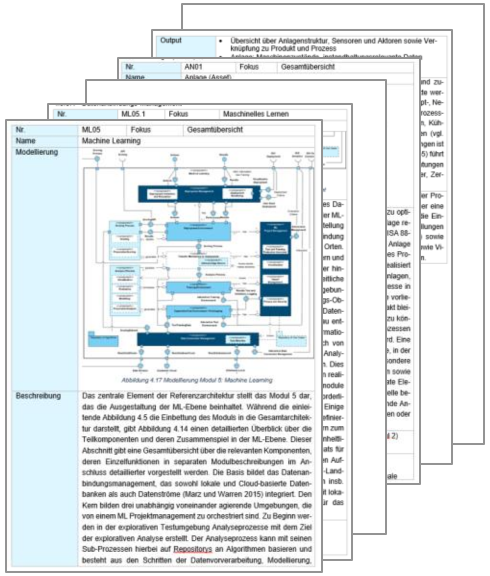
Next Steps

Next Steps

- Help us with the validation!
  - Online survey: 15-20 min. time required
  - Link: https://forms.office.com/r/UHdE9eC00U

- More detailed publication in 2022:





→ Let's get into conversation!

Literature

- ANSI/ISA-88.00.01-2010 Batch Control Part 1: Models and Terminology
- ANSI/ISA-95.00.01-2010 (IEC 62264-1 Mod) Enterprise-Control System Integration - Part 1: Models and Terminology
- Chapman, Pete; Clinton, Julian; Kerber, Randy; Khabaza, Thomas; Reinartz, Thomas; Shearer, Colin; Wirth, Rüdiger (2000): CRISP-DM 1.0. Step-by-step data mining guide: CRISP-DM Consortium.
- Deuse, Jochen; Wöstmann, René; Schulte, Lukas; Panusch, Thorben; Kimberger, Josef (2021): Citizen Data Science. Transdisciplinary competence development for role models in data-driven value creation. In: Wilfried Sihn und Sebastian Schlund (Hg.): Competence development and learning assistance systems for the data-driven future. Berlin: GITO Verlag.
- Eickelmann, Michel; Wiegand, Mario; Deuse, Jochen; Bernerstätter, Robert (2019): Bewertungsmodell zur Analyse der Datenreife. In: *ZWF* 114 (1-2), S. 29–33. DOI: 10.3139/104.112037.
- IEC 61512-1:1997, Batch control - Part 1: Models and terminology
- ISO 10628-2:2012, Diagrams for the chemical and petrochemical industry - Part 2: Graphical symbols.
- ISO 15926-1:2004, Industrial automation systems and integration - Integration of life-cycle data for process plants including oil and gas production facilities. Part 1: Overview and fundamental principles.
- ISO/TS 15926-4:2019-10, Industrial automation systems and integration - Integration of life-cycle data for process plants including oil and gas production facilities. Part 4: Initial reference data.
- Kadlec, Petr; Gabrys, Bogdan (2009): Architecture for development of adaptive on-line prediction models. In: Memetic Comp. 1 (4), S. 241–269. DOI: 10.1007/s12293-009-0017-8.
- Wöstmann, René; Schlunder, Philipp; Temme, Fabian; Klinkenberg, Ralf; Kimberger, Josef; Spichtinger, Andrea et al. (2020): Conception of a Reference Architecture for Machine Learning in the Process Industry. In: IEEE Computer Society (Hg.): 2020 IEEE International Conference on Big Data (Big Data). Atlanta, GA, 10.-13.12.2020. Institute of Electrical and Electronics Engineers (IEEE). Los Alamitos, CA, USA, 1726-1735. Online available at https://doi.ieeecomputersociety.org/10.1109/BigData50022.2020.9378290.
- Zietsch, Jakob; Weinert, Nils; Herrmann, Christoph; Thiede, Sebastian (2019): Edge Computing for the Production Industry A Systematic Approach to Enable Decision Support and Planning of Edge. In: Proceedings, 2019 IEEE 17th International Conference on Industrial Informatics (INDIN). Aalto University, Helsinki-Espoo, Finland, 22 - 25 July, 2019. 2019 IEEE 17th International Conference on Industrial Informatics (INDIN), 2019/07. Piscataway, NJ: IEEE.

Data-driven Process Optimisation using Machine Learning in the Beverage Industry (DaPro)

Contact

+49(0) 231 755 5733

+49(0) 231 755 2649

rene.woestmann@rif-ev.de

www.dapro-projekt.de