Project “FairMillData”

Stefan Trabesinger**, Franz Haas**

Thomas Klünsner*, Lukas Hanna*, Elias Jan Hagendorfer*, Manfred Mücke*

* Materials Center Leoben Forschung GmbH, Leoben, Austria

** Institute of Production Engineering, Graz University of Technology, Austria

DIHSÜD Digital Innovation Hub

FAIR Milling Dataset
for Open Milling Process Improvement

Quelle: [1]

Edge Computing als Basis für „FairMillData“

- Edge Computing: Motivation & Möglichkeiten
- Aufbau Netzwerk Architektur
- Durchgängige und integrierte Dateninfrastruktur
- Vorbereitung: Datenakquise und Aufbau Experimente
- Durchführung Experimente
- Entwicklung von Machine Learning Modellen
- Entwicklung, Implementierung und Anwendung einer Edge App zur Detektion von Bohrerbruch
- Aufbau Lernfabrik Showcase „Edge Computing“
- Zusammenfassung und Ausblick

Edge Computing

- Problem: Bohrerbruch
- Ziel: Vermeidung möglichst viele Bohrerbrüche
    - Repräsentative Charakteristik für Bohrerbrüche
- Lösungsansatz: System zur vorausschauenden Wartung

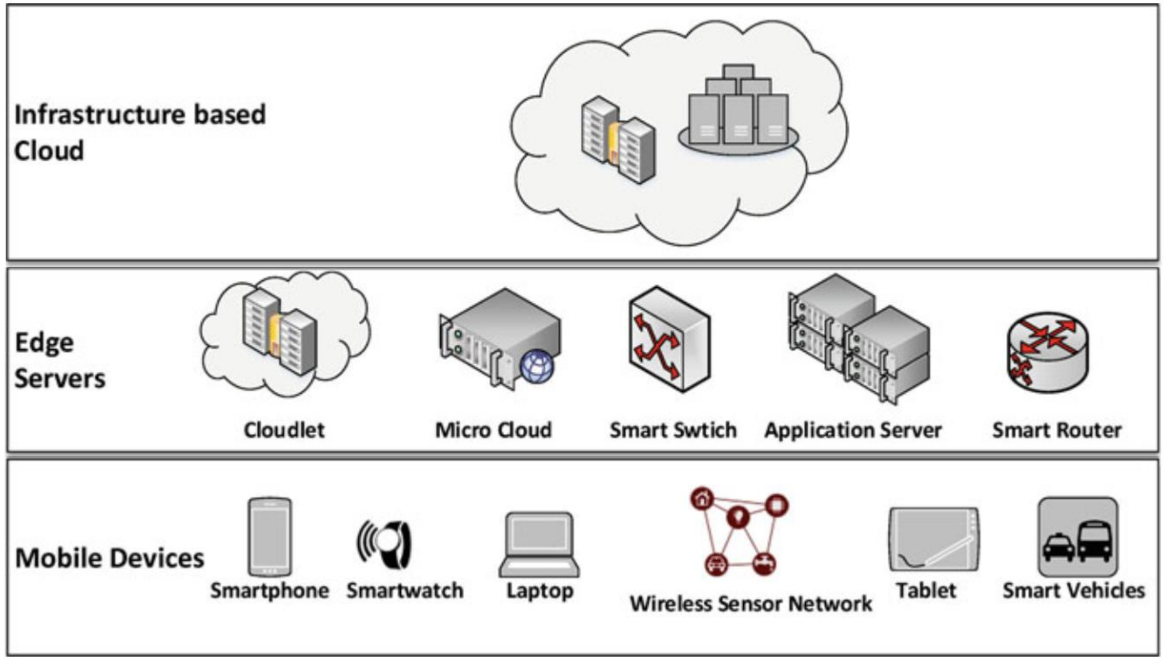
Quelle: [2]

Edge Computing

- Verteiltes Cloud Computing Paradigm
- Reduzierte Datenmenge in die Cloud
- Datenvorverarbeitung nahe der Datenquelle
- Lokale Analyse für schnelleres Feedback
- Entlastung der NCU der Werkzeugmaschine

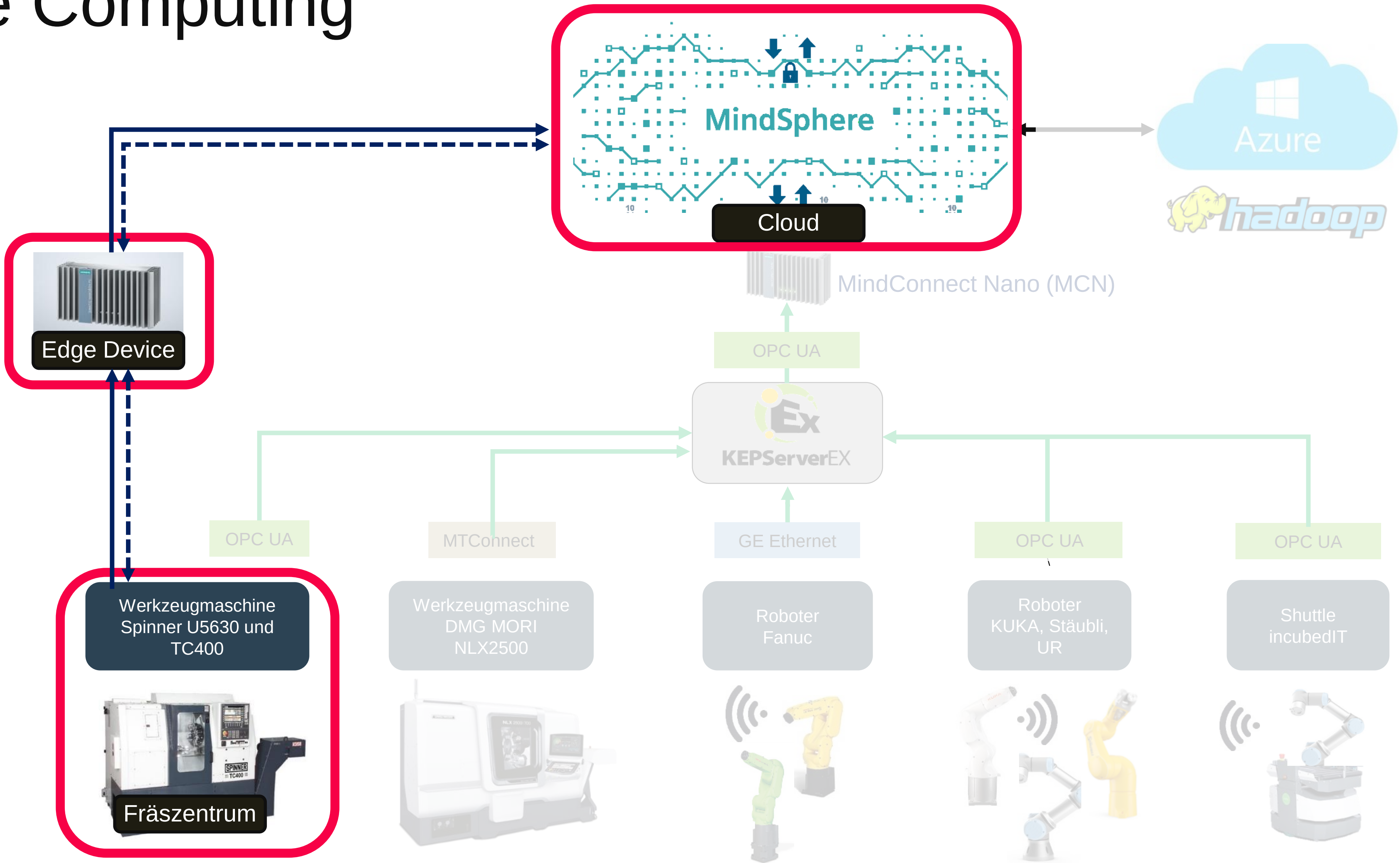
Quelle: [3]

Edge Computing



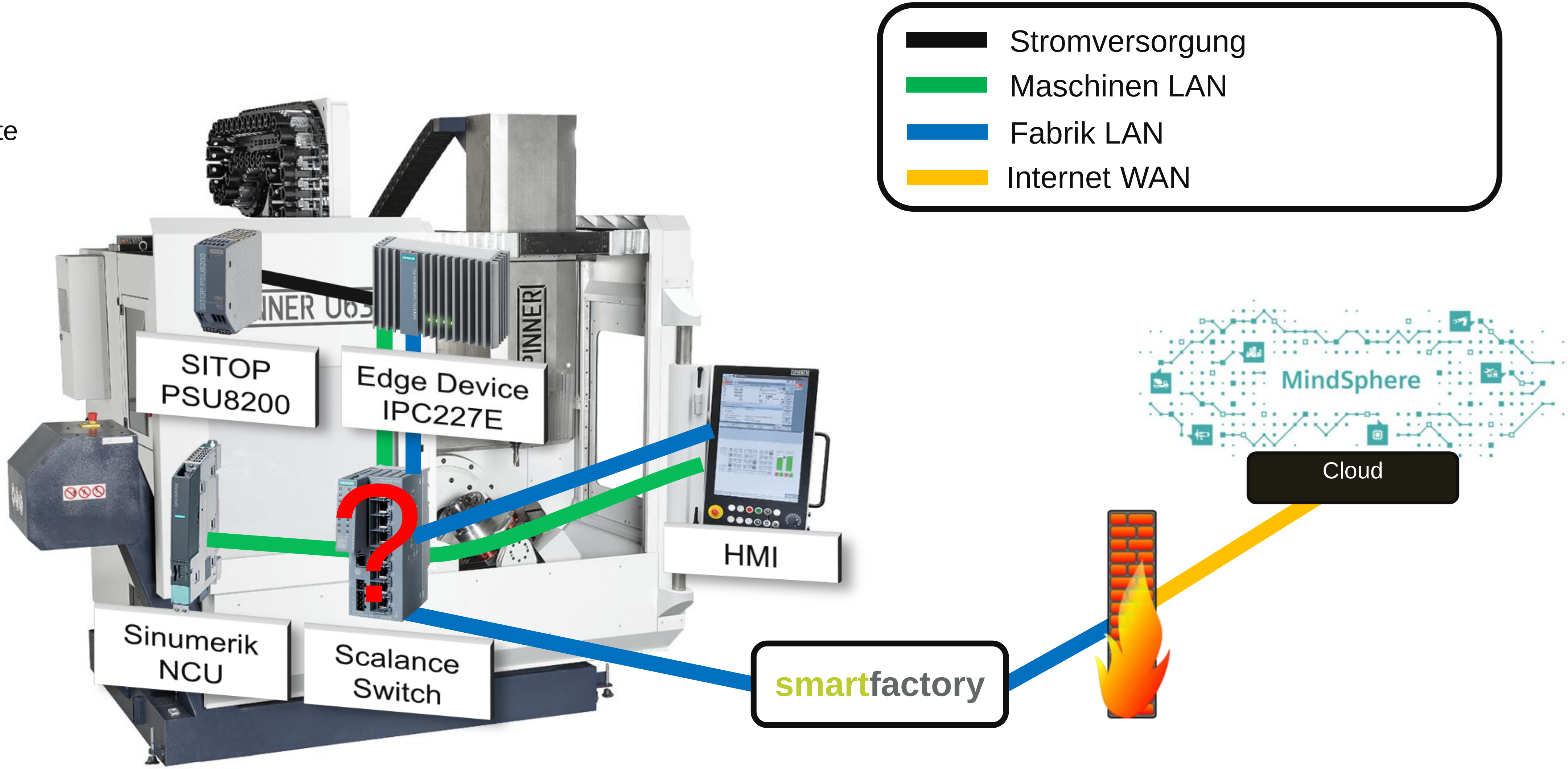
Quelle: [3]

Edge Computing



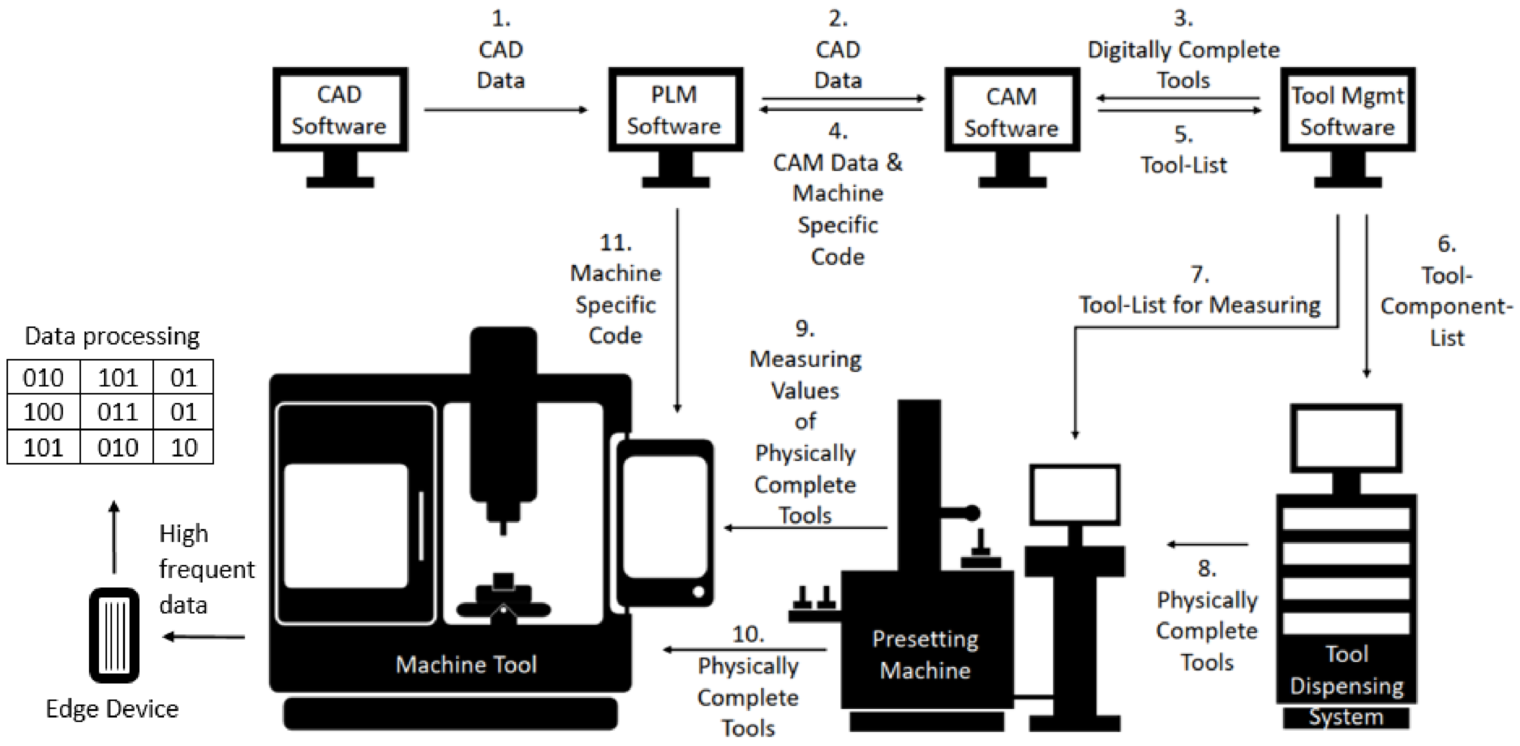
Edge Computing

- Netzwerkelemente



Edge Computing

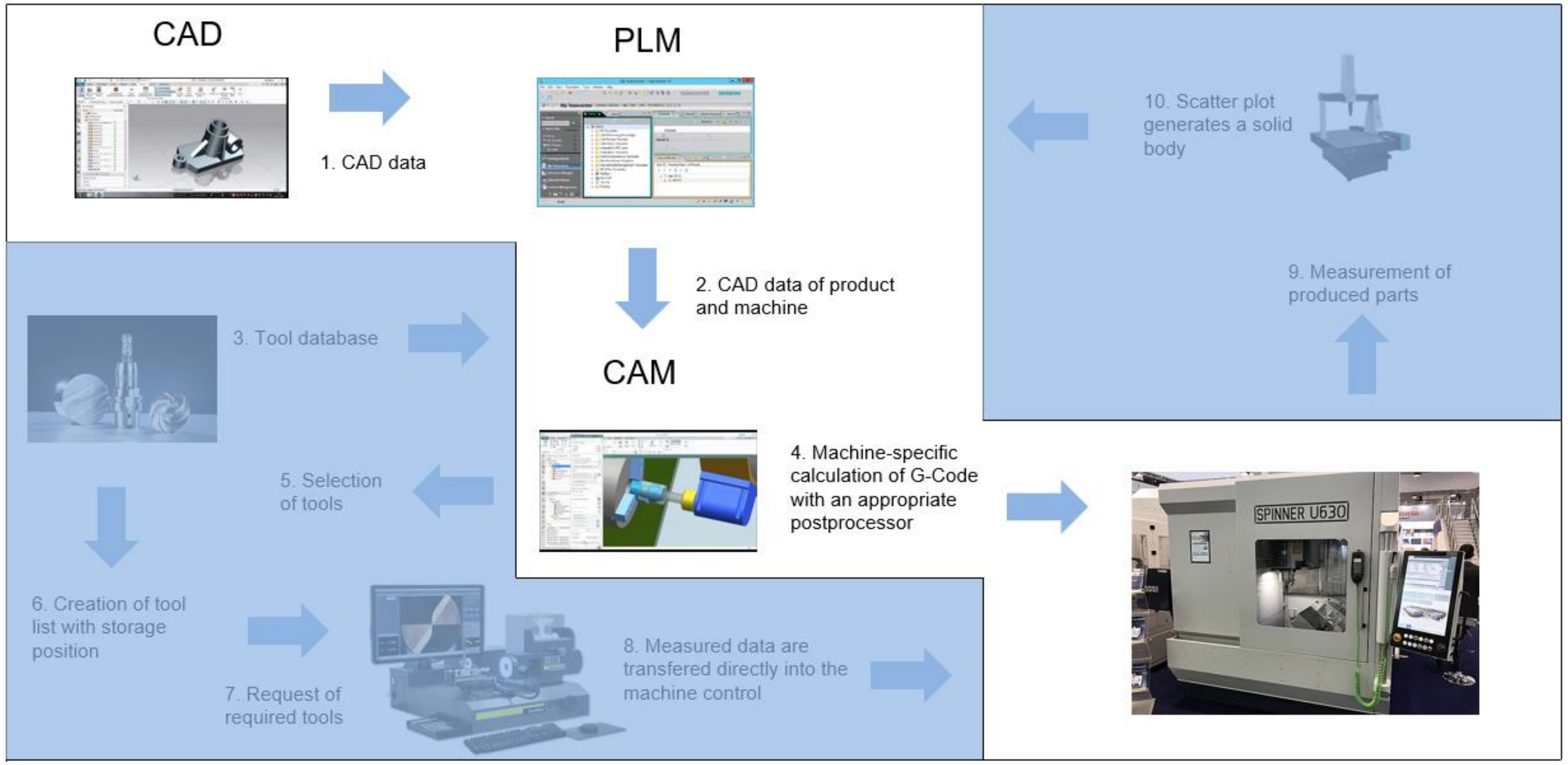
- Smarte Infrastruktur zur spanabhebenden Bearbeitung



Quelle: [6]

Edge Computing

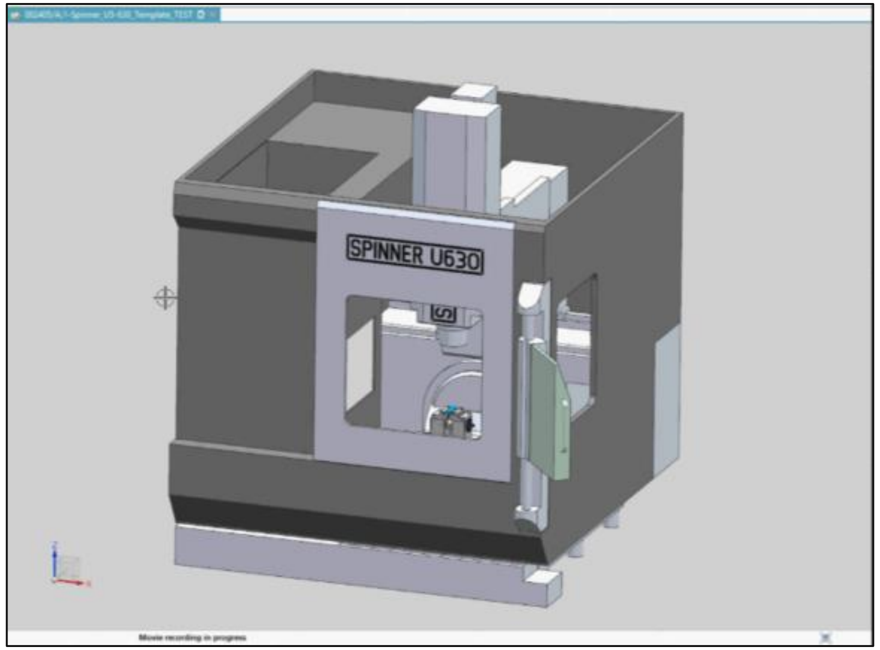
- Datenfluss vom Werkstück bis zur Maschine



Edge Computing

- Werkzeugmaschine: real und digital

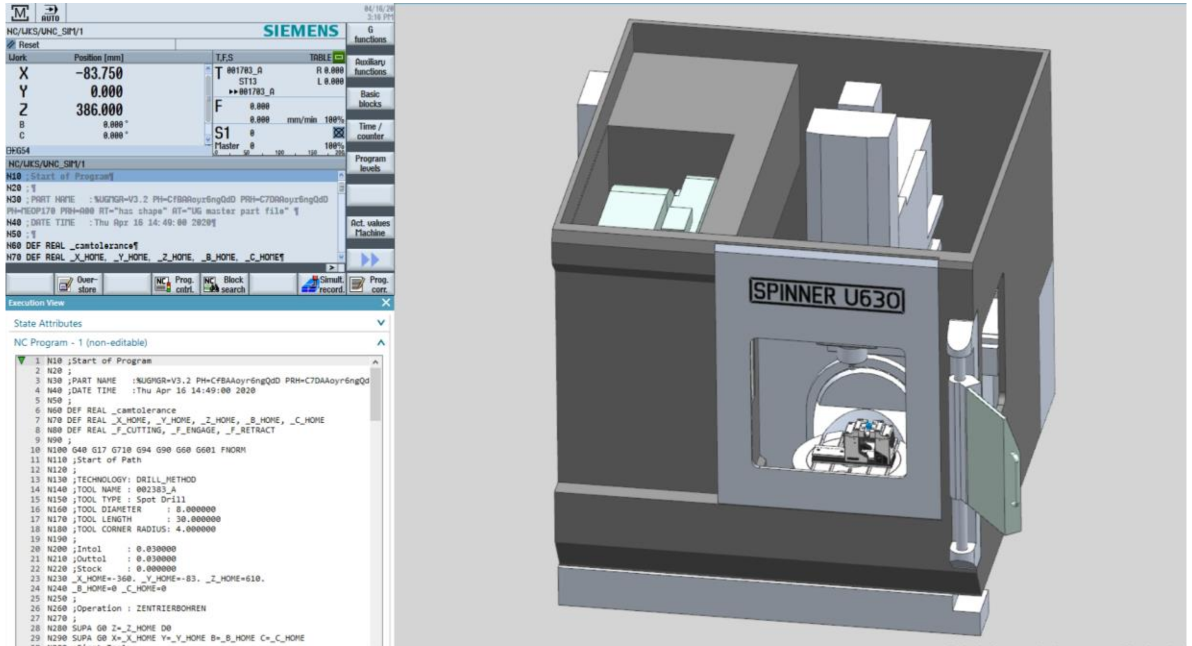
Reale Werkzeugmaschine



Digitales Maschinenmodell

Edge Computing

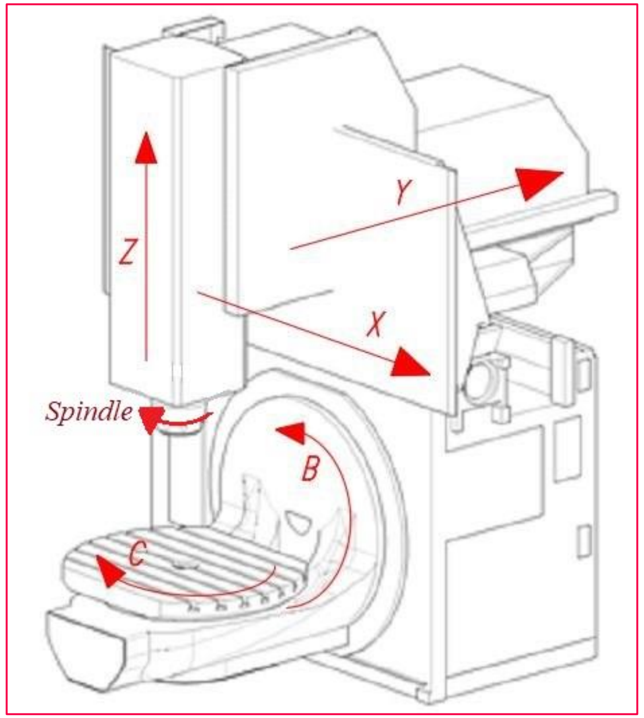
- CAM Simulation



Dipl.-Ing. Johannes Schmid

Edge Computing

- Maschinenkonfiguration
  - 5-Achs Fräsmaschine: Spinner U5-630
  - Maschinenachsen: X, Y, Z, B, C



Edge Computing

- Bohrer-Prozess: realistische Bohrerabnutzung mit gewollten Brüchen
- Erforderliche Parameter: Istwerte während dem Bohrerbruch
- Erfassen von Daten der Achsantriebe der NC Maschine
  - Niederfrequenzdaten (LFD) mittels OPC UA (Zeitintervall: 100 ms)
  - Hochfrequenzdaten (HFD) mittels SINUMERIK Edge (Zeitintervall: 2 ms)

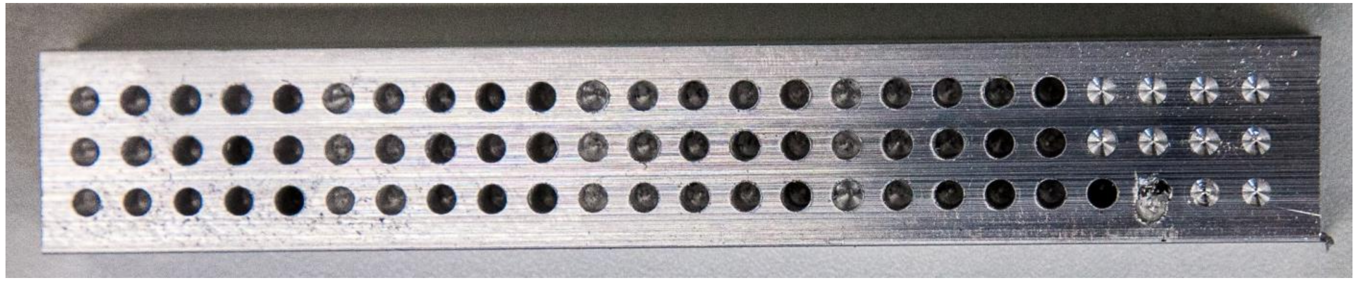
Edge Computing

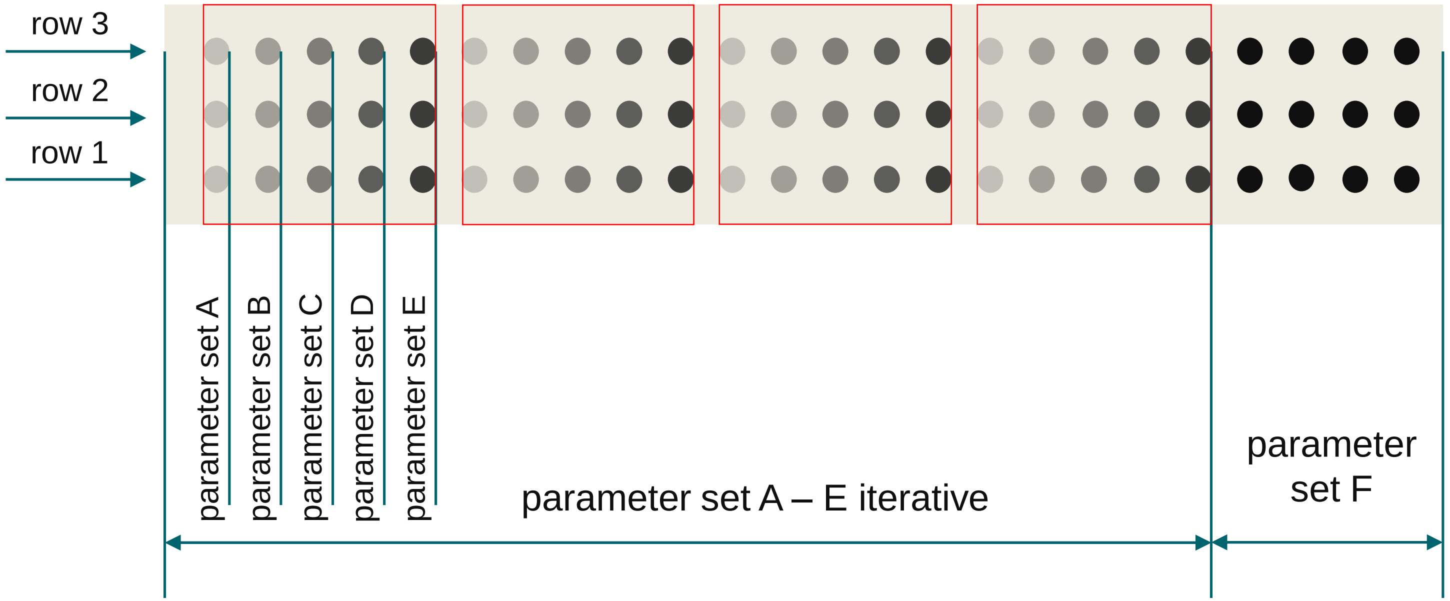
- Werkstück
  - Material: AlCuMgPb
  - Abmessungen: 20 mm x 15 mm x 125 mm

- Werkzeug
  - Name: Spiralbohrer
  - Material: HSS Titanium
  - Beschichtung: Nitrid
  - Abmessungen: 2.8 mm x 61 mm

Edge Computing

- Werkstück Parameter Sets



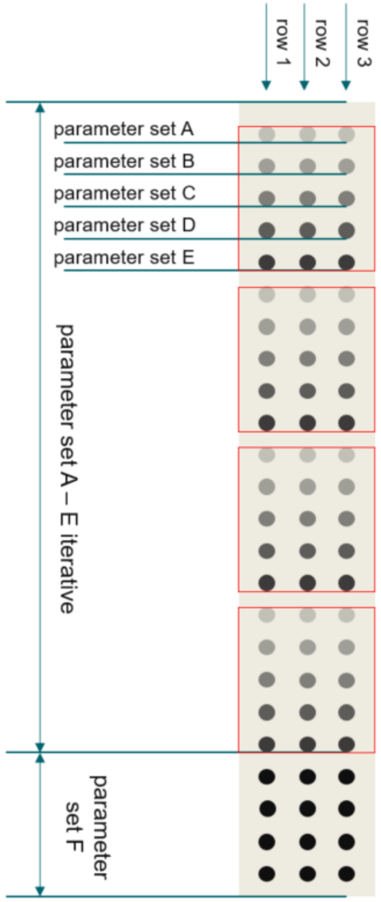


Edge Computing

- Werkzeugmaschine Parameter Sets

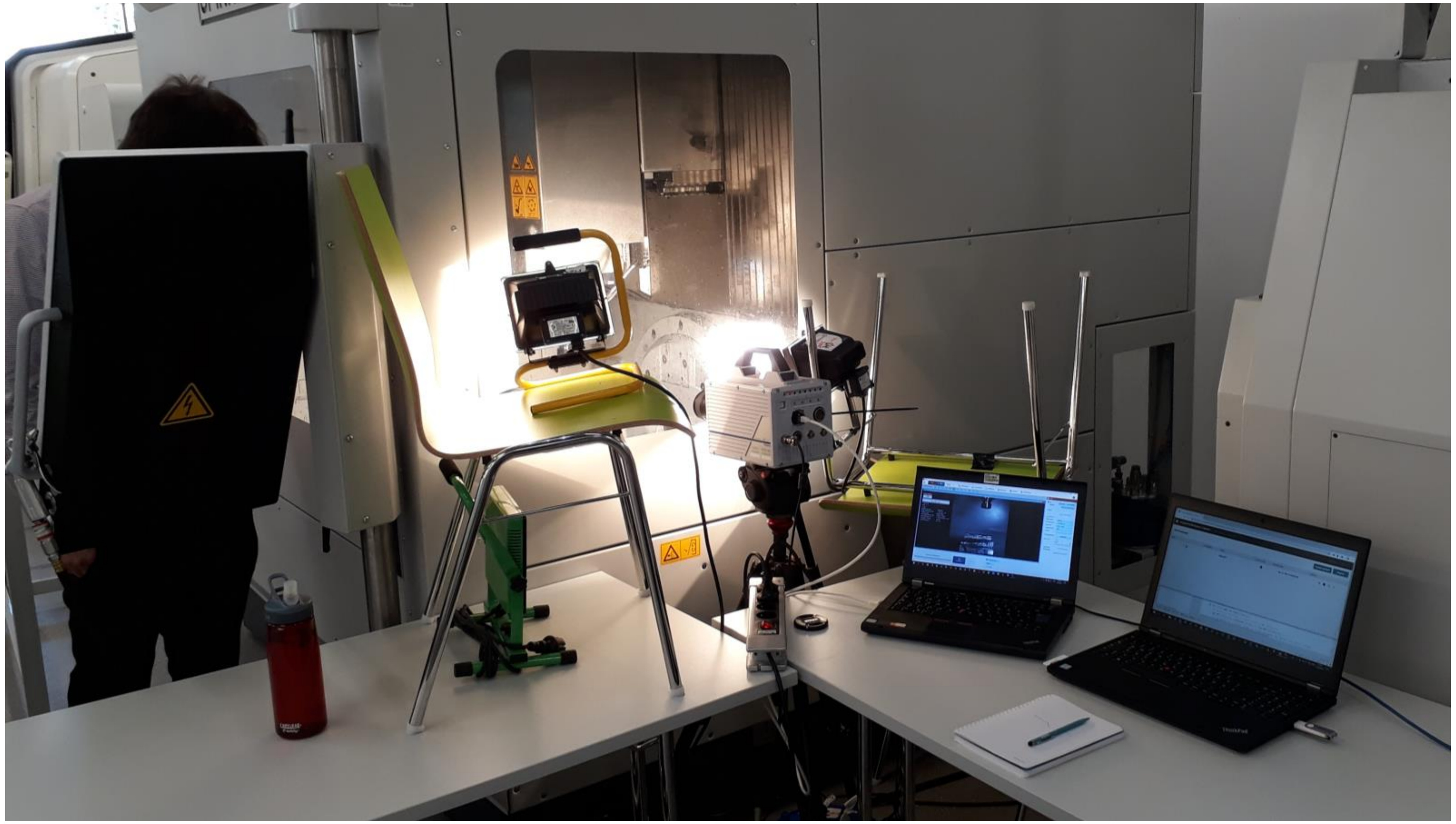
Parameter Set	Tabular Value	Rotational Speed	Drilling Depth
A	Lower Values	5456 min^{-1}	5,46 mm
B	Upper Values	7389 min^{-1}	7,39 mm
C	Upper Values + 20%	8867 min^{-1}	8,87 mm
D	Upper Values + 30%	9606 min^{-1}	9,61 mm
E	Upper Values + 40%	10345 min^{-1}	10,35 mm
F	Upper Values + 80%	13300 min^{-1}	13,30 mm

Jedes Parameterset mit konstanter Vorschubrate von 0.12 mm / Umdrehung

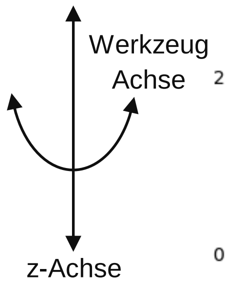


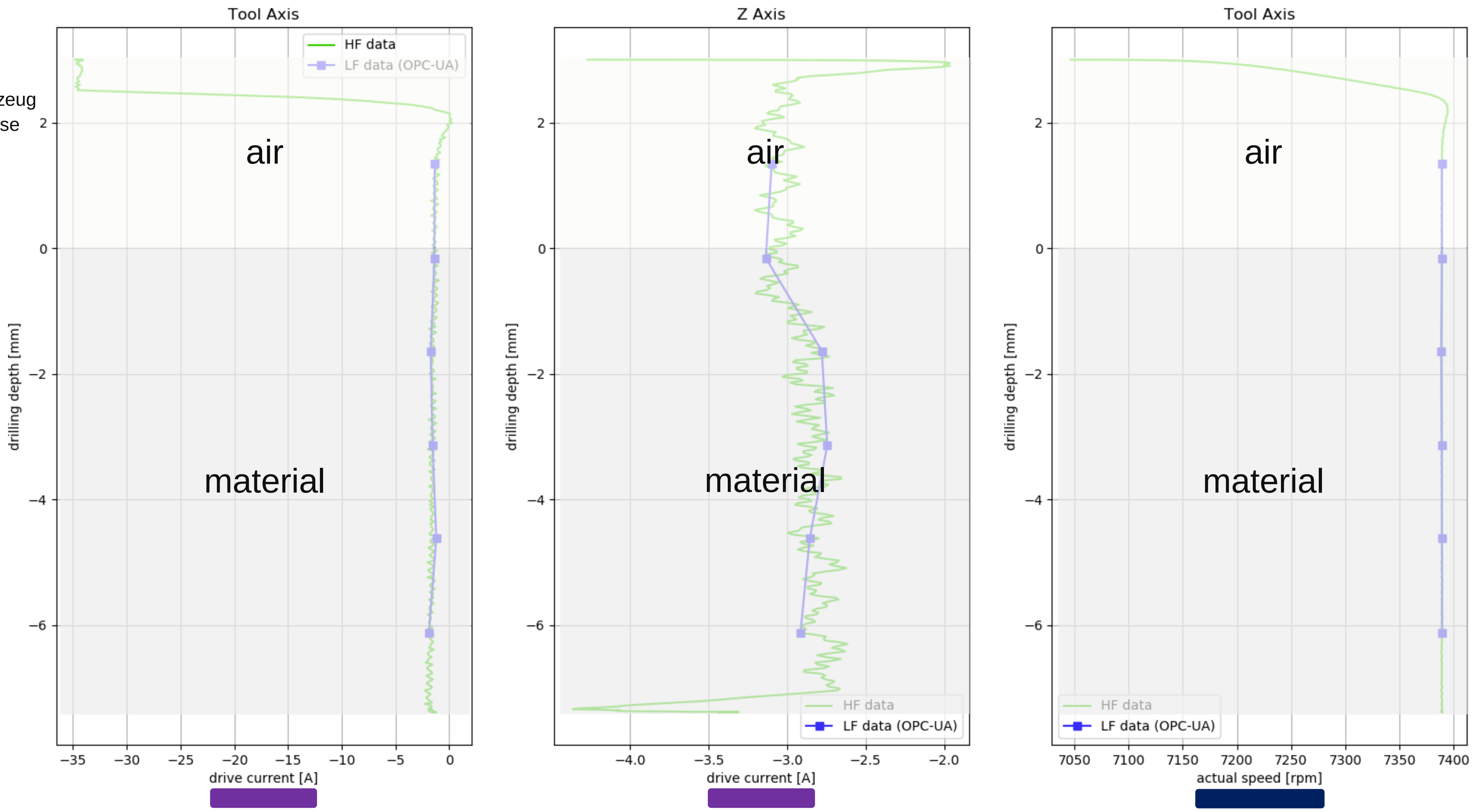
Edge Computing

- Vorbereitung für Aufnahme mit Hochgeschwindigkeitskamera

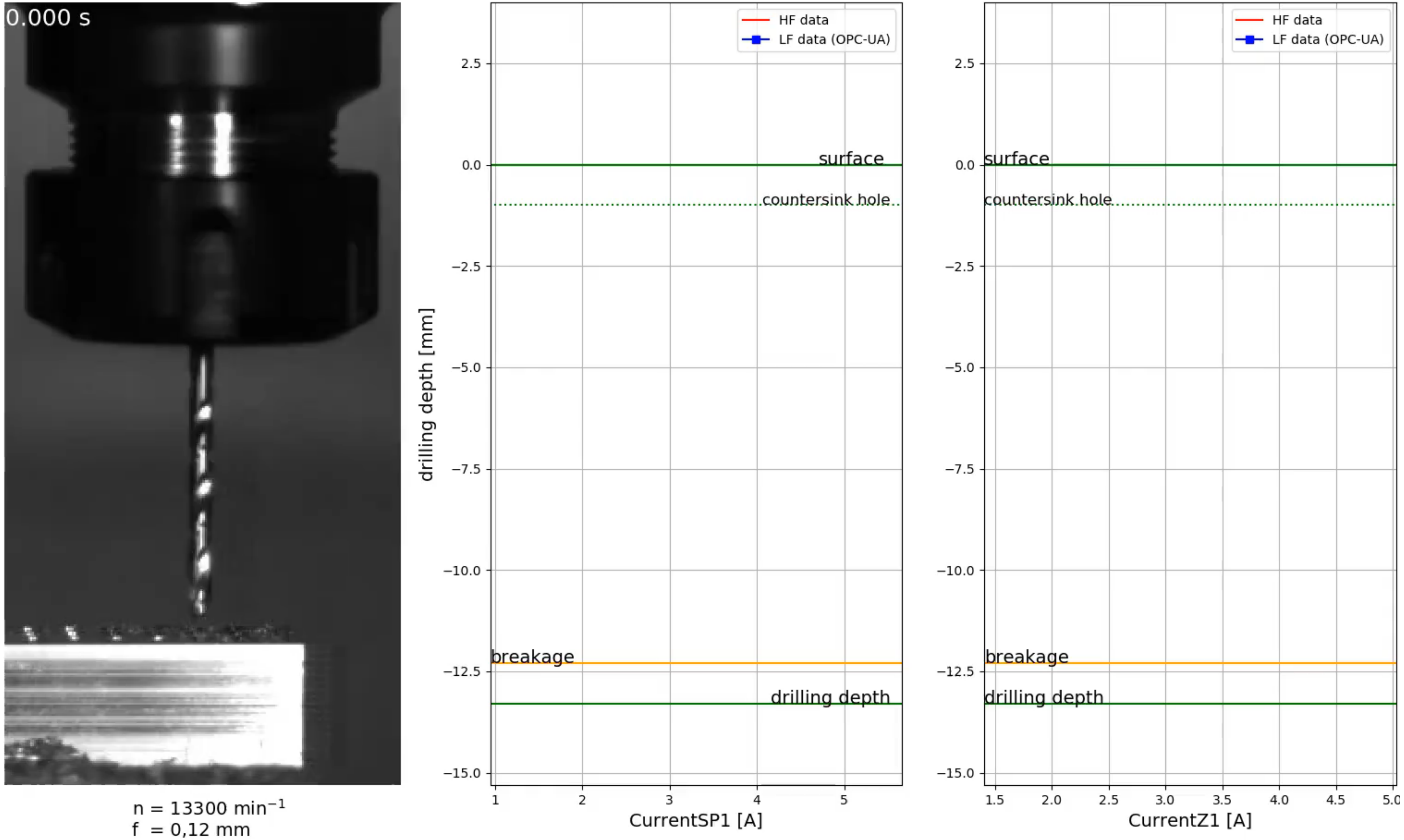


Durchführung Experimente





Durchführung Experimente



n = 13300 min^{-1}
f = 0,12 mm

Edge Computing

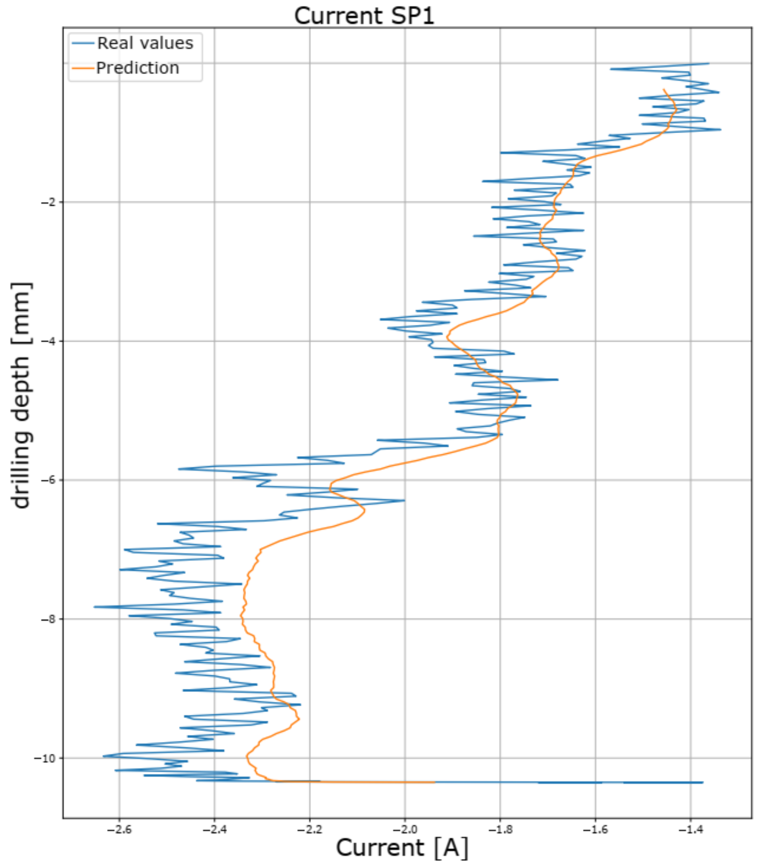
- Nach den Bohrversuchen

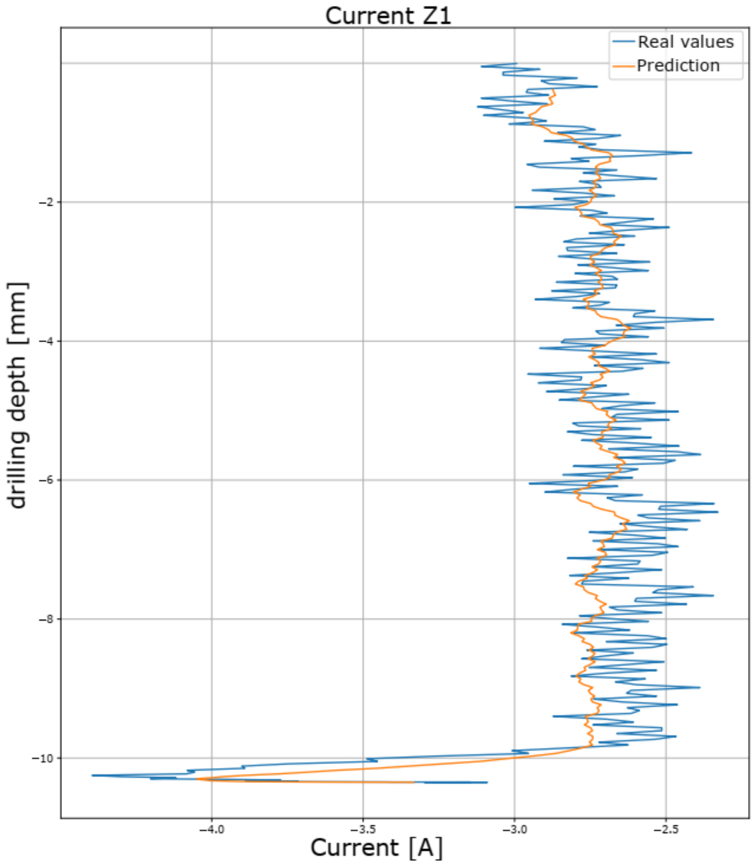
Edge Computing

- LSTM (Long Short Term Memory): Basis
  - Vorhersage des Antriebsstroms
  - Lernen durch Trainingsdaten
  - Zukunftswerte werden auf Basis von Vergangenheitswerten vorhergesagt

Edge Computing

- LSTM (Long Short Term Memory): Ergebnisse



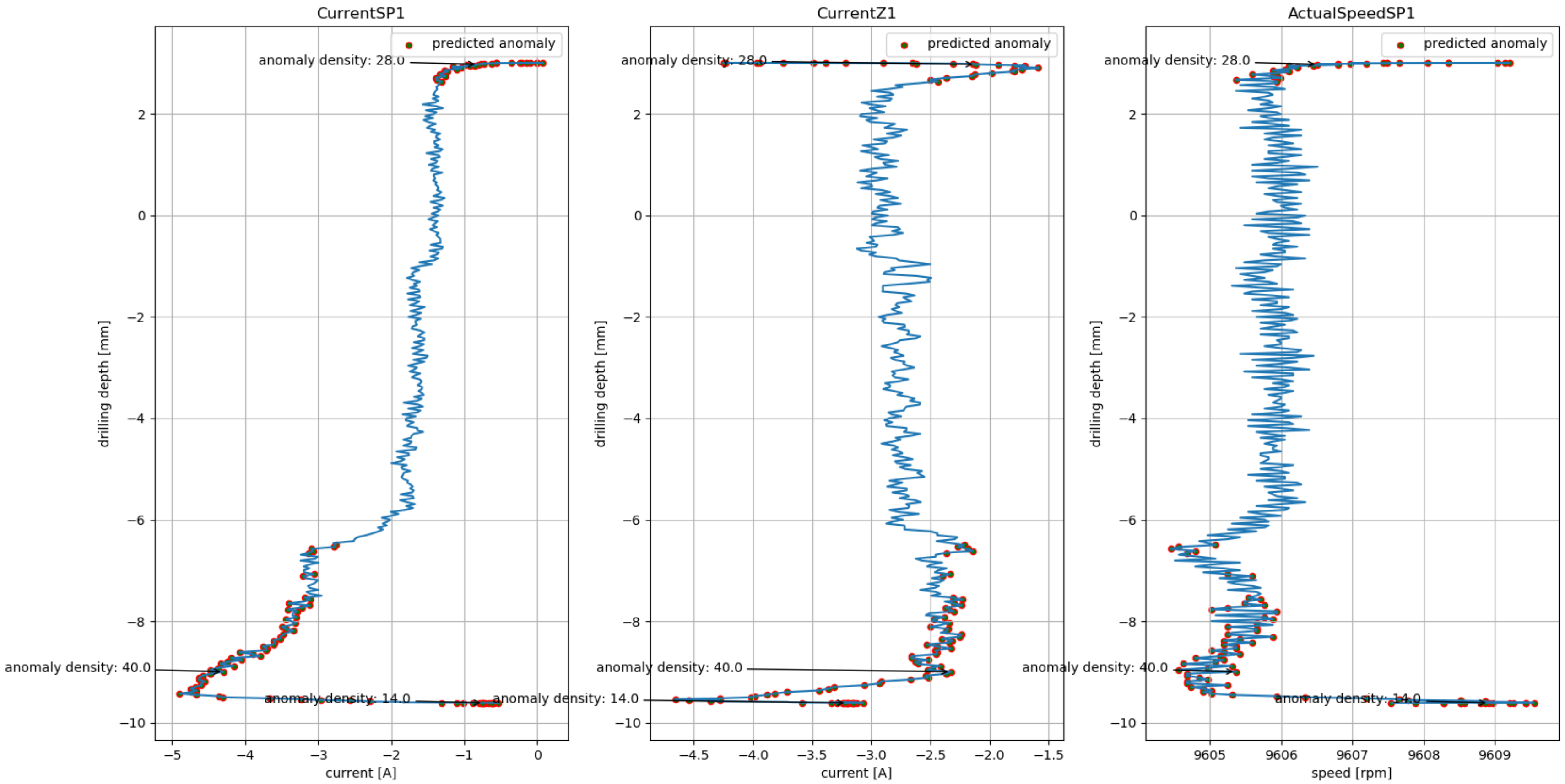


Edge Computing

- Isolation Forest
  - Klassifiziert Messdaten
  - Lernen durch Trainingsdaten
  - Einstellung der erwarteten Anomalien anhand dem Parameter “Contamination”

Edge Computing

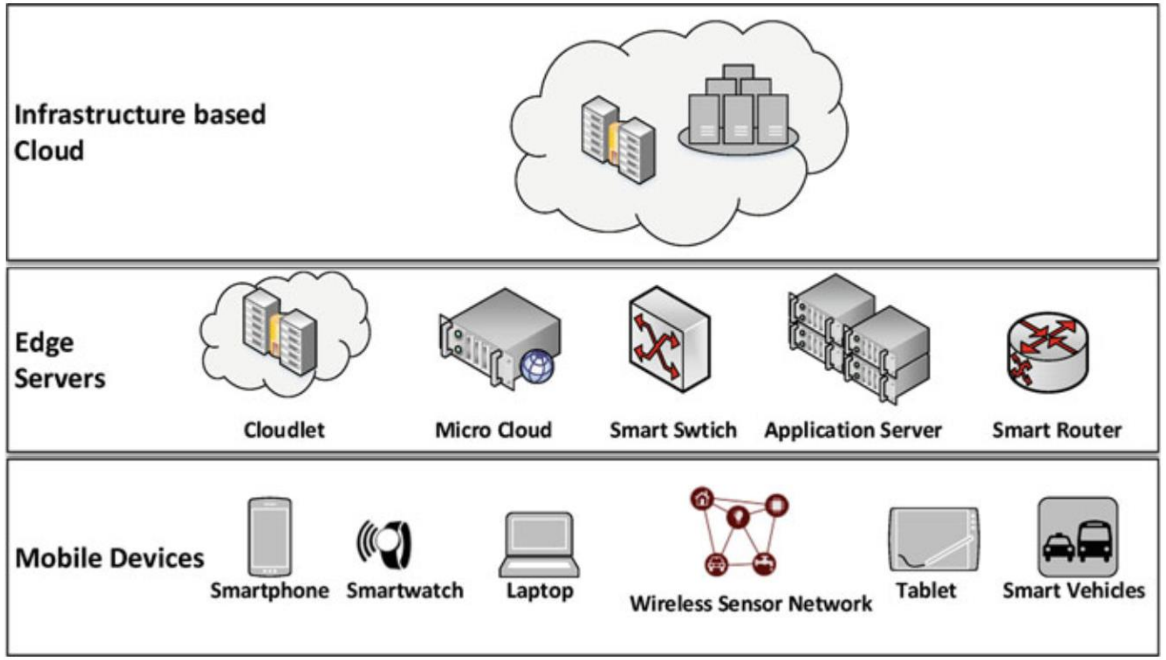
- Isolation Forest: Ergebnisse



Edge Computing

- Edge App zur Anomalie Detektion
  - Identifiziert Anomalien anhand von Echtzeitdaten mittels Isolation Forest
  - Möglichkeit des Senden von besonders interessanten Daten in die Cloud

Edge Computing



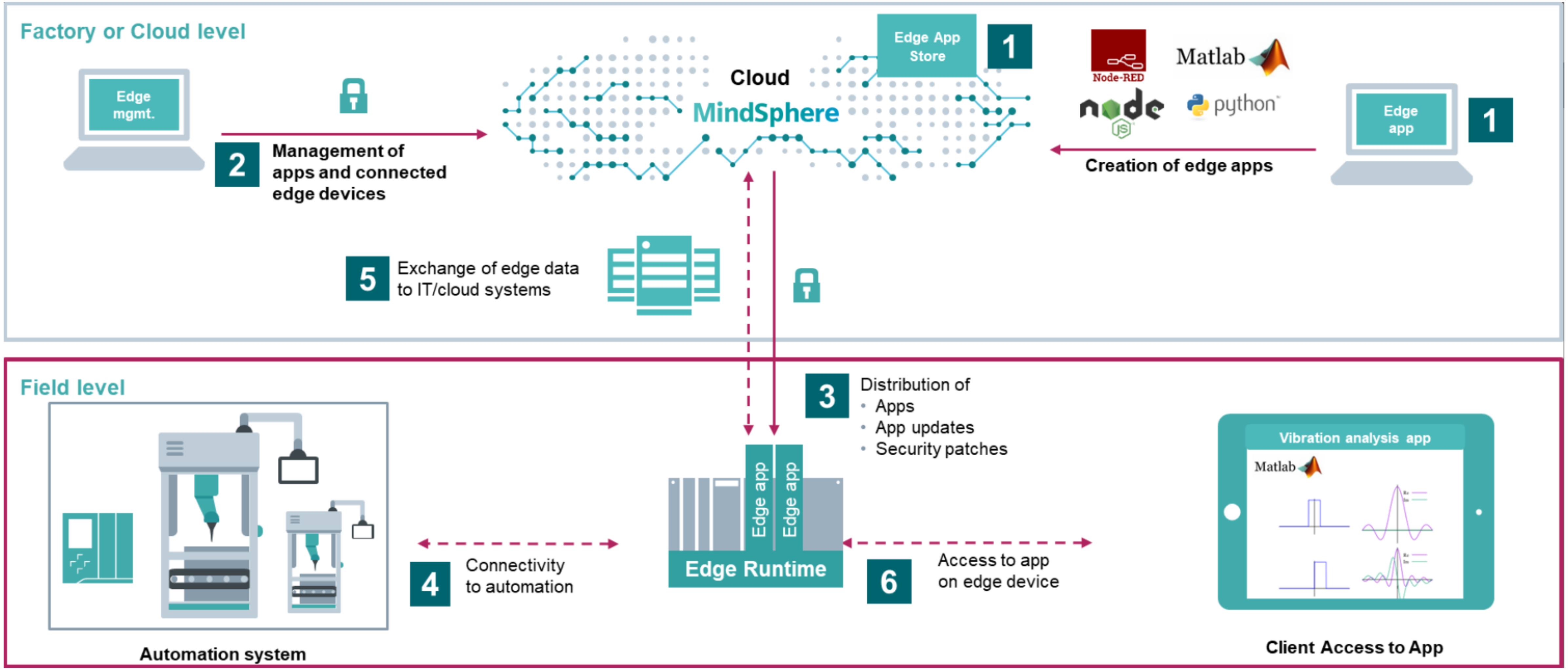
Cloud


Edge Device

Fräszentrum

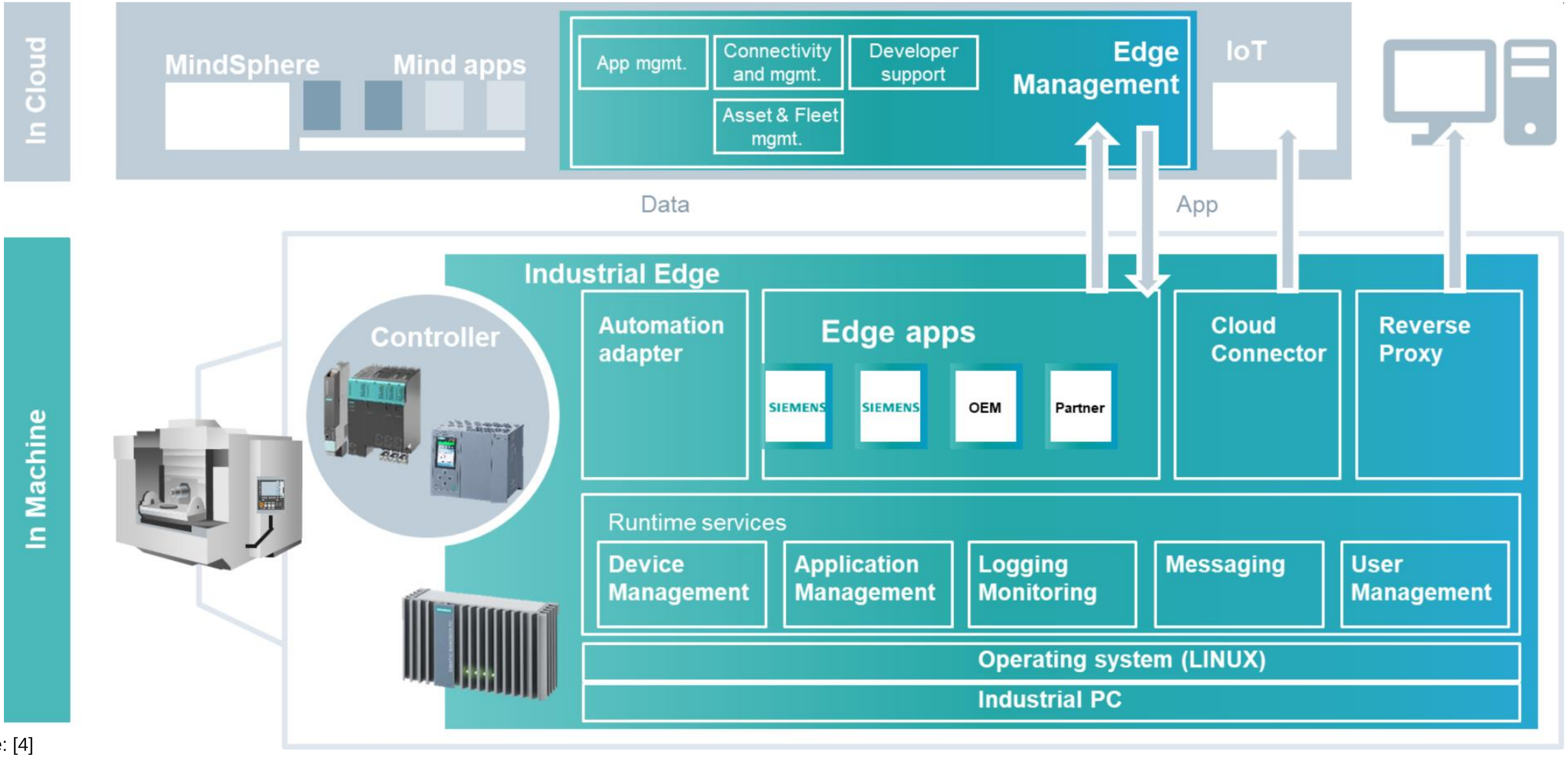
Quelle: [3]

Edge Computing



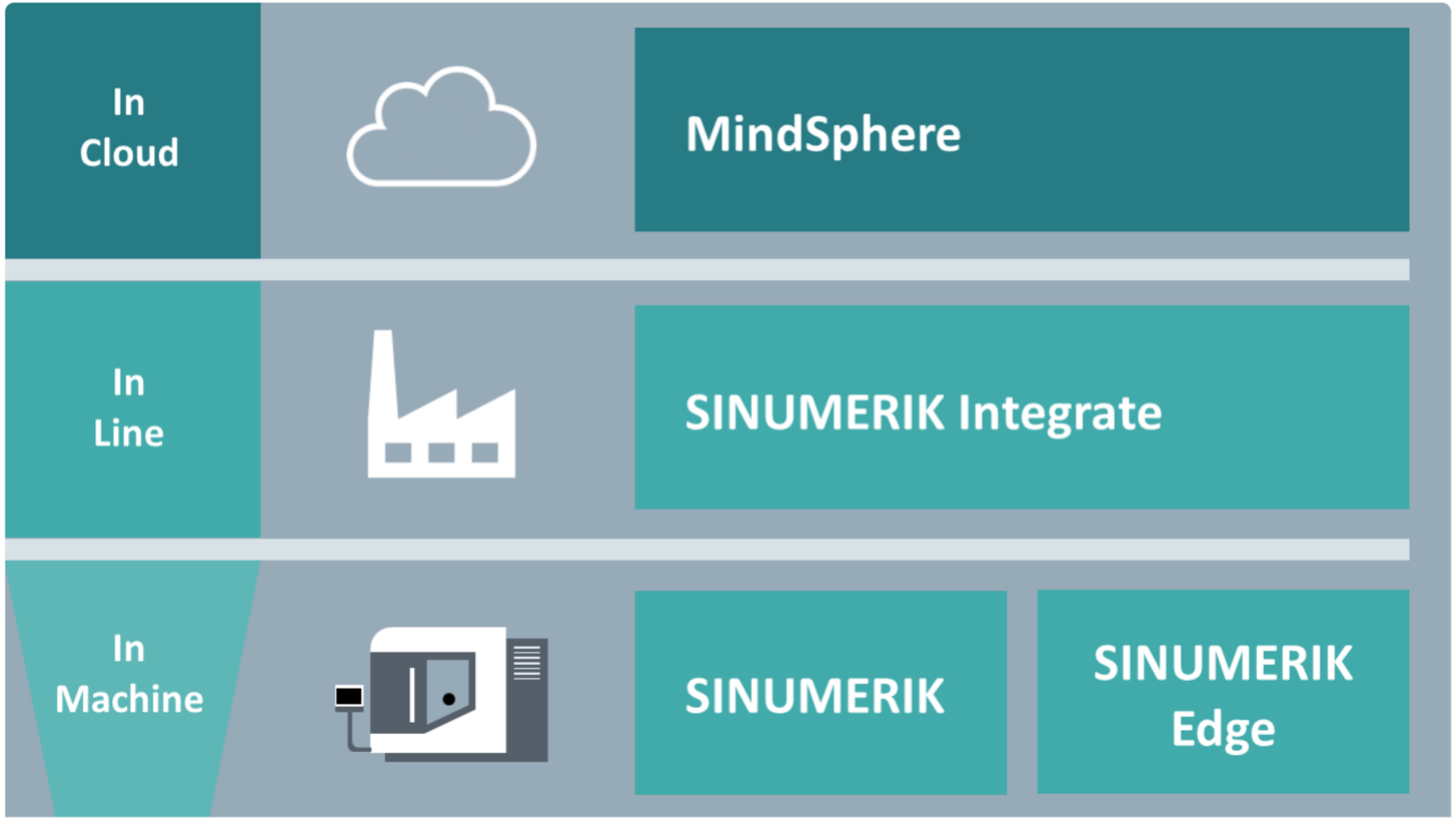
Quelle: [4]

Edge Computing



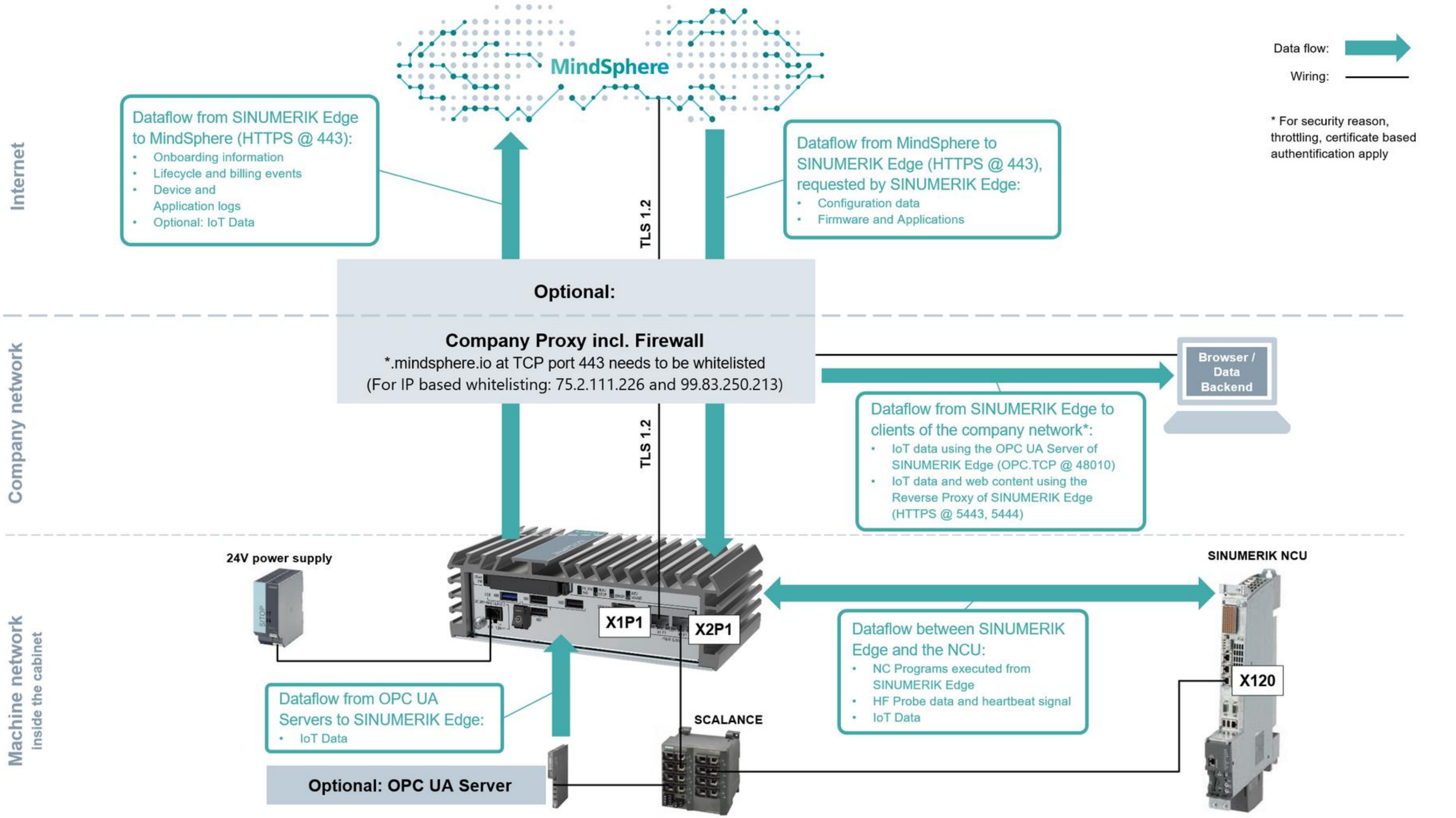
Quelle: [4]

Edge Computing



Quelle: [4]

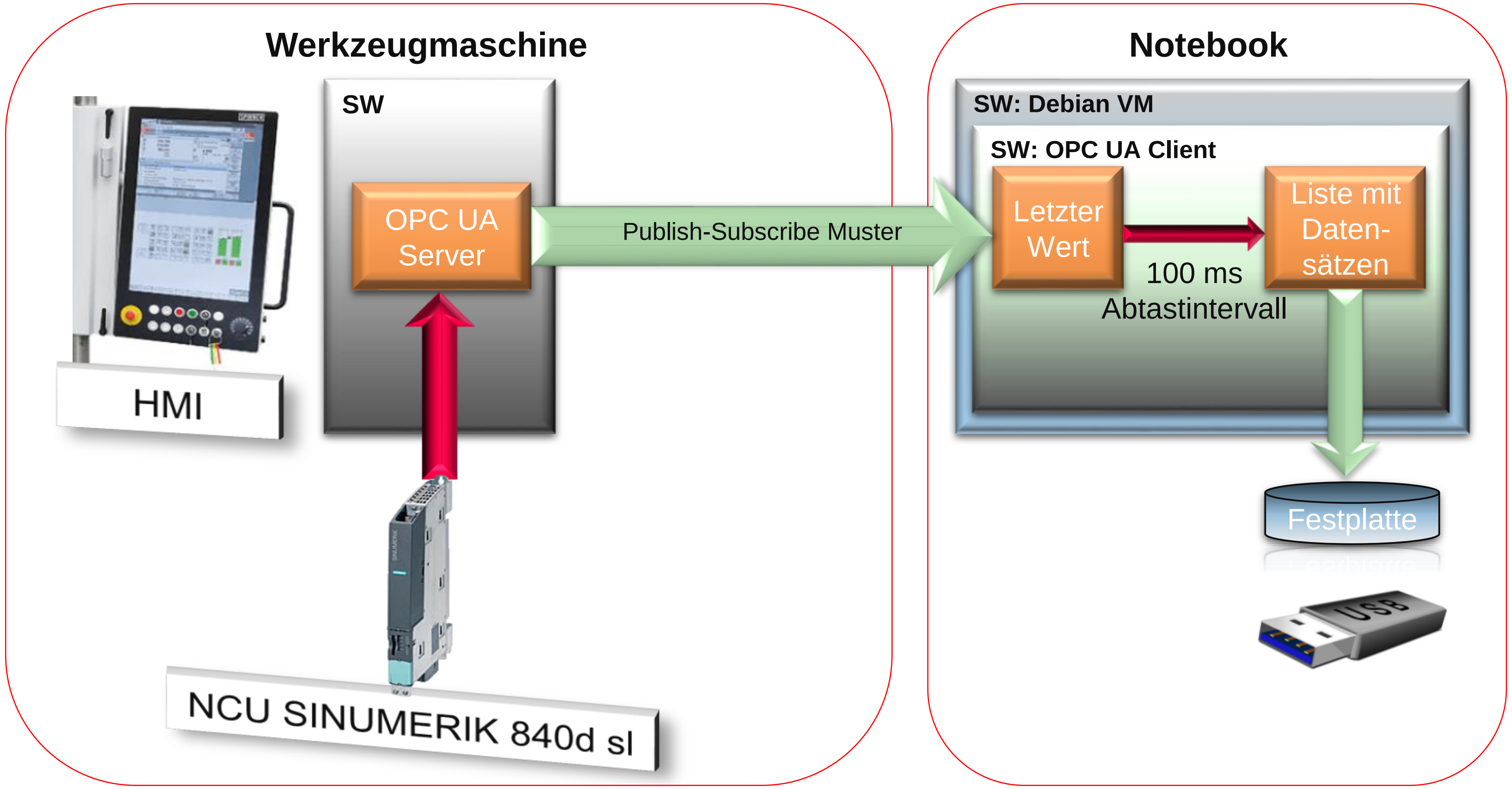
Edge Computing



Quelle: [4]

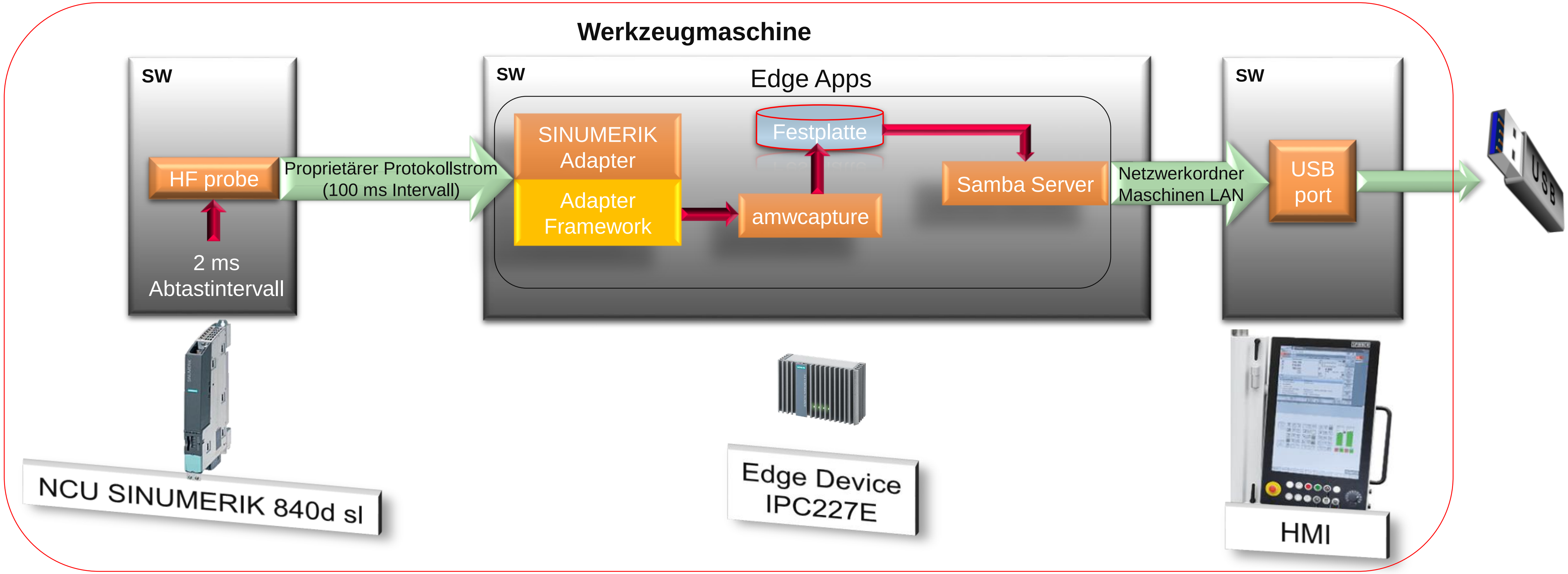
Edge Computing

- Architektur zur Datenerhebung (LF)



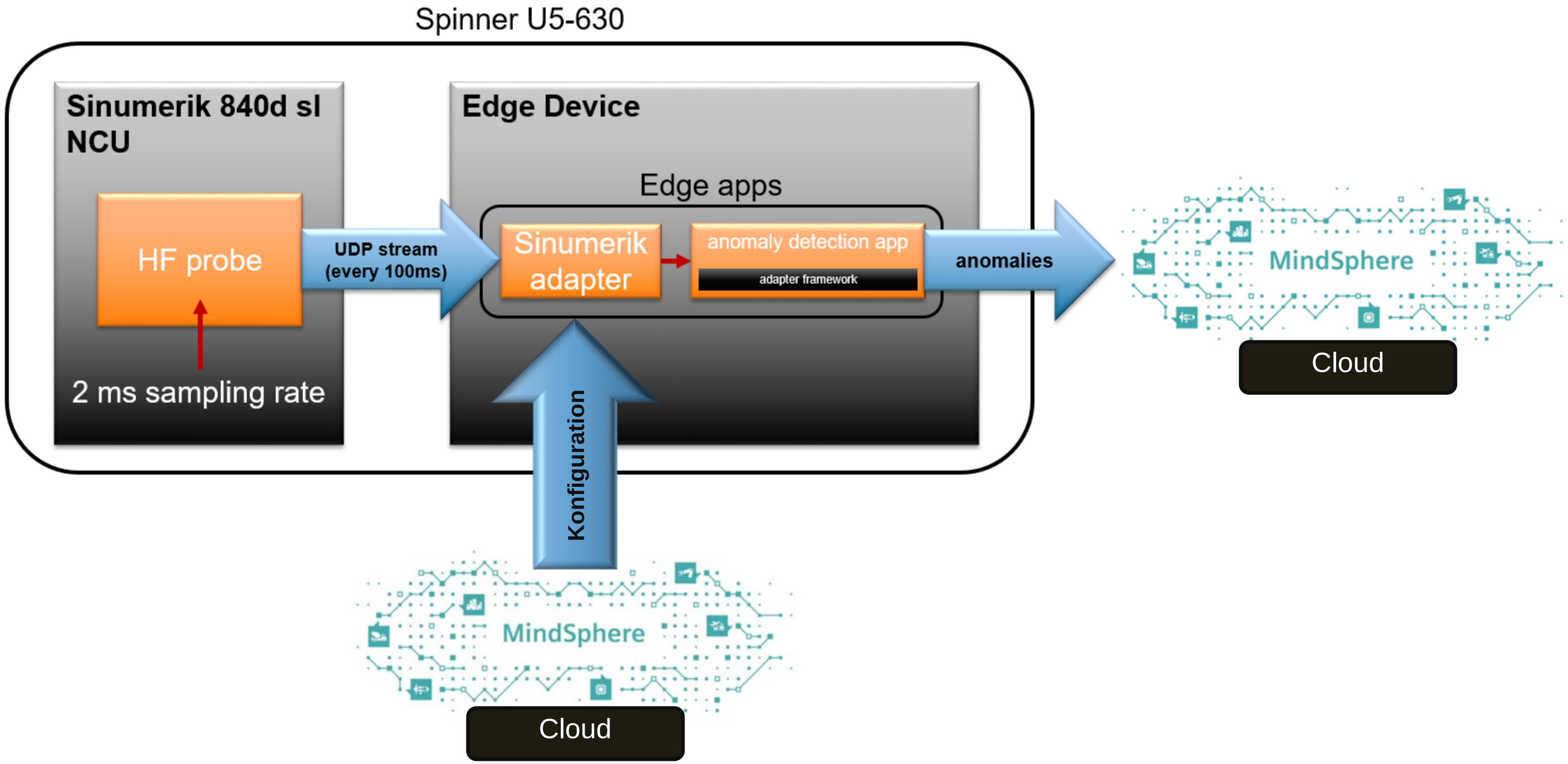
Edge Computing

- Architektur zur Datenerhebung (HF)



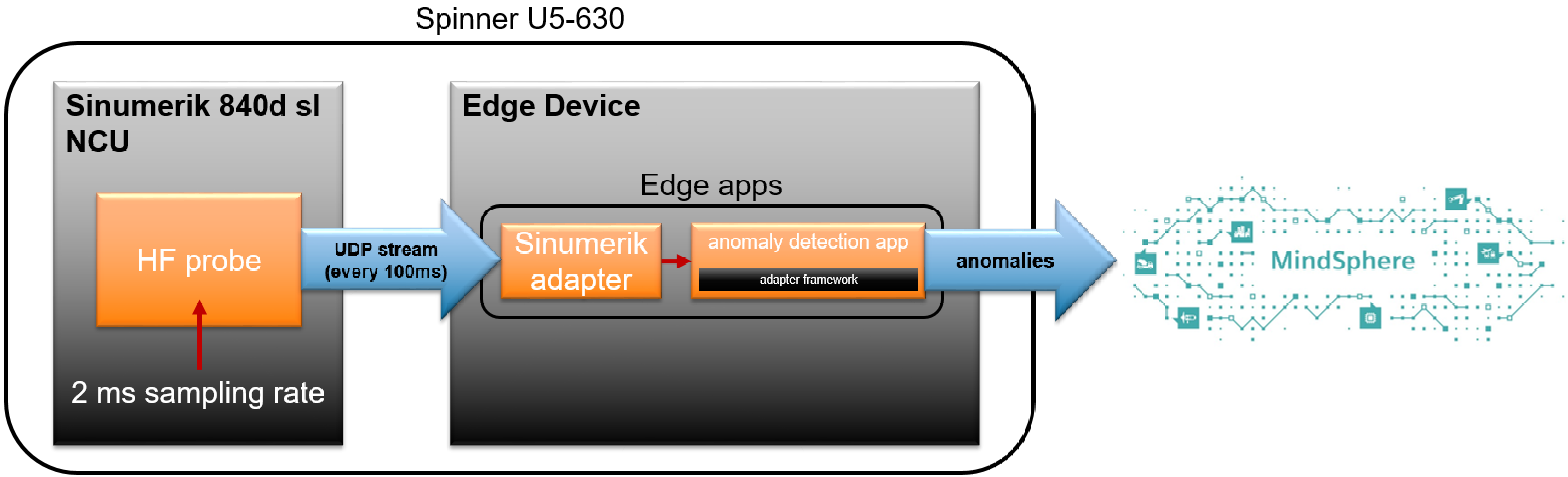
Edge Computing

- Konfiguration Edge Apps



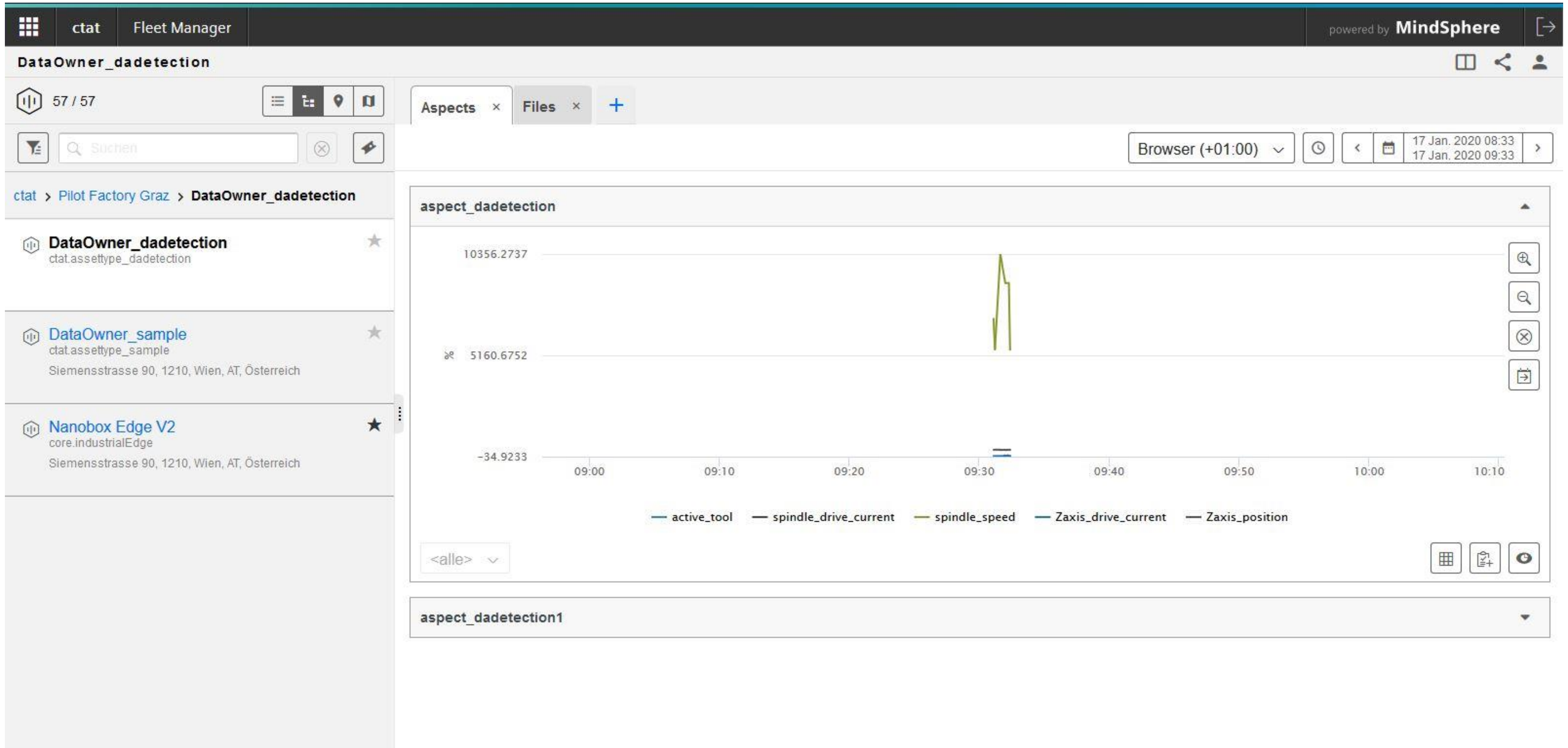
Edge Computing

- Edge App: Anomalie Transport in die Cloud



Edge Computing

- Edge App: Anomalie Transport in die Cloud



Edge Computing

- 5-Achs Bearbeitungszentrum Spinner U5 630
  - Fräsmaschine und Edge Device in Betrieb
  - Öffnen Edge App Analyze My Workpiece /Capture (AMW /Capture)
    - Neueste Version AMW /Capture4Analysis (freie Verwendung der Daten)
  - Konfiguration HF Signale
    - Triggerarten: manuell und automatisiert
    - Anlassloses und automatisiertes Speichern von Bearbeitungsdaten möglich (z.B.: durch Triggerung mittels NC-Code)
  - Abschluss der NC-Bearbeitung
    - Speicherung der HF-Daten am Edge Device
  - Datenübertragung auf USB-Stick
    - Daten liegen im .json Format vor

Edge Computing

Generierung – **Werkzeugmaschine Spinner U5-630**

Aufnahme – **SINUMERIK Edge Device IPC 227E**
Datendatei; Job Steuerung über das Fabriks-LAN

Übertragung – Hochfrequente Datenübertragung über das Fabriks-LAN

DATENAKQUISITION – Phase 1

Untersuchung – **Selektion**: Auswahl auszuwertender Kanäle
Konvertierung: Umwandlung von Rohdaten in strukturiertes Format (z.B.: Zeilen, Spalten)

Bereinigung – **Filterung:** Extraktion der wichtigsten Informationen
Datenspeicherung: Entwicklung von Datenfeldern

Kennzeichnung – **Analyse:** Kennzeichnung relevanter Informationen in Datenfeldern sowie Bestimmung deren gegenseitiger Abhängigkeiten

DATENANALYSE VORBEREITUNG – Phase 2

Datengewinnung – **Explorative Datenanalyse (EDA)**: mittels interaktiver Visualisierungtechniken, z.B.: Multi-Axen Plot, Korrelationsdiagramme, etc.

DATENANALYSE DURCHFÜHRUNG – Phase 3

Machine Learning – **Klassifikation, Regression, Gruppierung -> Vorhersage**
Ableitung erweiterter Prozess-Strategien

KPI VORHERSAGE – Phase 4

Ausführung – **Implementierung** erweiterter Prozess-Strategien

PROZESS VERBESSERUNGEN – Phase 5

Quelle: [5]

Edge Computing

- Phase 1: DATENAKQUISITION
  - Eingabe: NC Code Datei (.mpf Format)

```
N32 SUPA G0 Z=_Z_HOME D0
N34 SUPA X=_X_HOME Y=_Y_HOME
N36 T="T_A_SK40_SF_DM16_A"  ;  T_A_SK40_SF_DM16
N38 M6
N40 T="002383_A"
N42 _F_ENG = 500 ;Engage Move
N44 _F_CUT = 500 ;Cutting
N46 _F_RET = 500 ;Retract Move
N48 G54
N50 CYCLE800(1,"TABLE",0,57,0,0,0,0,0,0,0,0,0,1,0)
N52 G0 X-21. Y0. S10000 D1 M3
N54 Z5.
N56 M168
N58 Z0.
N60 G94 G1 G90 Z-3. F=_F_ENG
N62 X-10.
N64 Y62.5 F=_F_CUT
N66 X10.
N68 Y-62.5
N70 X-10.
N72 Y3.338
N74 G2 X-9.532 Y4.468 I1.6 J-0.001
N76 G1 X-2.468 Y11.532
N78 G3 X-2. Y12.662 I-1.132 J1.131
N80 G1 Y52.904
N82 G2 X-0.404 Y54.5 I1.6 J-0.004
N84 G1 X0.404
```

Ausgabe: hochfrequente Daten (.json Format)

{"Version":{"outputFileFormatVersion":"0.6","RecorderVersion":null},"Format":"DEFAU
e","Type":"DOUBLE","Axis":"Z1","Address":"TORQUE|3"},{"Name":"Torque","Type":"DOUBL
sition","Type":"DOUBLE","Axis":"B1","Address":"ENC2_POS|4"},{"Name":"Encoder2Positi
BLE","Axis":"Z1","Address":"ENC1_POS|3"},{"Name":"Encoder1Position","Type":"DOUBLE"
","Type":"STRING","Axis":"Y1","Address":"CMD_SPEED|2"},{"Name":"CommandedSpeed","Ty
"Axis":"X1","Address":"CURRENT|1"},{"Name":"Current","Type":"STRING","Axis":"Y1","A
xt":"/data/dataoutput/JOB2020-07-13T07_07_44.967.1.json","Number":0},"TimeStamp":"2
,"Payload":
[{"HFData":[[27777,-0.001183,-0.00441,609.997863,1.0E-5,0.4882,3.5E-4,157.49975,0.0
.405121,3.5E-4,157.499751,5.0E-6,-7.0E-6,2.96E-4,4.29E-4,0.0,9.0E-6,-1.8E-5,0.0,-0.
,609.973943,0.0,0.0,3.5E-4,157.499751,0.0,-0.0,4.0E-6,1.3E-5,0.0,0.0,-1.0E-6,-0.001
63,1.0E-5,0.4882,3.5E-4,157.49975,0.004105,0.014301,-7.8072,-0.568666,-0.032056,-0.
,0.0,9.0E-6,-1.8E-5,0.0,-0.0,4.0E-6,1.3E-5,0.083079,0.0,-1.0E-6,0.00174,0.011749,-3
E-5,0.0,0.0,-1.0E-6,-0.001183,-0.004395,609.997859,-3.0E-6,0.405121,3.5E-4,157.4997
1,-7.8072,-0.558037,0.0,-0.009501,0.060593,0.006104,0.024414,10.089111,0.787354,0.0
74,0.006714,-3.631256,-0.284271,-0.003162,-0.005249,0.044067,0.0,0.0,0.0,0.0,0.0,0.
7.499751,-0.001183,-0.00441,609.997859,0.0,0.405121,3.5E-4,157.499751,5.0E-6,-7.0E-
07,0.604248,-0.009064,-0.004425,609.973943,0.0,0.0,3.5E-4,157.499751,0.0,-1.5E-5,4.
,0.0,0.0],[27794,-0.001183,-0.00441,609.997863,1.0E-5,0.4882,3.5E-4,157.49975,0.004
4,157.499751,5.0E-6,-7.0E-6,2.96E-4,4.29E-4,0.0,9.0E-6,-1.8E-5,0.0,-0.0,4.0E-6,1.3E
.5E-4,157.499751,0.0,-0.0,4.0E-6,1.3E-5,0.0,0.0,-1.0E-6,-0.001183,-0.00441,609.9978
5,0.4882,3.5E-4,157.49975,0.004105,0.017876,-7.8072,-0.563351,-0.006411,-0.009501,0
,-1.8E-5,0.0,-0.0,4.0E-6,1.3E-5,0.083079,0.0,-1.0E-6,0.00174,0.008392,-3.631256,-0.
.0E-5,0.0,0.0,-1.0E-6,-0.001183,-0.004395,609.997859,0.0,0.405121,3.5E-4,157.499751
,-7.8072,-0.616498,-0.006411,-0.009501,0.060593,0.006104,0.030518,10.089111,0.79345

Edge Computing

- Phase 1: DATENAKQUISITION
  - Ausgabe: hochfrequente Daten (https://jsonformatter.org/)
  - Beispiel HF-Daten (Minimalauszug)

Edge Computing

- Phase 2: DATENANALYSE VORBEREITUNG
  - Nach Datenextraktion sequenzieller Prozess für Untersuchung, Bereinigung und Kennzeichnung von Daten
  - Programmierung z.B.: mittels Python

Achsen
X-Achse
Y-Achse
Z-Achse
B-Achse
C-Achse
Spindel-Achse translatorisch
TC Achse rotatorisch

Kanäle pro Achse (Auswahl)
SOLL Geschwindigkeit
IST Geschwindigkeit
Drehmoment
Leistung
Last
Verschiedene Encoder Positionen
Strom, SOLL- und IST Geschwindigkeit, etc.

Edge Computing

Phase 3 - 5:

- Phase 3: DATENANALYSE DURCHFÜHRUNG
- Phase 4: KPI Vorhersage
- Phase 5: Prozessverbesserungen

Abschluss

Vielen Dank für Ihre Aufmerksamkeit

Quellennachweis

- [1] https://www.counterpointresearch.com/5g-edge-computing-emerging-technology-slowly-transitioning-commercial-reality/
- [2] https://de.123rf.com/photo_28493217_abgebrochenen-bohrer-f%C3%BCr-metall-isoliert-auf-wei%C3%9Fem-hintergrund.html
- [3] Al-Turjman, Fadi. (2019). Edge Computing: From Hype to Reality. Internet resource.
- [4] https://documentation.mindsphere.io/resources/html/manage-my-sinumerik-edge-app-management/en-US/user-docu/industrialedge.html
- [5] https://www.sciencedirect.com/science/article/pii/S0959652621027566#fig5
- [6] https://ieeexplore.ieee.org/document/9080346

Project “FairMillData”

FAIR Milling Dataset for Open Milling Process Improvement

T. Klünsner*, L. Hanna*, E. Hagendorfer*, M. Mücke, S. Trabesinger**, F. Haas**

* Materials Center Leoben Forschung GmbH, Leoben, Austria

** Institute of Production Engineering, Graz University of Technology, Austria

Graz, 23.3.2023

Agenda

- Motivation & project goal
- Milling experiments
- Dataset
- Outlook

Motivation & project goal

Collect sensor data from milling processes

- Create & publish a set of sensor data available from machine tool instrumentation
- Label data with useful "metadata" that describe their collection conditions
- Enable future data-analysis-based optimization of milling operations

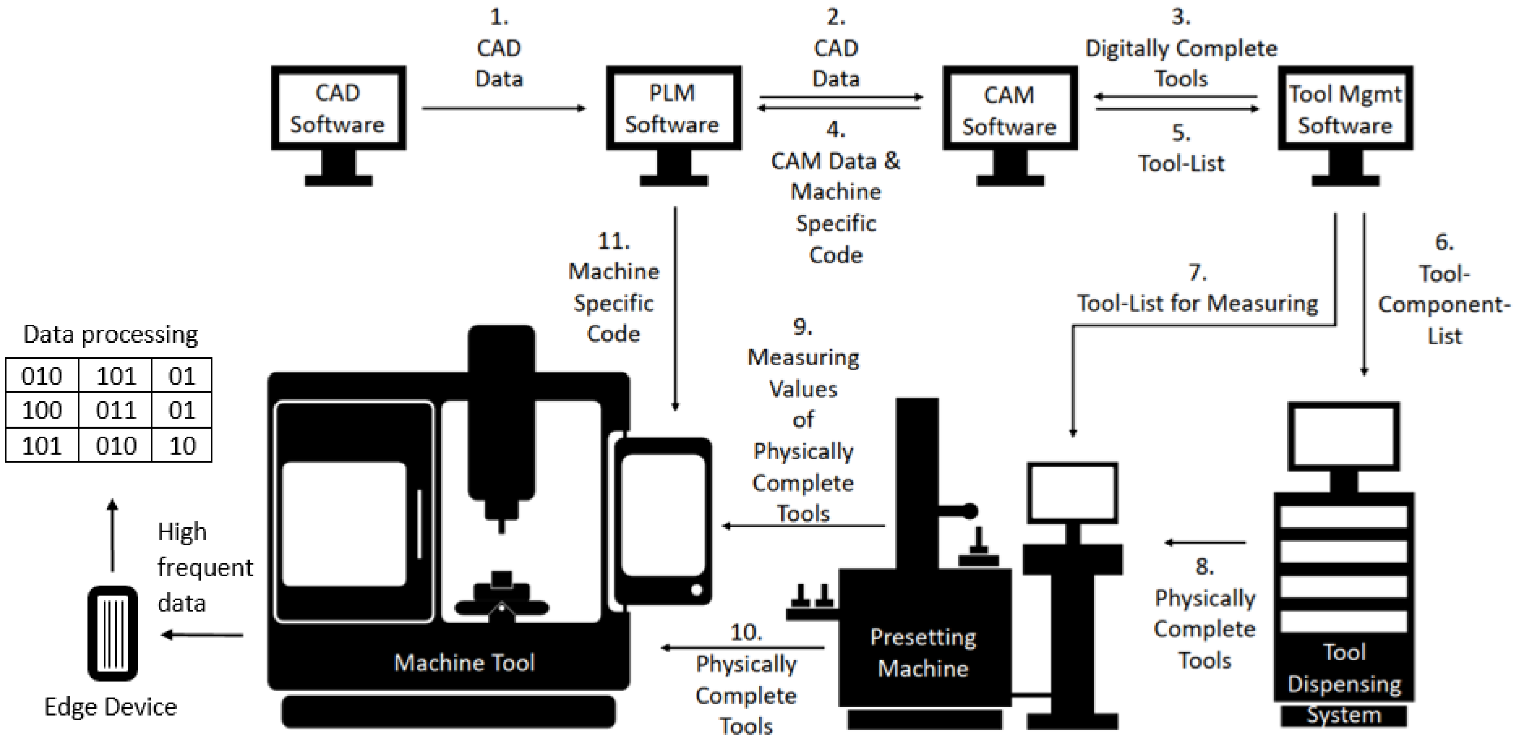
Possible milling process optimization goals

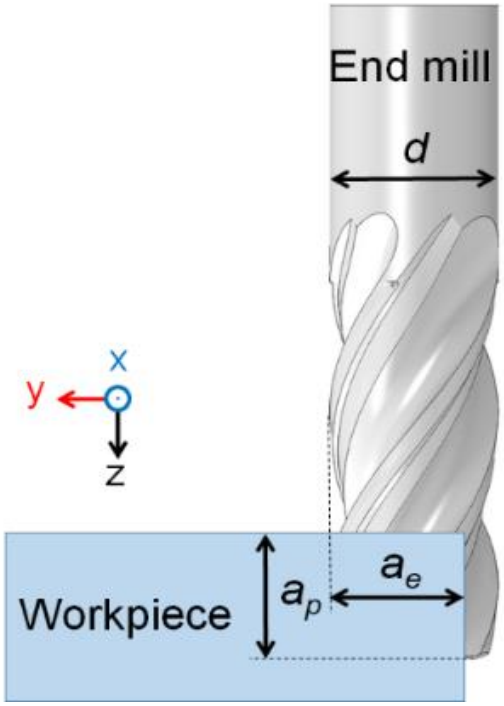
Future sensor-data-based analysis of impact of variations in milling parameters on quantities such as:

- Energy consumption
- Positional accuracy
- Actual tool acceleration patterns
- Tool cutting edge loads as f(tool path)
- Machine tool motor loads

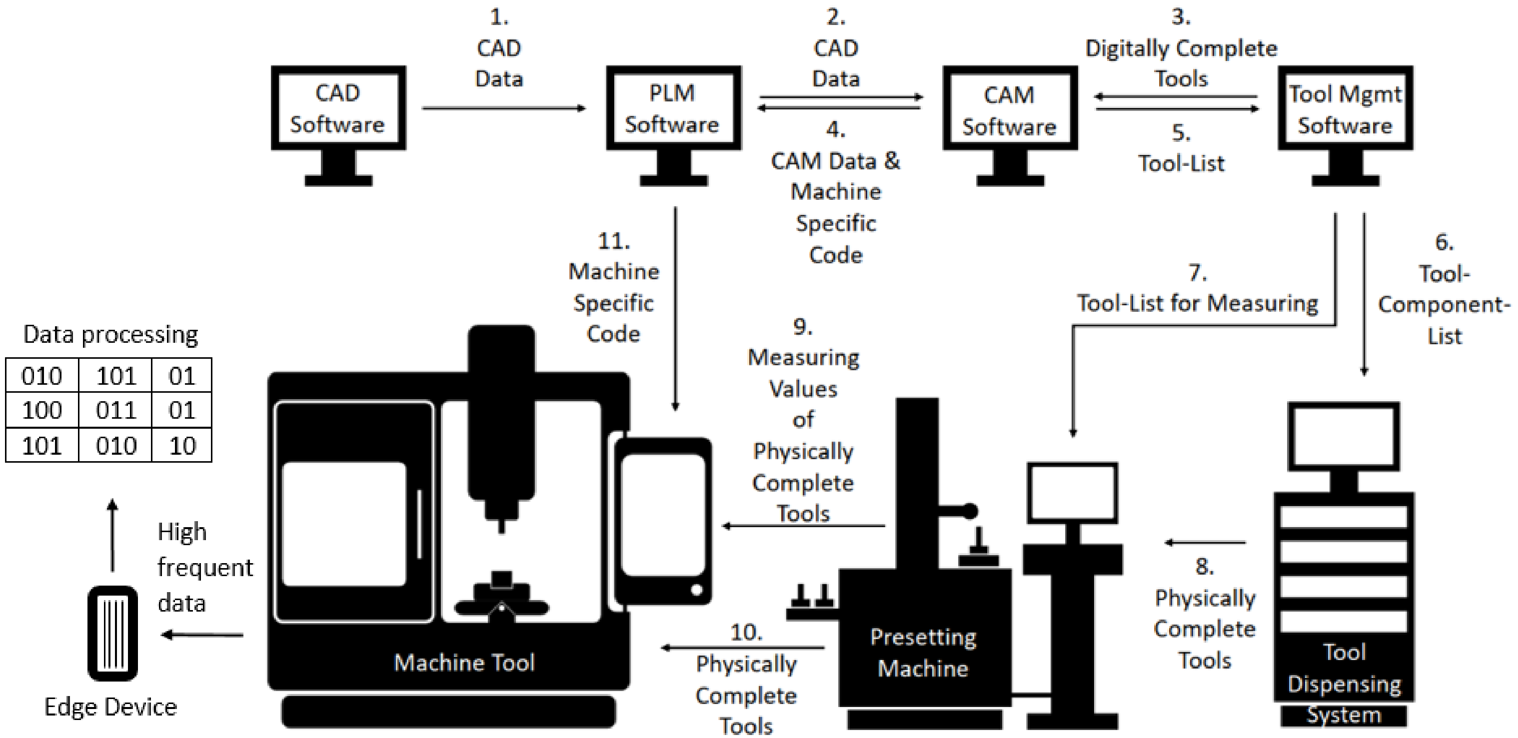
Motivation & project goal

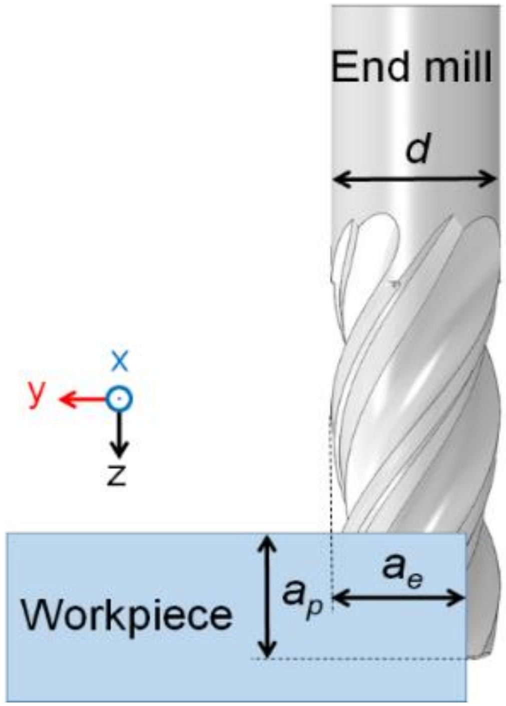
FairMillData dataset





- Well structured “metadata” prerequisite for future data analysis
- Metadata describe the boundary conditions of (ideally) the complete milling process chain

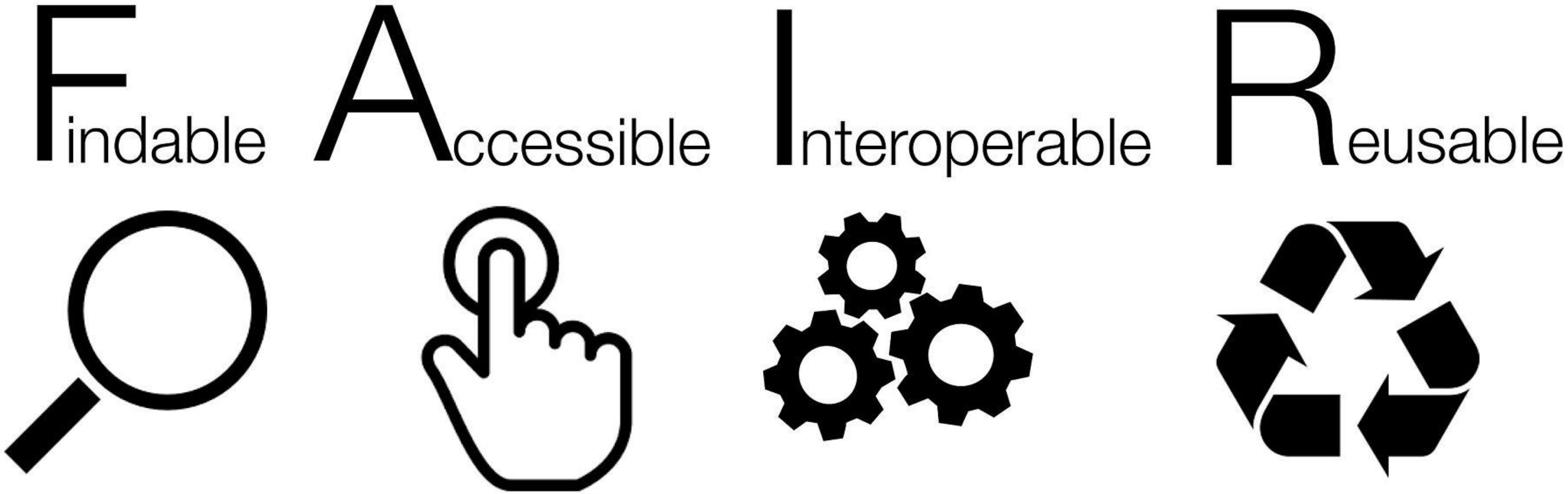




"FairMillData" dataset contains description of code versions necessary for milling process execution, variation in milling parameters, workpiece material, milling tool geometry, etc.

Motivation & project goal

FAIR principles

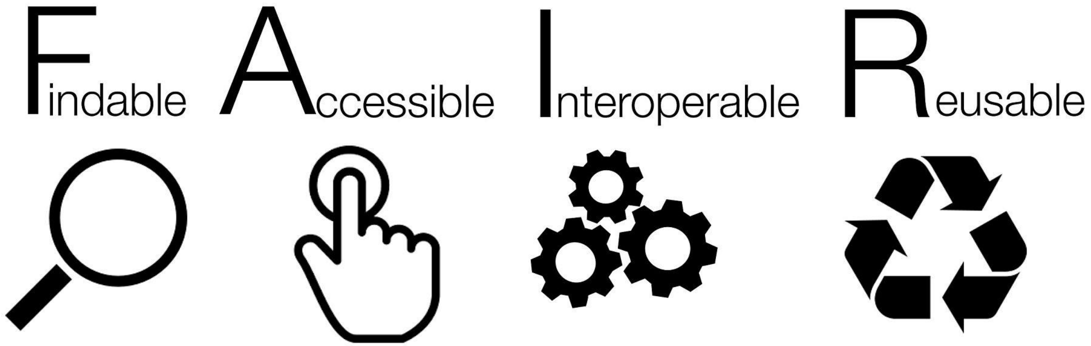


- FAIR principles aim at maximizing effective (re)use of acquired datasets

  → [Wilkinson2016], https://www.go-fair.org/fair-principles/

- We believe milling needs many datasets, used by many!

M. Wilkinson, *et al. Sci Data* **3** (2016) 160018.



Publish dataset on Zenodo.org

- Receives DOI (digital object identifier) → Findable
- Funded by European Commission → Accessible

Metadata:

- use precise & understandable language → Interoperable
- are richly described with a plurality of accurate and relevant attributes → Reusable

Agenda

- Motivationv & project goal
- Milling experiments
- Dataset
- Outlook

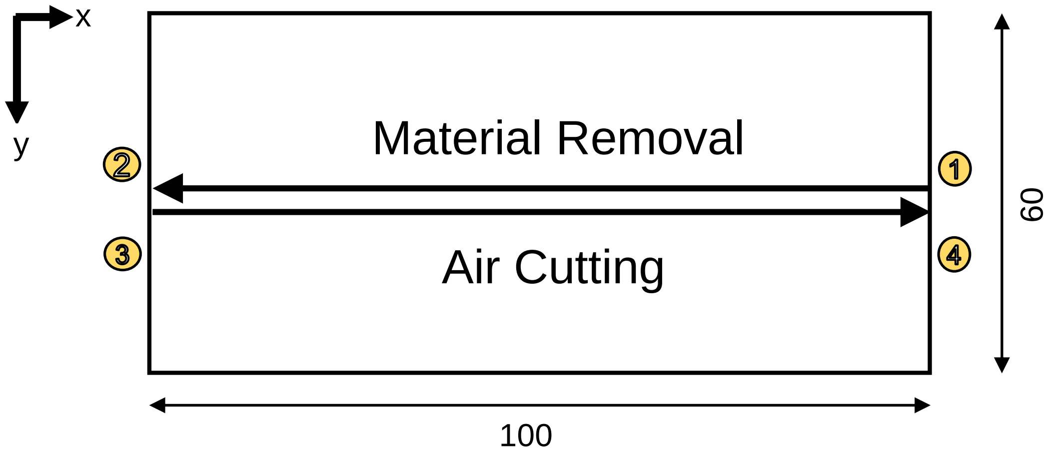
Milling experiments

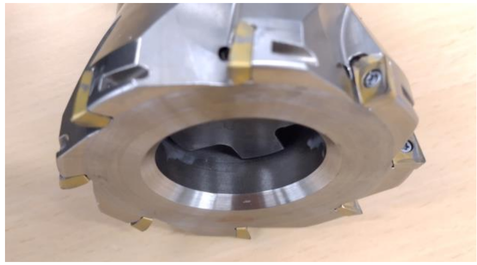
Overview

	Face Milling	End Milling	Plunge Milling	Slot milling
Varied Parameters	Feed Rate $\mathbf{v_f}$ Cutting Speed $\mathbf{v_c}$	Radial Depth $\mathbf{a_e}$ Axial Depth $\mathbf{a_p}$	Plunging Strategy	Radial Depth $\mathbf{a_e}$ Axial Depth $\mathbf{a_p}$ Corner Velocity $\mathbf{v_{EP}}$
Number of Experiments (Runs)	48	10	12	20
Tool	Indexable Cutter (d = 80 mm)	End mill (d = 10 mm)	End mill (d = 10 mm)	End mill (d = 10 mm)
Material	1.2083 (X42Cr 13) Stainless Mold Steel			

- Four milling operations are performed using a different set of parameters
- Dataset allows studying the effect of parameter variation on milling target quantities, e.g. energy consumption

	Cutting Speed	Spindle Speed	Feed	Feed per Tooth	Feed Rate
Run Index	v_c [m/min]	n [1/min]	f [mm]	f_z [mm]	v_f [mm/min]
1-8	80,00	318,31	0,80 – 2,48	0,10 – 0,31	254,65 – 789,41
9-16	100,00	397,89	0,80 – 2,48	0,10 – 0,31	318,31 – 986,76
17-24	120,00	477,76	0,80 – 2,48	0,10 – 0,31	381,97 – 1184,11
25-32	80,00 – 108,00	318,31 – 429,72	0,80	0,10	254,65 – 343,77
33-40	80,00 – 108,00	318,31 – 429,72	2,00	0,25	636,62 – 859,44
41-48	80,00 – 108,00	318,31 – 429,72	2,96	0,37	942,20 – 1271,97

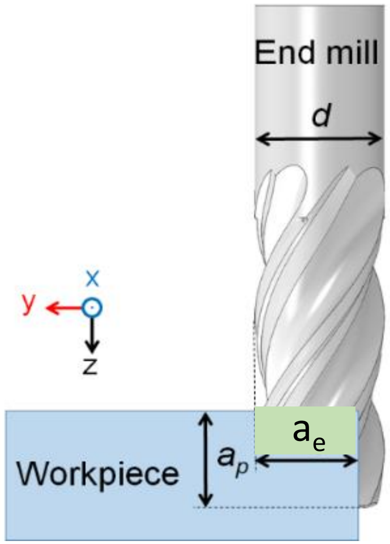




$n_{Inserts} = 8$
$d_{Tool} = 80$ mm

- For runs 1-24, feed parameters are varied with constant speed parameters
- For runs 25 – 48, speed parameters are varied with constant feed parameters

		v_c = constant = 100 m/min f_z = constant = 0,05 mm
Run 1	a_e = constant	2
	a_p = variable	10
Run 2	a_e = constant	2
	a_p = variable	8
Run 3	a_e = constant	2
	a_p = variable	6
Run 4	a_e = constant	2
	a_p = variable	4
Run 5	a_e = constant	2
	a_p = variable	2



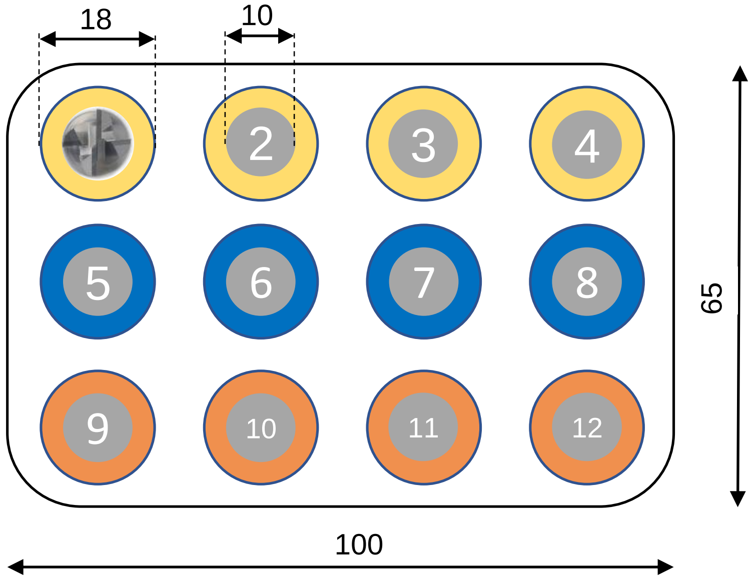
		v_c = constant = 100 m/min f_z = constant = 0,05 mm
Run 6	a_e = variable	10
	a_p = constant	2
Run 7	a_e = variable	8
	a_p = constant	2
Run 8	a_e = variable	6
	a_p = constant	2
Run 9	a_e = variable	4
	a_p = constant	2
Run 10	a_e = variable	2
	a_p = constant	2

- 10 runs of end milling are performed
- For runs 1 to 5, a_p is varied while a_e is kept at a constant level
- For runs 6 to 10, a_e is varied while a_p is kept at a constant level

Milling experiments

Plunge milling

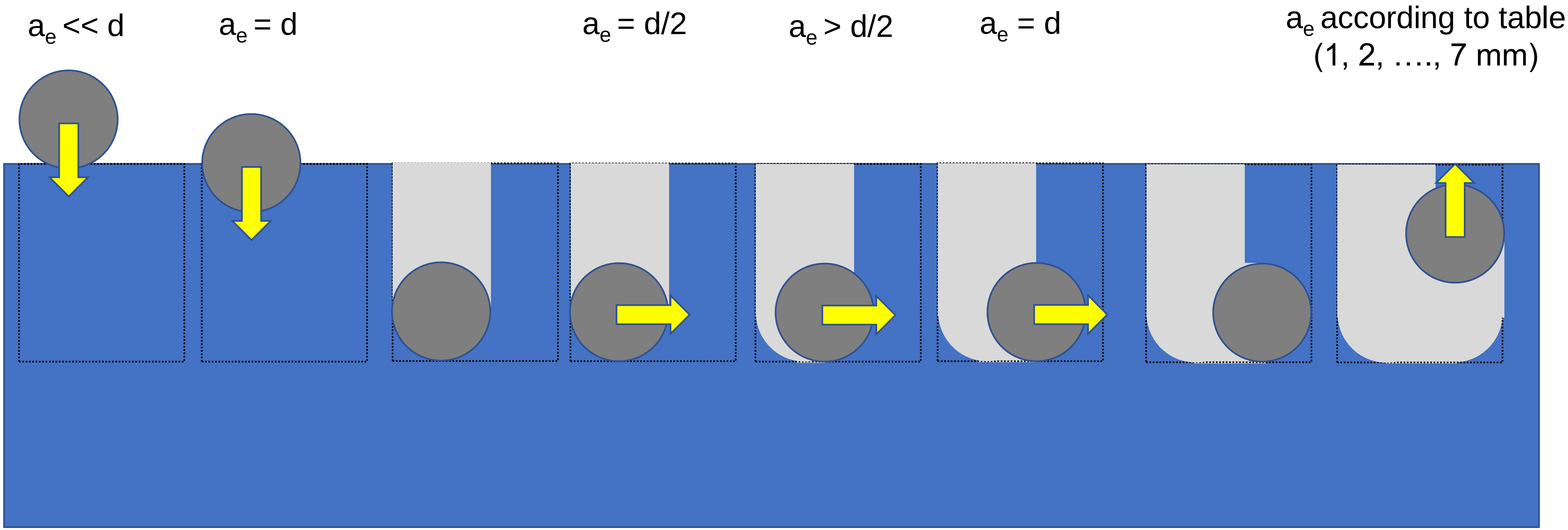
Run	a_p	Removal [mm]	Strategy	Immersion	Feed Rate v_f [mm/min]	Roughing
1	a_{p1}	3	Vertical Immersion	Vertical	50	Circular
2	a_{p2}	6		Vertical	25	Circular
3	a_{p3}	9		Vertical	50	Spiral & Circular Finish
4	a_{p4}	12		Vertical	25	Spiral & Circular Finish
5	a_{p1}	3	Diagonal Immersion	Diagonal	286,48	Circular
6	a_{p2}	6		Diagonal	286,48	Circular
7	a_{p3}	9		Diagonal	286,48	Spiral & Circular Finish
8	a_{p4}	12		Diagonal	286,48	Spiral & Circular Finish
9	a_{p1}	3	Helical Immersion	Helical	286,48	Circular
10	a_{p2}	6		Helical	286,48	Circular
11	a_{p3}	9		Helical	286,48	Spiral & Circular Finish
12	a_{p4}	12		Helical	286,48	Spiral & Circular Finish



- 12 cylindrical pockets are milled into a workpiece with varying strategies
- Change in immersion (tool entering the workpiece) and roughing (extending the diameter from 10 to 18 mm)
  - Immersion is performed in a vertical, diagonal or helical manner
  - Roughing is performed circular or spiral with a circular finish
- Reasoning behind experiment: low tool wear, practical, interesting with respect to energy consumption

Milling experiments

Slot milling - Operation principles



- Draft shows tool path while performing one-sided extension of slot
- Curve radius of 6 mm with d_{Tool} = 10 mm → simplification makes underlying phenomena easier understandable

Milling Experiments

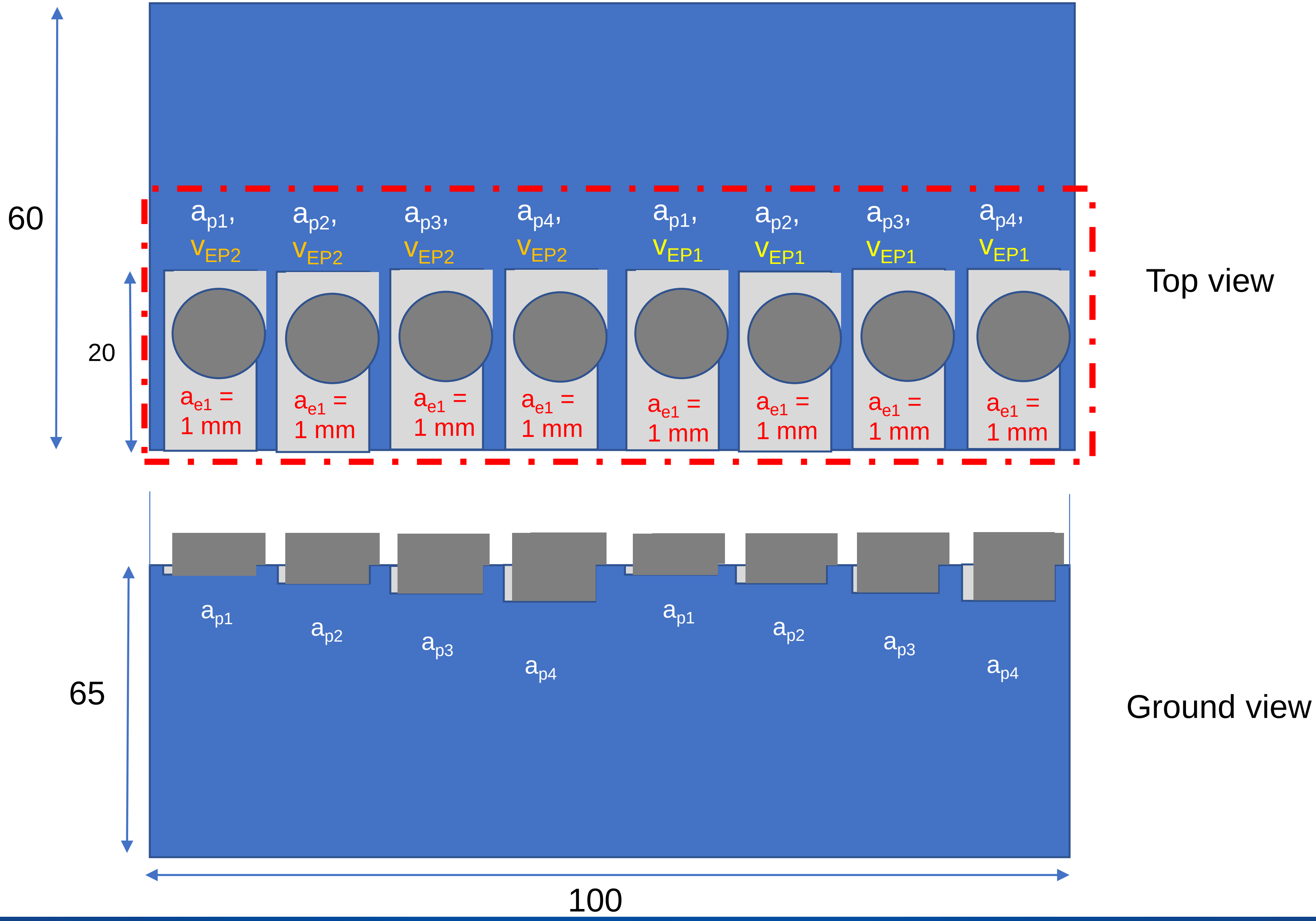
Slot milling - Parameter overview

Run		a_p in mm	v_{EP} in mm/min			a_e in mm
1	a_{p1}	1	v_{EP1}	80,21	ae_1	1
2	a_{p2}	2	v_{EP1}	80,21	ae_1	1
3	a_{p3}	3	v_{EP1}	80,21	ae_1	1
4	a_{p4}	4	v_{EP1}	80,21	ae_1	1
5	a_{p1}	1	v_{EP2}	187,17	ae_1	1
6	a_{p2}	2	v_{EP2}	187,17	ae_1	1
7	a_{p3}	3	v_{EP2}	187,17	ae_1	1
8	a_{p4}	4	v_{EP2}	187,17	ae_1	1
9	a_{p4}	4	v_{EP1}	80,21	ae_2	2
10	a_{p4}	4	v_{EP1}	80,21	ae_3	3
11	a_{p4}	4	v_{EP1}	80,21	ae_4	4
12	a_{p4}	4	v_{EP1}	80,21	ae_5	5
13	a_{p4}	4	v_{EP1}	80,21	ae_6	6
14	a_{p4}	4	v_{EP1}	80,21	ae_7	7
15	a_{p4}	4	v_{EP2}	187,17	ae_2	2
16	a_{p4}	4	v_{EP2}	187,17	ae_3	3
17	a_{p4}	4	v_{EP2}	187,17	ae_4	4
18	a_{p4}	4	v_{EP2}	187,17	ae_5	5
19	a_{p4}	4	v_{EP2}	187,17	ae_6	6
20	a_{p4}	4	v_{EP2}	187,17	ae_7	7

- 20 milling runs overall
- Two different corner velocities (v_{EP}) are tried:
  - Run 1 - 4: v_{EP1} with varying a_p and constant a_e
  - Run 5 - 8: v_{EP2} with varying a_p and constant a_e
  - Run 9 -14: v_{EP1} with constant a_p and varying a_e
  - Run 15 -20: v_{EP2} with constant a_p and varying a_e
- Slot milling consisting of
  - Immersion
  - Synchronous milling (Counterclockwise)

Milling Experiments

Slot milling - Operation details (1/3)



Process parameters:

v_f = 3055 mm / min
f = 1,2 mm
v_c = 80 mm/ min
d = 10 mm

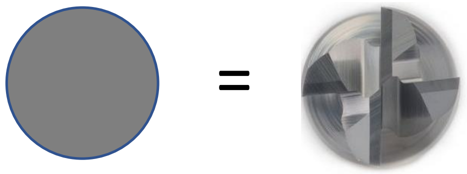
Corner velocity:

$v_{EP1} < v_{EP2}$
$v_{EP1} = 0.3 * v_f$
$v_{EP2} = 0.7 * v_f$

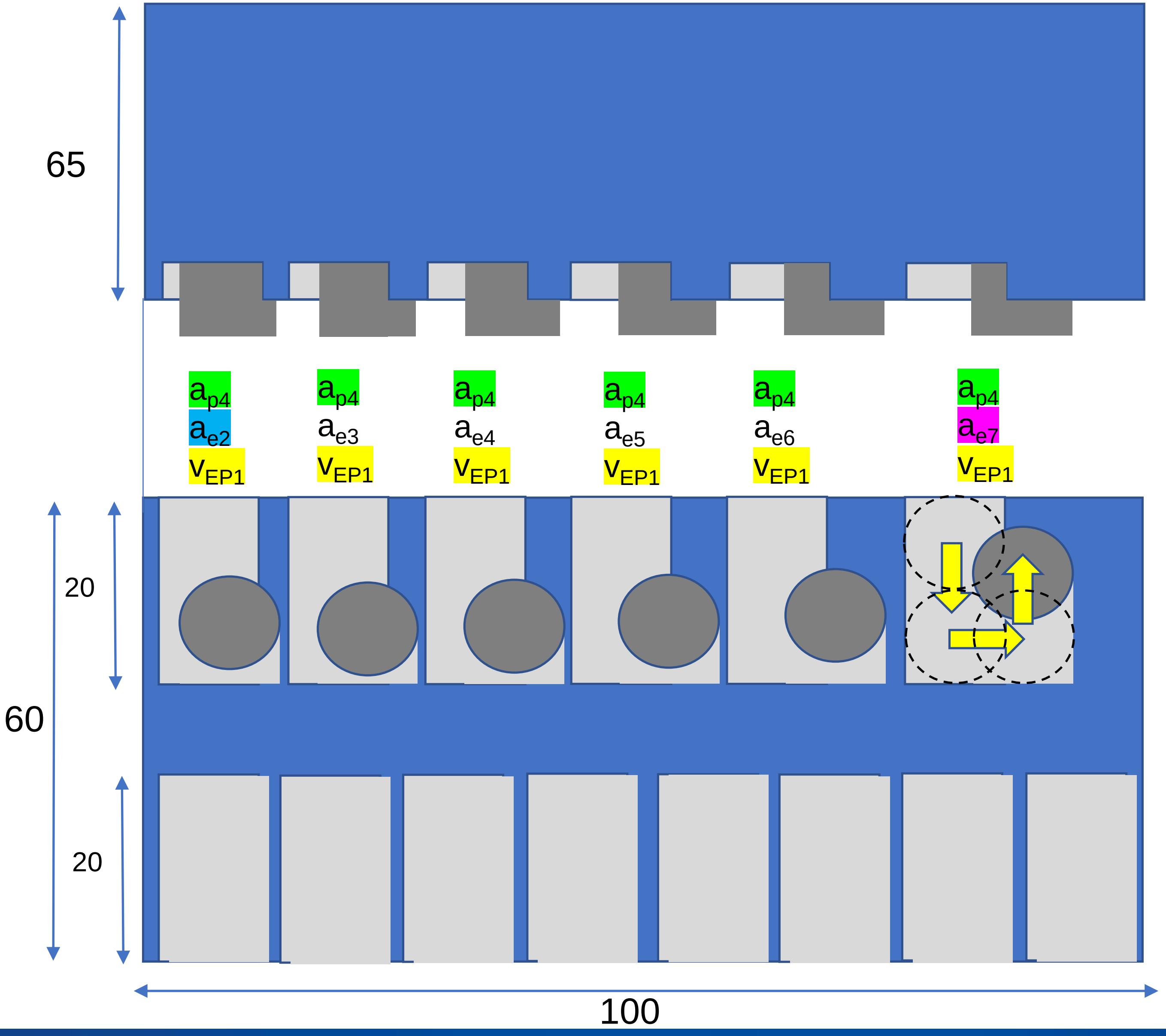
a_p & a_e

a_{p1} = 1 mm
a_{p2} = 2 mm
a_{p3} = 3 mm
a_{p4} = 4 mm

a_{e1} = 1 mm
a_{e2} = 2 mm
a_{e3} = 3 mm
a_{e4} = 4 mm



Slot milling - Operation details (2/3)



Ground view

Process parameters:

v_f = 3055 mm / min
f = 1,2 mm
v_c = 80 mm/ min
d = 10 mm

Corner velocity:

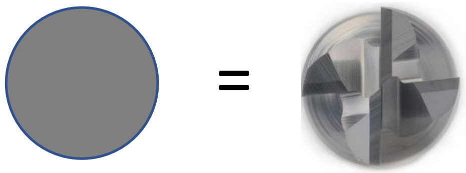
$v_{EP1} < v_{EP2}$
$v_{EP1} = 0.3 * v_f$
$v_{EP2} = 0.7 * v_f$

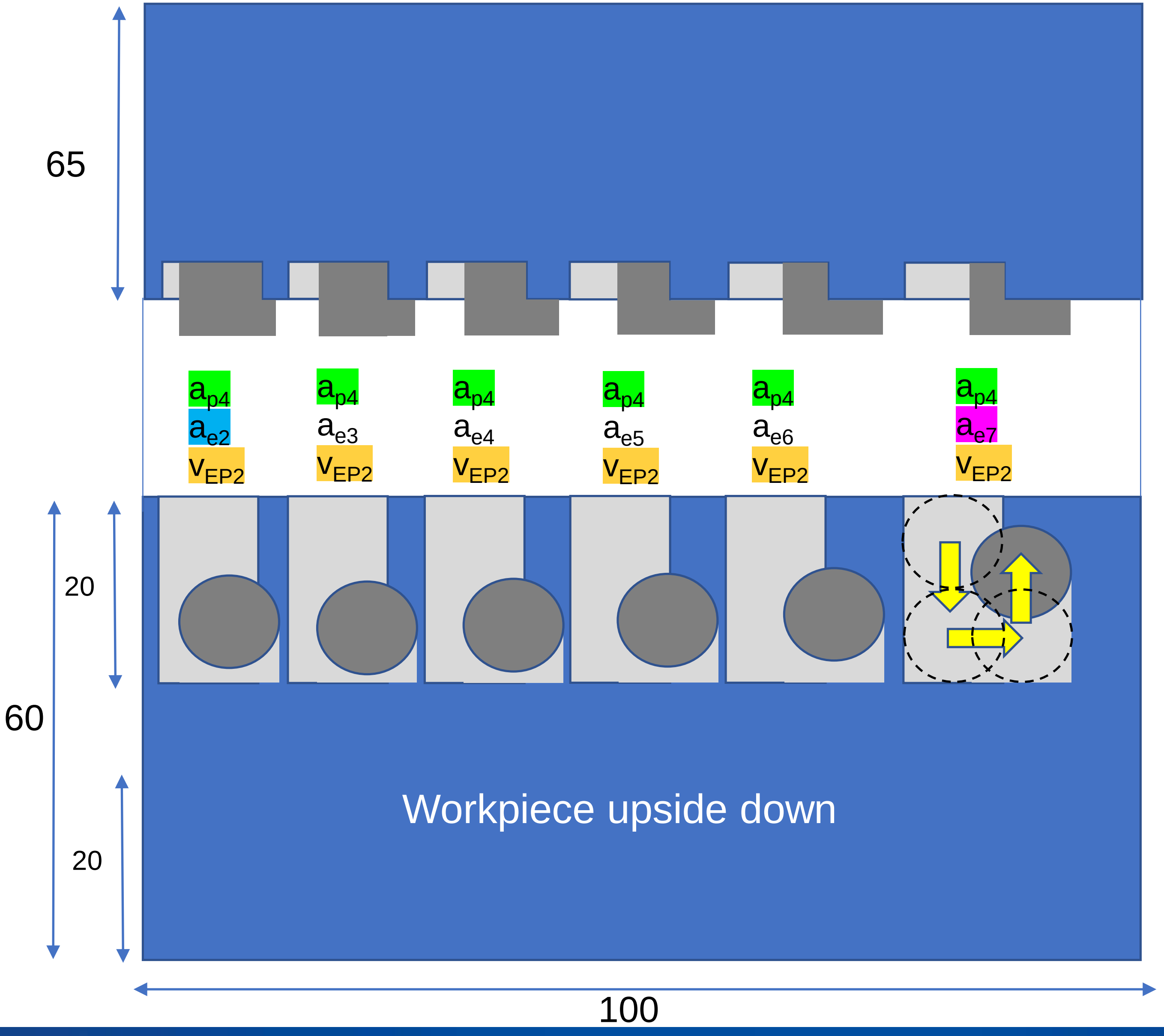
a_p & a_e

a_{p1} = 1 mm
a_{p2} = 2 mm
a_{p3} = 3 mm
a_{p4} = 4 mm

a_{e1} = 1 mm
a_{e2} = 2 mm
a_{e3} = 3 mm
a_{e4} = 4 mm
a_{e5} = 5 mm
a_{e6} = 6 mm
a_{e7} = 7 mm

Top view





Ground view

Process parameters:

v_f = 3055 mm / min
f = 1,2 mm
v_c = 80 mm/ min
d = 10 mm

Corner velocity:

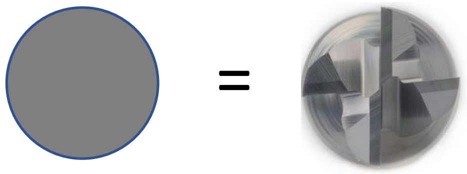
$v_{EP1} < v_{EP2}$
$v_{EP1} = 0.3 * v_f$
$v_{EP2} = 0.7 * v_f$

a_p & a_e

a_{p1} = 1 mm
a_{p2} = 2 mm
a_{p3} = 3 mm
a_{p4} = 4 mm

a_{e1} = 1 mm
a_{e2} = 2 mm
a_{e3} = 3 mm
a_{e4} = 4 mm
a_{e5} = 5 mm
a_{e6} = 6 mm
a_{e7} = 7 mm

Top view



Agenda

- Motivation & project goal
- Milling experiments
- Dataset
- Outlook

Dataset

Data structure

Dataset overview:

- 1_Hardware-Fingerprint
- 2_Software-Fingerprint
- 3_Machine-tool_tool-information
- 4_FACEMILLING-MEOP247
- 5_EndMILLING_MEOP245
- 6_PlungeMILLING_MEOP246
- 7_SlotMILLING_MEOP248

Size: 2.4 GB
Zipped size: ~1 GB

- 4.1_CAD-Files
- 4.2_CAM-Machining-Strategy
- 4.3_Toolpath-Simulation-before-Postprocessing
- 4.5_NC-Code-after-Postprocessing
- 4.6_NC-Code-after-check-at-machine-tool
- 5_High-frequent-machining-data
- 6_Photos_final-workpiece-REAL

facemilling_2023_02_22_1_df_.h5	HDF5 File
facemilling_2023_02_22_2_df_.h5	HDF5 File
⋮	
facemilling_2023_02_22_47_df_.h5	HDF5 File
facemilling_2023_02_22_48_df_.h5	HDF5 File

Face milling organized in 48 files. Each file is a hdf5 container with a python pandas DataFrame inside. Key is always „df“.

⋮
5_High-frequent-machining-data
⋮

slotmilling_2023_02_21_1_df.h5	HDF5 File
slotmilling_2023_02_21_2_df.h5	HDF5 File
⋮	
slotmilling_2023_02_21_19_df.h5	HDF5 File
slotmilling_2023_02_21_20_df.h5	HDF5 File

Slot milling organized in 20 files.

⋮
5_High-frequent-machining-data
⋮

plungemilling_2023_02_10_1_df.h5	HDF5 File
plungemilling_2023_02_10_2_df.h5	HDF5 File
⋮	
plungemilling_2023_02_13_11_df.h5	HDF5 File
plungemilling_2023_02_13_12_df.h5	HDF5 File

Plunge milling organized in 12 files.

⋮
5_High-frequent-machining-data
⋮

endmilling_2023_02_09_01_df.h5	HDF5 File
endmilling_2023_02_09_02_df.h5	HDF5 File
⋮	
endmilling_2023_02_09_09_df.h5	HDF5 File
endmilling_2023_02_09_10_df.h5	HDF5 File

End milling organized in 10 files.

Dataset

Data structure – JSON to HDF - Signals

Overview of all signals:

	Name	Type	Axis	Address
0	Cycle	INTEGER	CYCLE	CYCLE
1	ControlDiff2	DOUBLE	X1	CTRL_DIFF2|1
2	ControlDiff2	DOUBLE	Y1	CTRL_DIFF2|2
3	ControlDiff2	DOUBLE	Z1	CTRL_DIFF2|3
4	ControlDiff2	DOUBLE	B1	CTRL_DIFF2|4
5	ControlDiff2	DOUBLE	SP1	CTRL_DIFF2|5
6	ControlDiff2	DOUBLE	C1	CTRL_DIFF2|6
7	Torque	DOUBLE	X1	TORQUE|1
8	Torque	DOUBLE	Y1	TORQUE|2
9	Torque	DOUBLE	Z1	TORQUE|3
10	Torque	DOUBLE	B1	TORQUE|4
11	Torque	DOUBLE	SP1	TORQUE|5
12	Torque	DOUBLE	C1	TORQUE|6
13	CommandedAxisPosition	DOUBLE	X1	DES_POS|1
14	CommandedAxisPosition	DOUBLE	Y1	DES_POS|2
15	CommandedAxisPosition	DOUBLE	Z1	DES_POS|3
16	CommandedAxisPosition	DOUBLE	B1	DES_POS|4
17	CommandedAxisPosition	DOUBLE	SP1	DES_POS|5
18	CommandedAxisPosition	DOUBLE	C1	DES_POS|6
19	Current	DOUBLE	X1	CURRENT|1
20	Current	DOUBLE	Y1	CURRENT|2
21	Current	DOUBLE	Z1	CURRENT|3
22	Current	DOUBLE	B1	CURRENT|4
23	Current	DOUBLE	SP1	CURRENT|5
24	Current	DOUBLE	C1	CURRENT|6
25	ControlDiff	DOUBLE	X1	CTRL_DIFF|1
26	ControlDiff	DOUBLE	Y1	CTRL_DIFF|2
27	ControlDiff	DOUBLE	Z1	CTRL_DIFF|3
28	ControlDiff	DOUBLE	B1	CTRL_DIFF|4
29	ControlDiff	DOUBLE	SP1	CTRL_DIFF|5
30	ControlDiff	DOUBLE	C1	CTRL_DIFF|6

	Name	Type	Axis	Address
31	ControlPos	DOUBLE	X1	CTRL_POS|1
32	ControlPos	DOUBLE	Y1	CTRL_POS|2
33	ControlPos	DOUBLE	Z1	CTRL_POS|3
34	ControlPos	DOUBLE	B1	CTRL_POS|4
35	ControlPos	DOUBLE	SP1	CTRL_POS|5
36	ControlPos	DOUBLE	C1	CTRL_POS|6
37	VelocityFeedForward	DOUBLE	X1	VEL_FFW|1
38	VelocityFeedForward	DOUBLE	Y1	VEL_FFW|2
39	VelocityFeedForward	DOUBLE	Z1	VEL_FFW|3
40	VelocityFeedForward	DOUBLE	B1	VEL_FFW|4
41	VelocityFeedForward	DOUBLE	SP1	VEL_FFW|5
42	VelocityFeedForward	DOUBLE	C1	VEL_FFW|6
43	Power	STRING	X1	POWER|1
44	Power	STRING	Y1	POWER|2
45	Power	STRING	Z1	POWER|3
46	Power	STRING	B1	POWER|4
47	Power	STRING	SP1	POWER|5
48	Power	STRING	C1	POWER|6
49	CountourDeviation	DOUBLE	X1	CONT_DEV|1
50	CountourDeviation	DOUBLE	Y1	CONT_DEV|2
51	CountourDeviation	DOUBLE	Z1	CONT_DEV|3
52	CountourDeviation	DOUBLE	B1	CONT_DEV|4
53	CountourDeviation	DOUBLE	SP1	CONT_DEV|5
54	CountourDeviation	DOUBLE	C1	CONT_DEV|6
55	Synchronuous Action variable	INTEGER		A_DBD|0
56	CommandedSpeed	DOUBLE	X1	CMD_SPEED|1
57	CommandedSpeed	DOUBLE	Y1	CMD_SPEED|2
58	CommandedSpeed	DOUBLE	Z1	CMD_SPEED|3
59	CommandedSpeed	DOUBLE	B1	CMD_SPEED|4
60	CommandedSpeed	DOUBLE	SP1	CMD_SPEED|5

	Name	Type	Axis	Address
61	CommandedSpeed	DOUBLE	C1	CMD_SPEED|6
62	TorqueFeedForward	DOUBLE	X1	TORQUE_FFW|1
63	TorqueFeedForward	DOUBLE	Y1	TORQUE_FFW|2
64	TorqueFeedForward	DOUBLE	Z1	TORQUE_FFW|3
65	TorqueFeedForward	DOUBLE	B1	TORQUE_FFW|4
66	TorqueFeedForward	DOUBLE	SP1	TORQUE_FFW|5
67	TorqueFeedForward	DOUBLE	C1	TORQUE_FFW|6
68	Encoder1Position	DOUBLE	X1	ENC1_POS|1
69	Encoder1Position	DOUBLE	Y1	ENC1_POS|2
70	Encoder1Position	DOUBLE	Z1	ENC1_POS|3
71	Encoder1Position	DOUBLE	B1	ENC1_POS|4
72	Encoder1Position	DOUBLE	SP1	ENC1_POS|5
73	Encoder1Position	DOUBLE	C1	ENC1_POS|6
74	Encoder2Position	DOUBLE	X1	ENC2_POS|1
75	Encoder2Position	DOUBLE	Y1	ENC2_POS|2
76	Encoder2Position	DOUBLE	Z1	ENC2_POS|3
77	Encoder2Position	DOUBLE	B1	ENC2_POS|4
78	Encoder2Position	DOUBLE	SP1	ENC2_POS|5
79	Encoder2Position	DOUBLE	C1	ENC2_POS|6
80	Load	DOUBLE	X1	LOAD|1
81	Load	DOUBLE	Y1	LOAD|2
82	Load	DOUBLE	Z1	LOAD|3
83	Load	DOUBLE	B1	LOAD|4
84	Load	DOUBLE	SP1	LOAD|5
85	Load	DOUBLE	C1	LOAD|6
86	ActualAxisPosition	DOUBLE	X1	ENC_POS|1
87	ActualAxisPosition	DOUBLE	Y1	ENC_POS|2
88	ActualAxisPosition	DOUBLE	Z1	ENC_POS|3
89	ActualAxisPosition	DOUBLE	B1	ENC_POS|4
90	ActualAxisPosition	DOUBLE	SP1	ENC_POS|5
91	ActualAxisPosition	DOUBLE	C1	ENC_POS|6

JSON data is converted to a Pandas (python library) DataFrame (tabular data structure):

The unit of the signal is appended at the end of each signal name.

Index in seconds:

TIME_s	CYCLE_CYCLE_counter	DES_POS_1_X1_mm	DES_POS_2_Y1_mm	DES_POS_3_Z1_mm	DES_POS_4_B1_deg	DES_POS_5_SP1_deg	DES_POS_6_C1_deg
0.000	2005766	89.651664	-123.583230	315.951358	0.0	2857347.444	0.0
0.002	2005767	89.651664	-123.583044	315.951358	0.0	2857352.592	0.0
0.004	2005768	89.651664	-123.582537	315.951358	0.0	2857357.740	0.0
0.006	2005769	89.651664	-123.581550	315.951358	0.0	2857362.888	0.0
0.008	2005770	89.651664	-123.579924	315.951358	0.0	2857368.036	0.0

The JSON contains „BlockEvents“ which then contain G-code.
The G-code is extracted, matched with the HFProbeCounter & CYCLE_counter and then added to the DataFrame:

Example BlockEvent:

```
{'HFBlockEvent': {'HFProbeCounter': 1384042,
  'Channel': 1,
  'SeekOffset': 31,
  'SelectedTool': 40,
  'ActiveTool': 40,
  'GCode': 'G4 F1;START',
  'IpoGC': 'G0',
  'ipoReadError': None,
  'laBuf': 148}}
```

G-code added to DataFrame:

	CYCLE_CYCLE_counter	DES_POS_1_X1_mm	...	TORQUE_6_C1_Nm	G-code
TIME_s					
0.000	1384042	89.651664		-0.522540	G4 F1;START
0.002	1384043	89.651664		-0.522540	G4 F1;START
0.004	1384044	89.651664	...	-0.522540	G4 F1;START
0.006	1384045	89.651664		-0.527291	G4 F1;START
0.008	1384046	89.651664		-0.522540	G4 F1;START
...	...	...		...	...

G-code stays active until another G-code line is activated.

Data structure – JSON to HDF – Reading Data

Data can be easily loaded via python and pandas:

```
import pandas as pd
path = r"...\facemilling_2023_02_22_1_df.h5"
df = pd.read_hdf(path, key = "df")
df
```

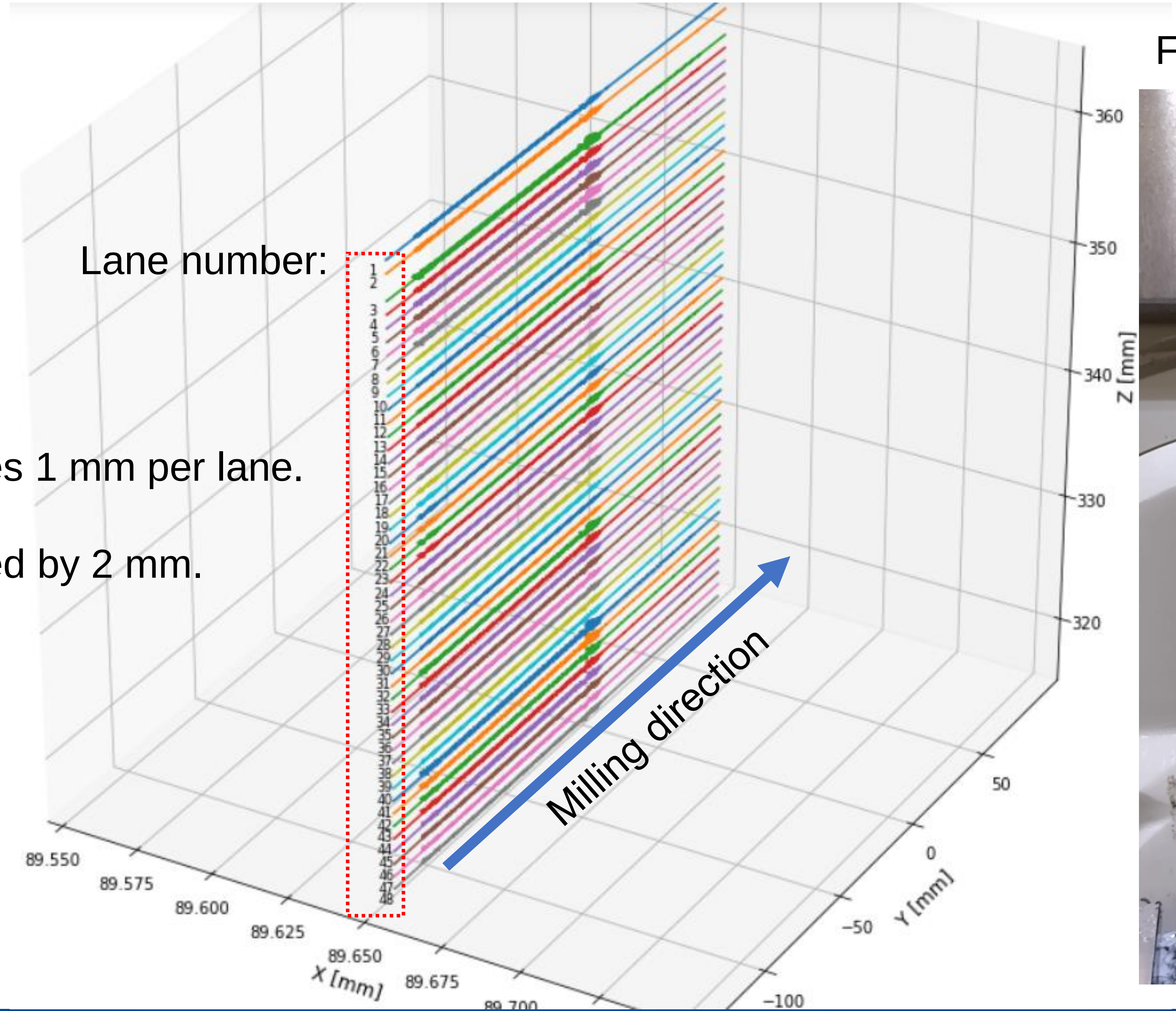
	CYCLE_CYCLE_counter	DES_POS_1_X1_mm	DES_POS_2_Y1_mm	DES_POS_3_Z1_mm	DES_POS_4_B1_deg	DES_POS_5_SP1_deg	DES_POS_6_C1_deg
TIME_s							
0.000	1384994	89.651664	-123.582230	363.951358	0.0	36808.680	0.0
0.002	1384995	89.651664	-123.582044	363.951358	0.0	36812.496	0.0
0.004	1384996	89.651664	-123.581537	363.951358	0.0	36816.312	0.0
...	...	...	...	...	...	...	...
50.258	1410123	89.651664	89.268743	363.951358	0.0	132700.944	0.0
50.260	1410124	89.651664	89.268743	363.951358	0.0	132704.760	0.0
50.262	1410125	89.651664	89.268743	363.951358	0.0	132708.576	0.0

25132 rows × 93 columns

Face milling – Path tool center point

48 lanes.
Z level decreases 1 mm per lane.
Error for lane 3:
Z level decreased by 2 mm.

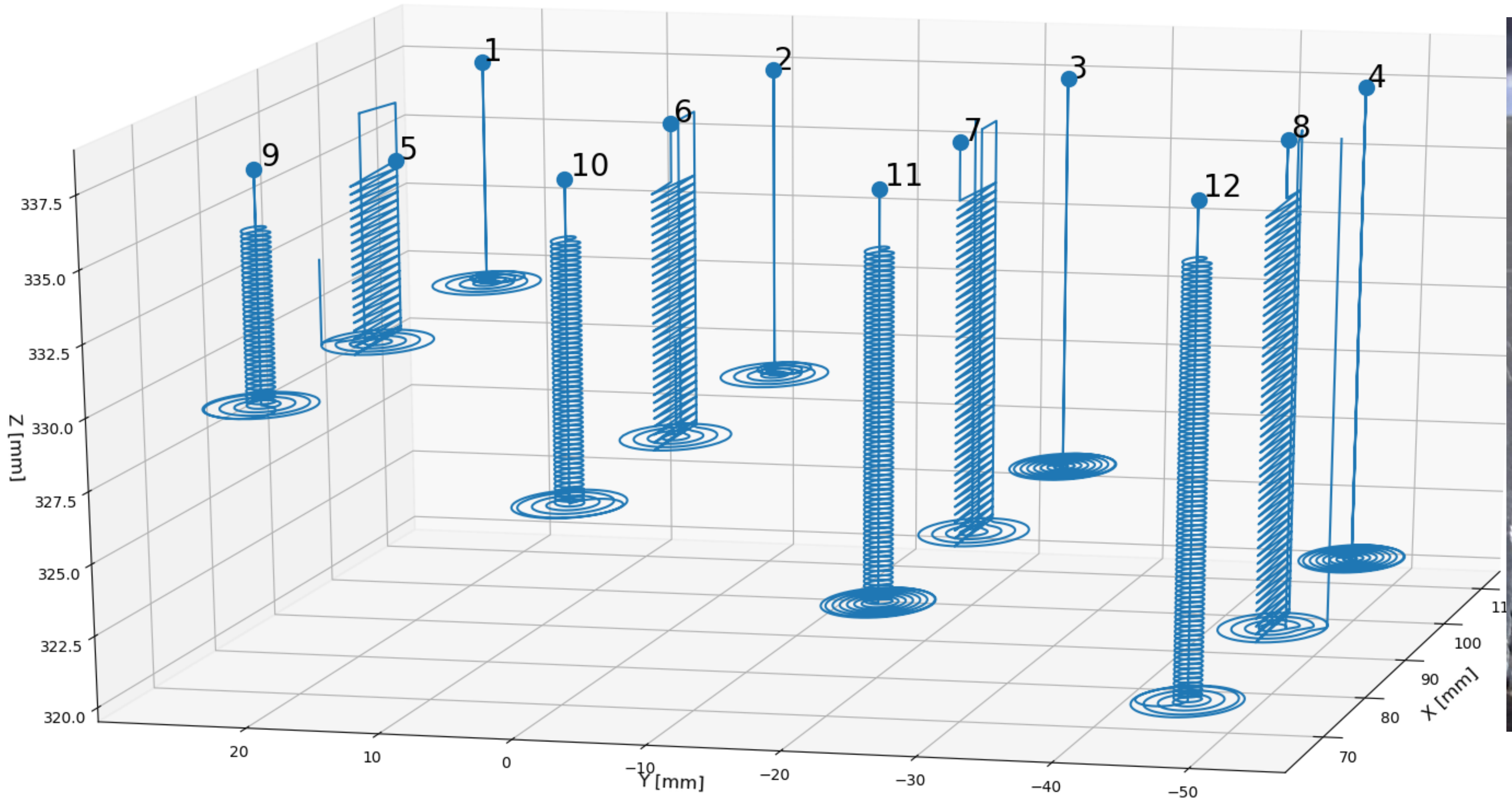
Lane number:



Final workpiece with tool:

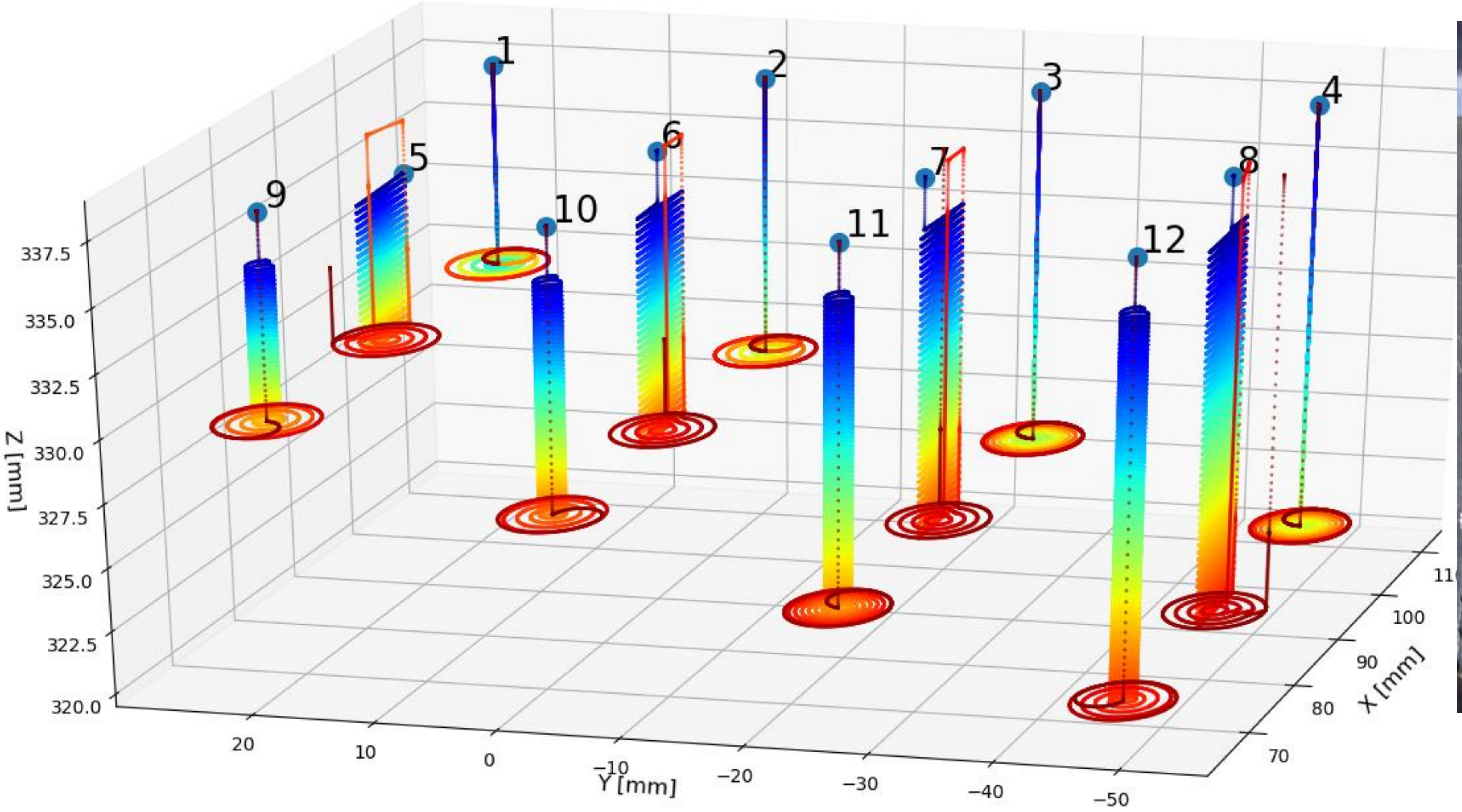
Dataset

Plunge milling strategy – Path tool center point



Different immersion strategies applied to holes 1 - 4, 5 - 8 & 9 - 12 use .

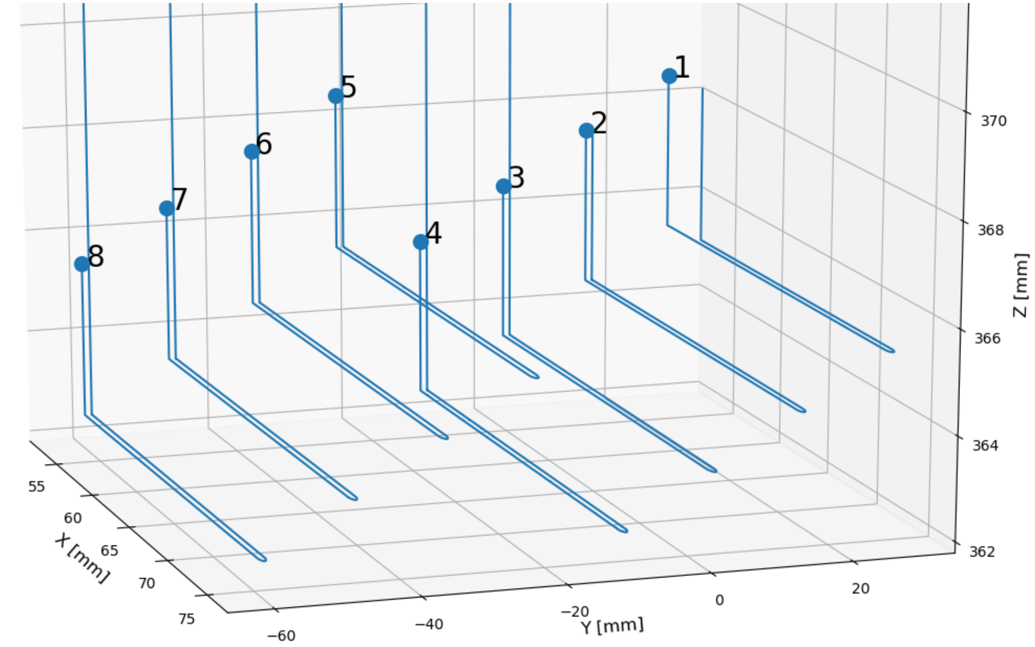
Plunge milling strategy – Path tool center point



Tool center point position visualized from early plunge stage to late plunge stage.

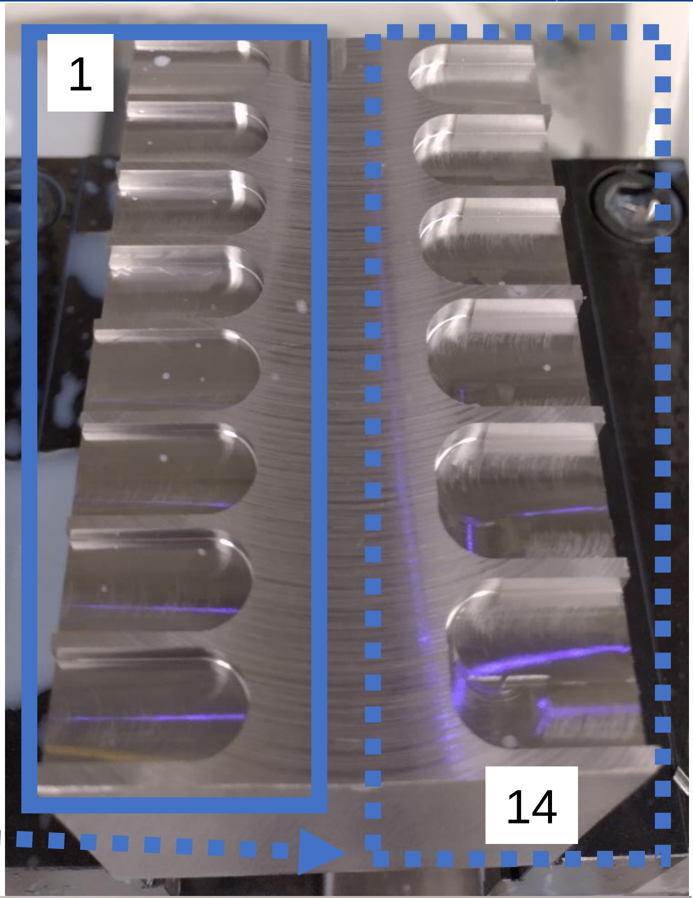
Dataset

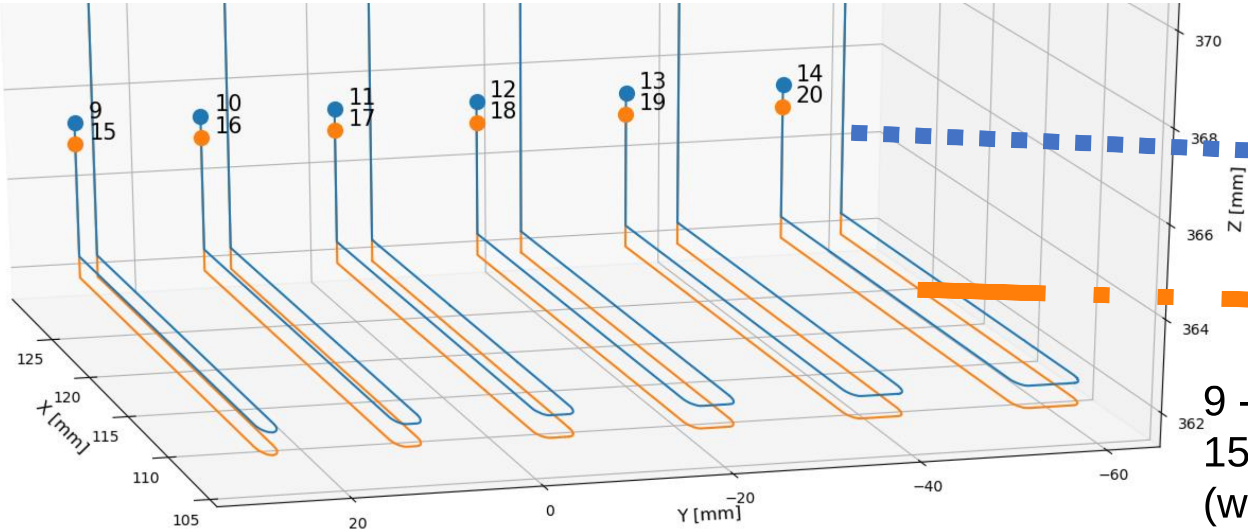
Slot milling strategy – Path tool center point

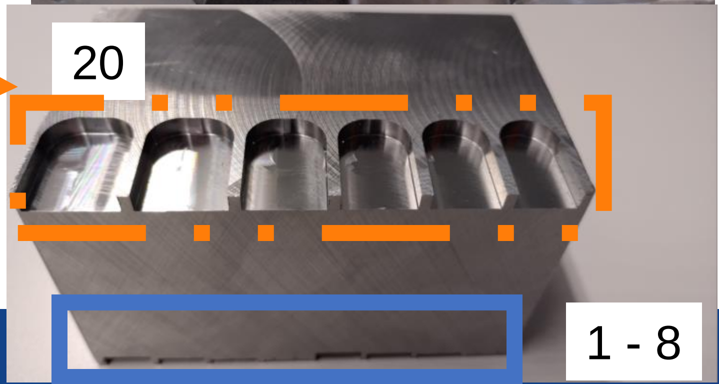


Two different corner velocities (v_{EP}) :
1 - 4: v_{EP1}, varying a_p & constant a_e
5 - 8: v_{EP2}, varying a_p & constant a_e
9 - 14: v_{EP1}, constant a_p 6 varying a_e
15 - 20: v_{EP2}, constant a_p & varying a_e

1 - 8 are milled on the left side





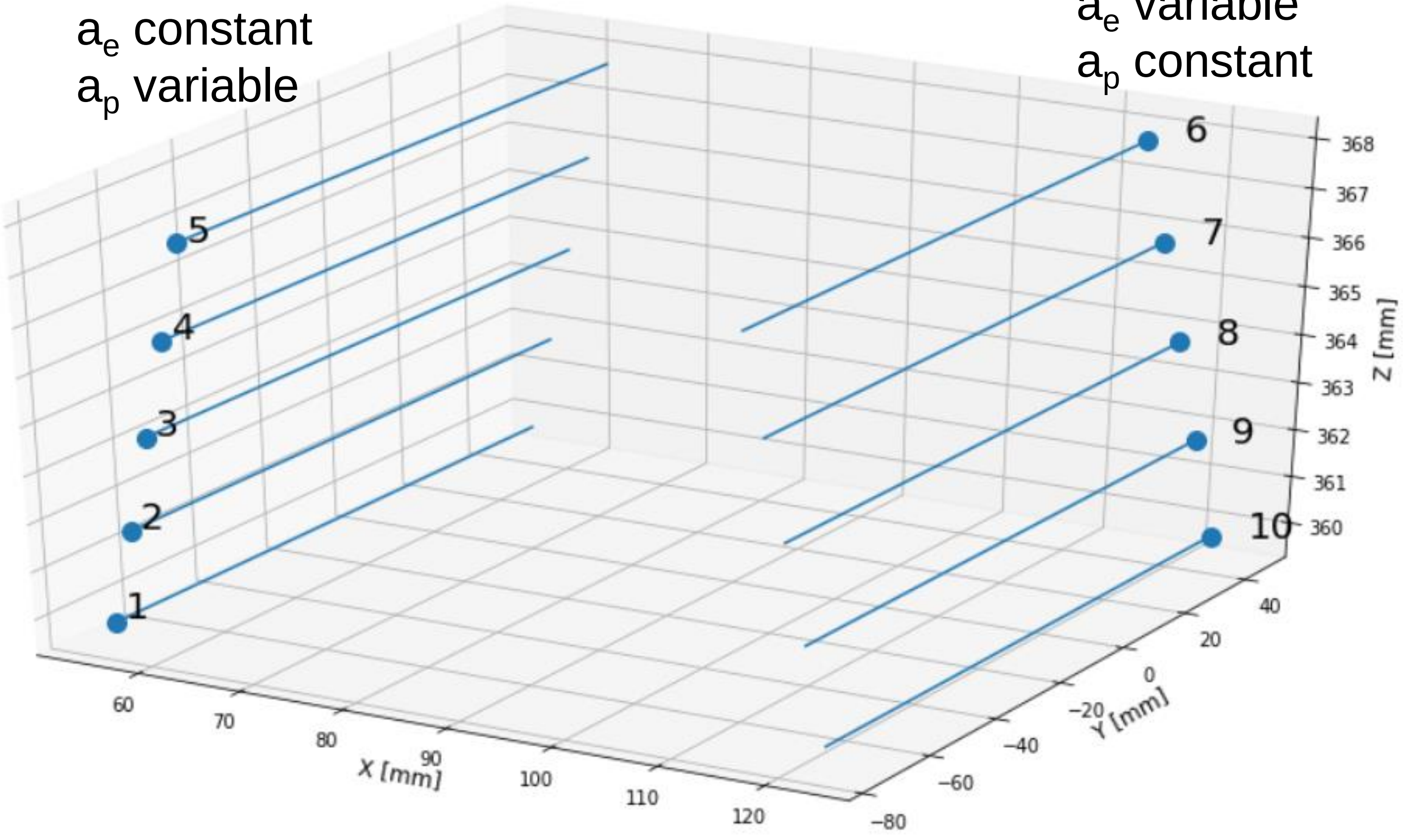


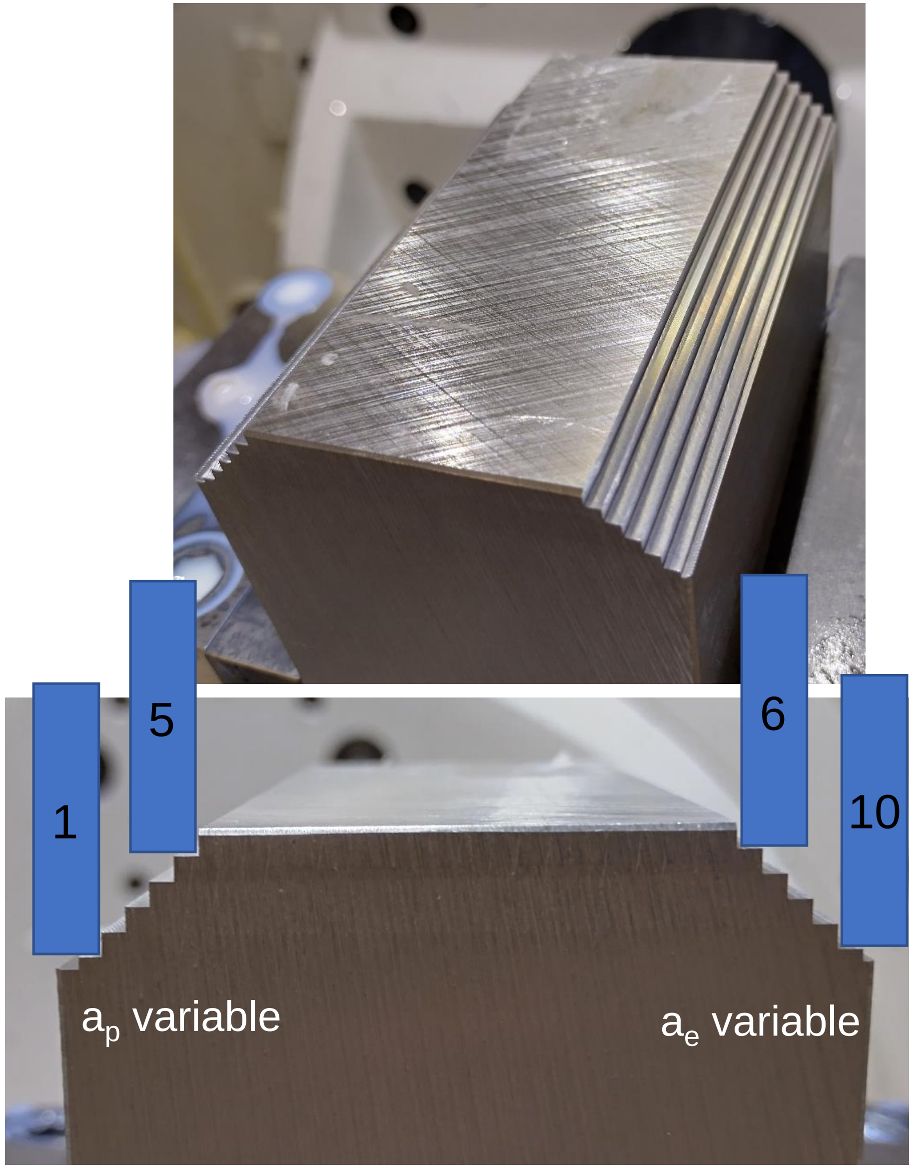
9 - 14 are milled on the right side
15 - 20 are milled on the bottom
(work piece flipped)

End milling strategy – Path tool center point

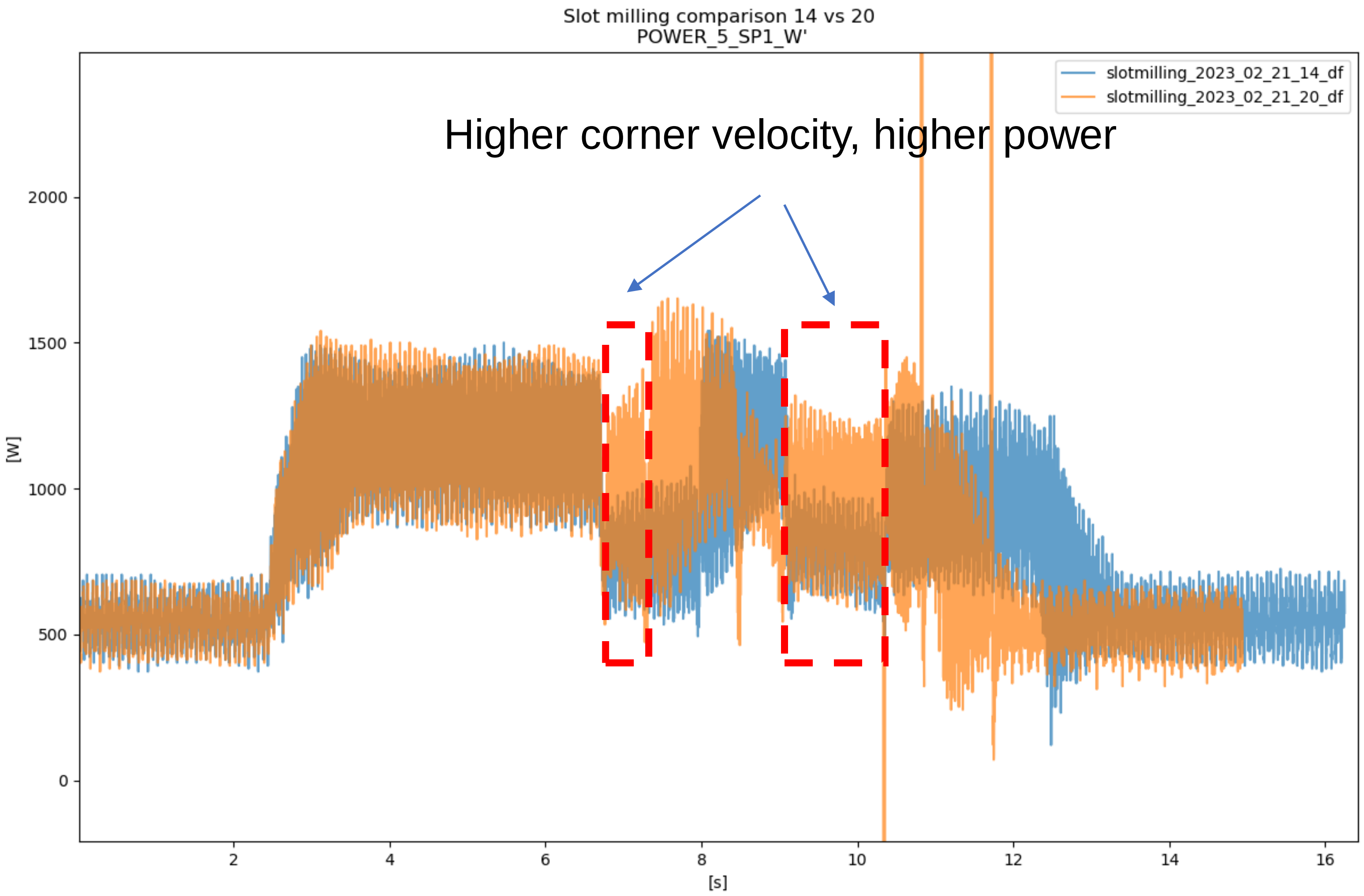
For 1 - 5:
a_e constant
a_p variable

For 6 - 10:
a_e variable
a_p constant



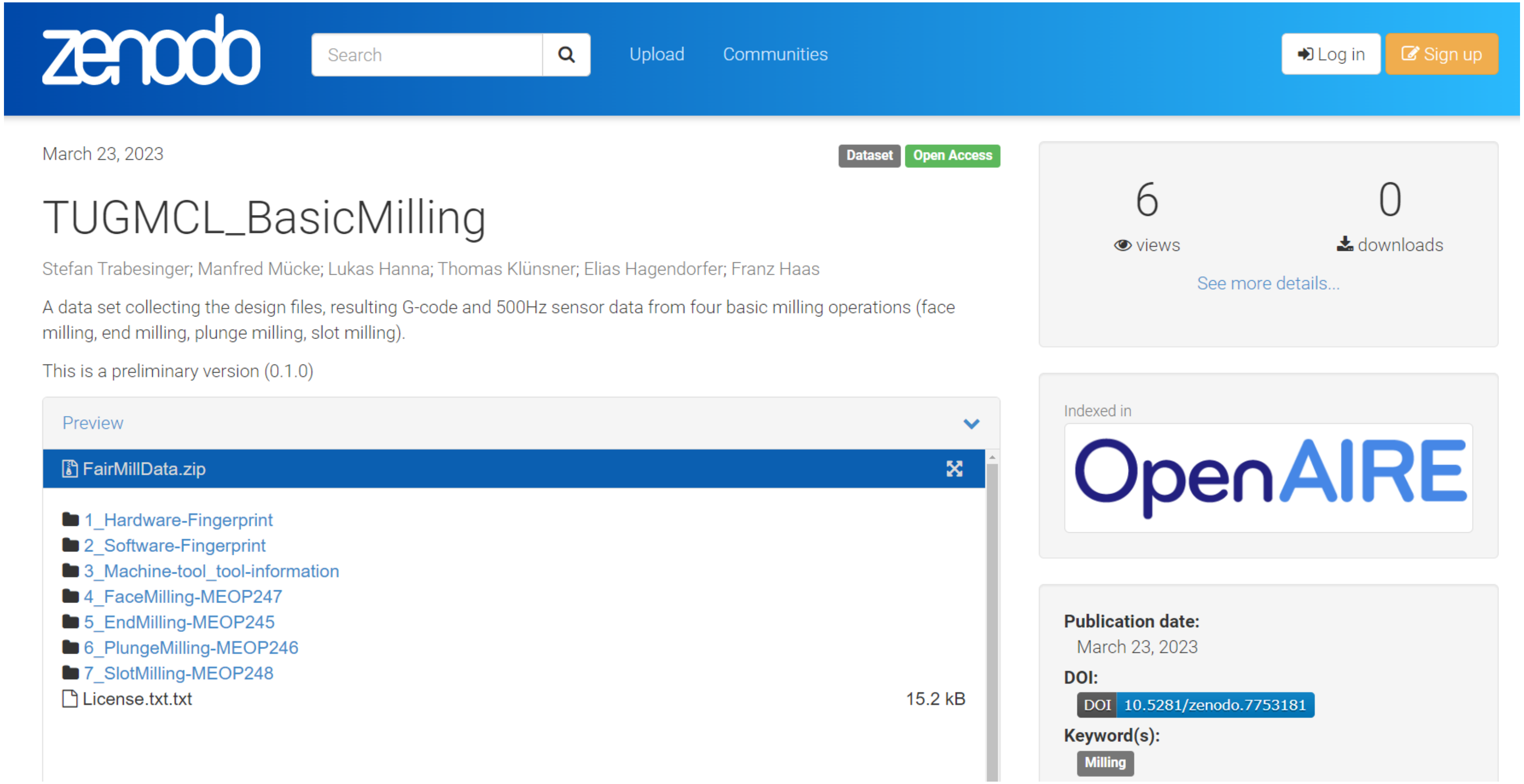


Energy / power as f(t) – Example slot milling



Run 14 (blue): lower corner velocity
Run 20 (orange): higher corner velocity

Relations between milling parameters and e.g. consumed electricic power can now be explored systematically.

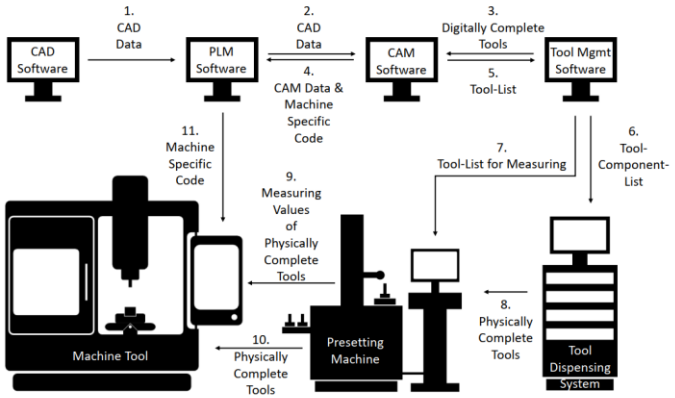


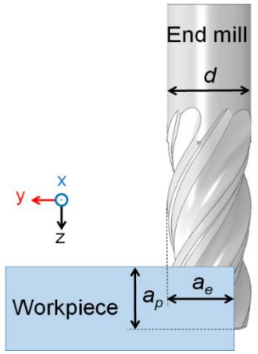
Dataset available on: https://doi.org/10.5281/zenodo.7753180

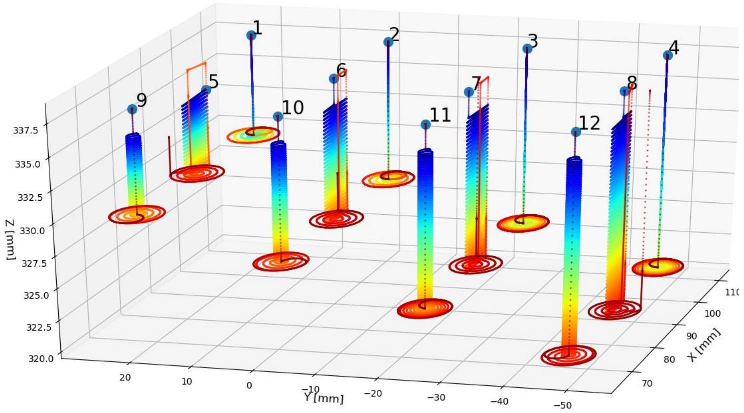
- Motivation & project goal
- Milling experiments
- Dataset
- Outlook

Outlook

Use of FairMillData dataset







Study impact of performed variation in milling operations on quantities such as:

- Energy consumption e.g. channels “Power” (43 - 48)
- Positional accuracy e.g. channels “ActualAxisPosition (86 - 91)
- Actual tool acceleration patterns e.g. channels “ActualAxisPosition (86 - 91)
- Relative tool load e.g. channels “Torque” (7 - 12)
- Time efficiency e.g. channels “VelocityFeedForward“ (37 - 42)