



LARAsuite

full open source automation

mark doerr, stefan maak & stefan born

university greifswald / tu-berlin

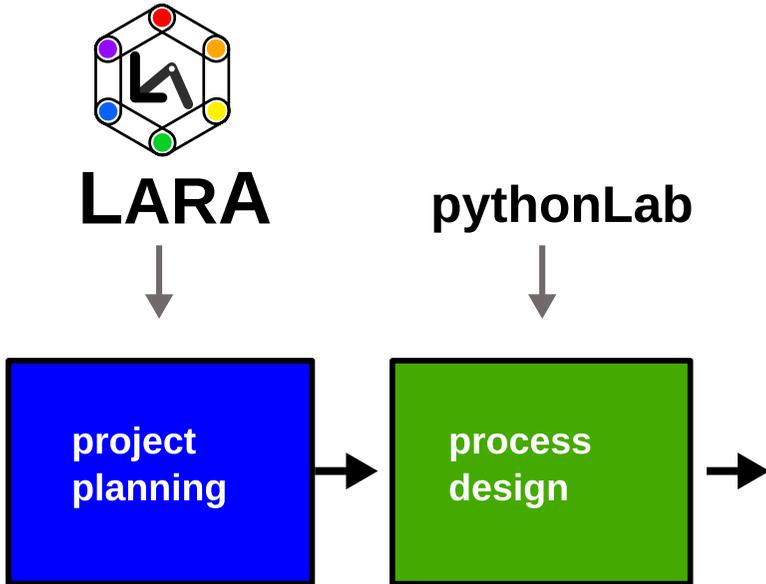
2024-10-11

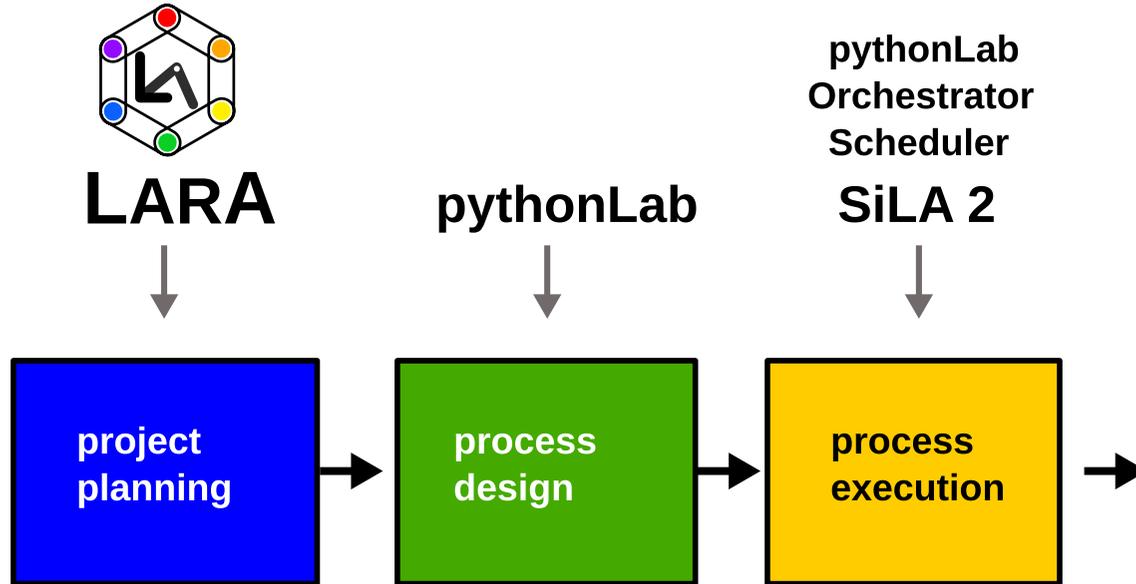
LARA vision: a general purpose
open source Lab Automation
Infrastructure

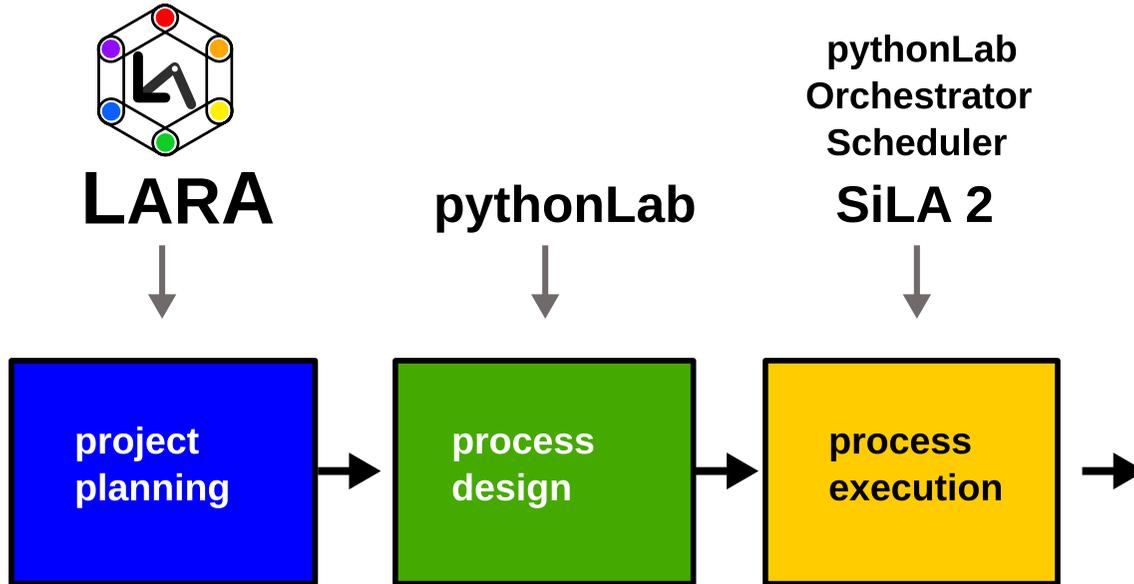


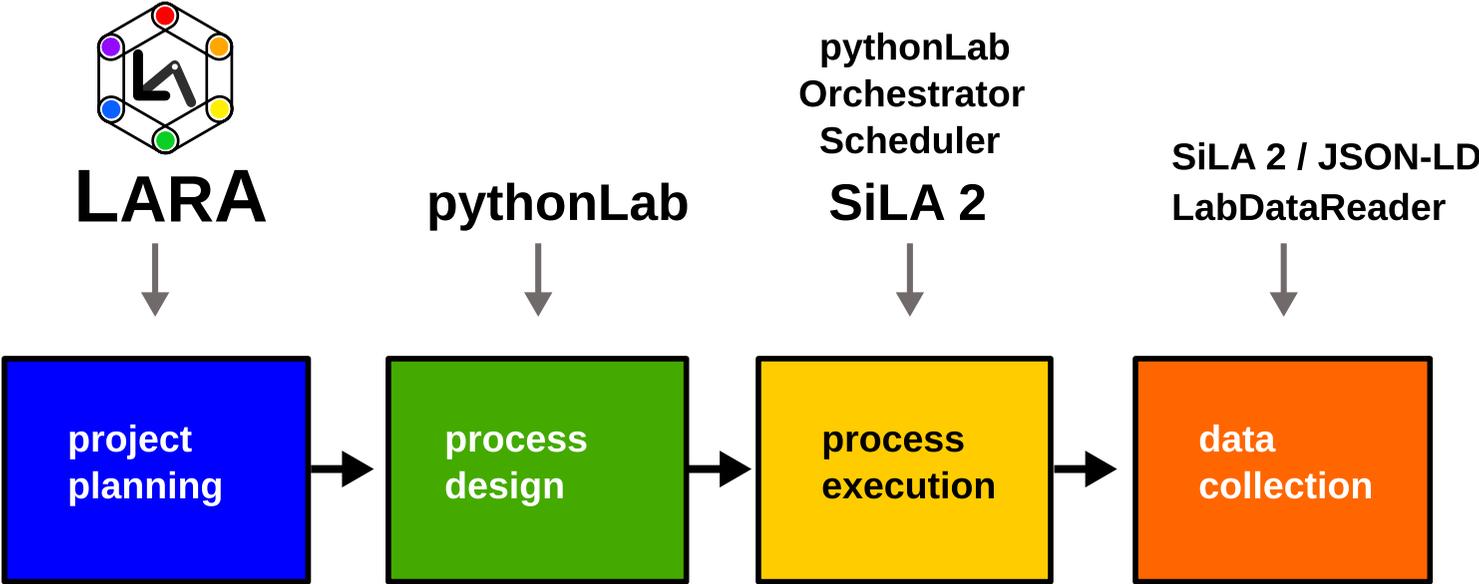
LARA

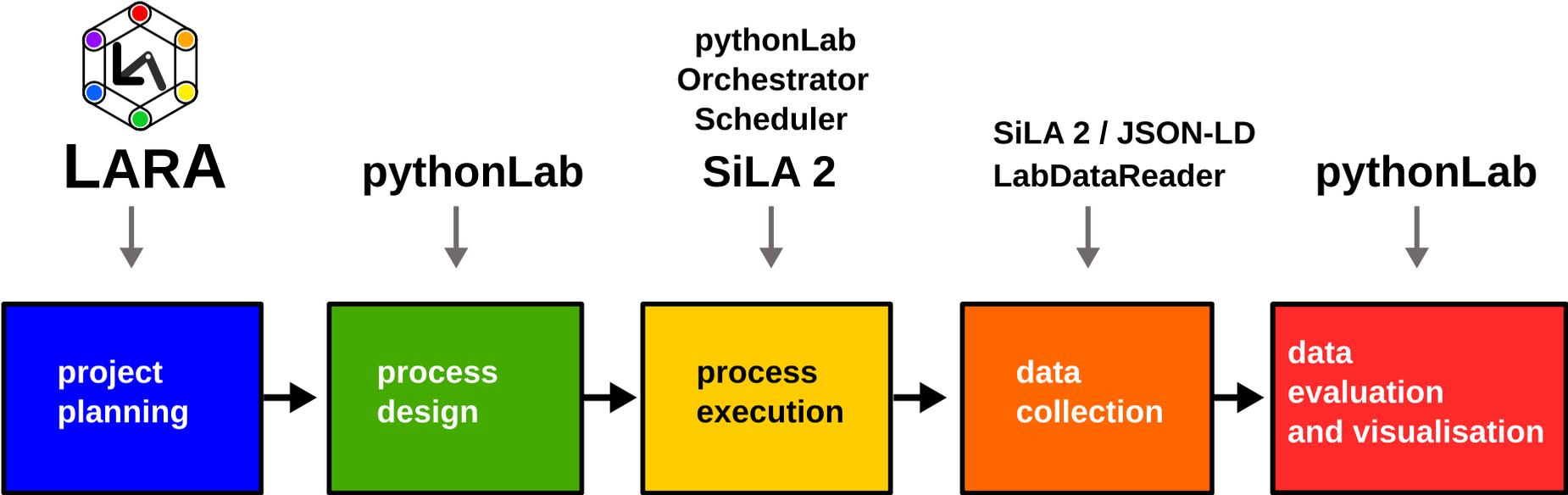


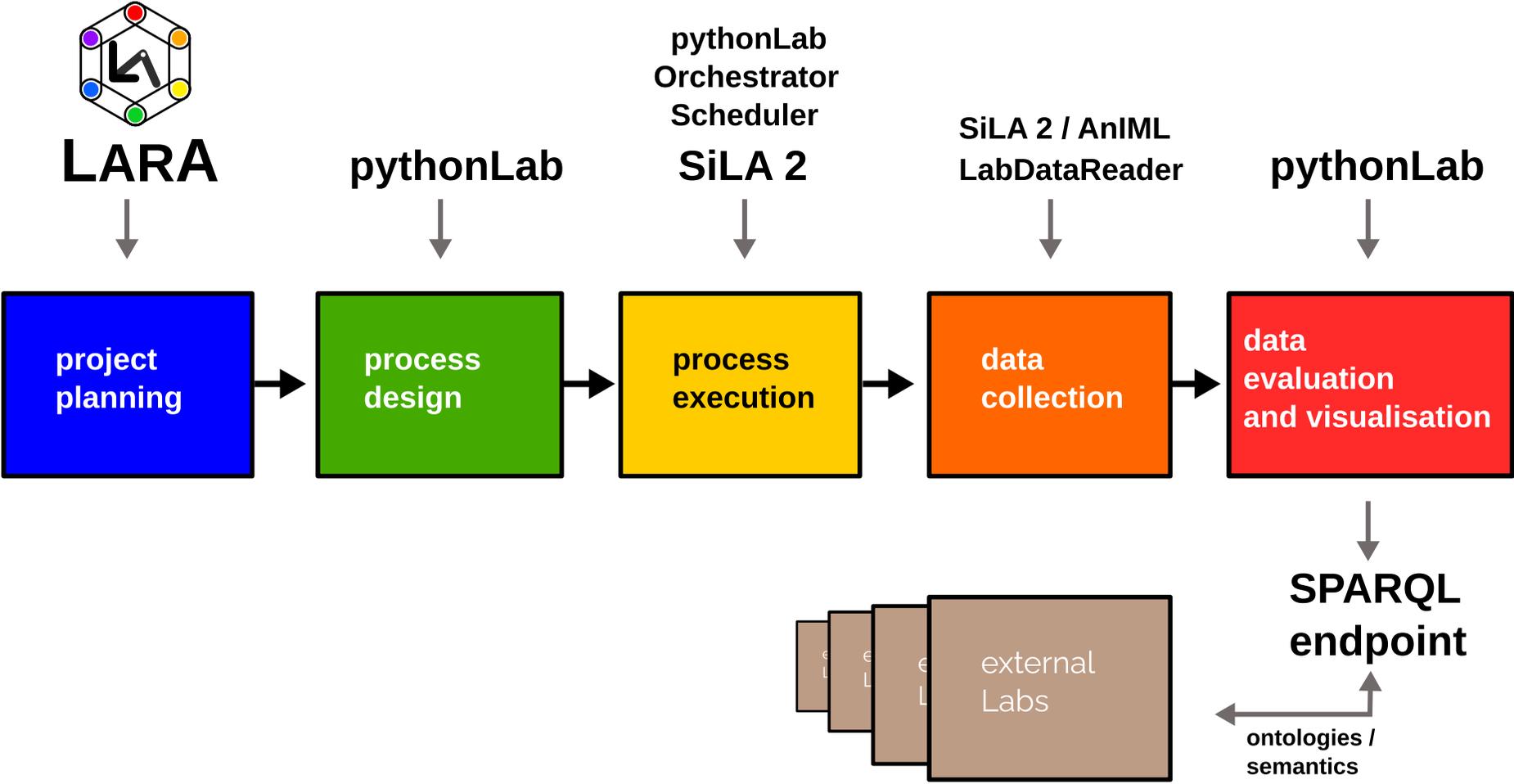








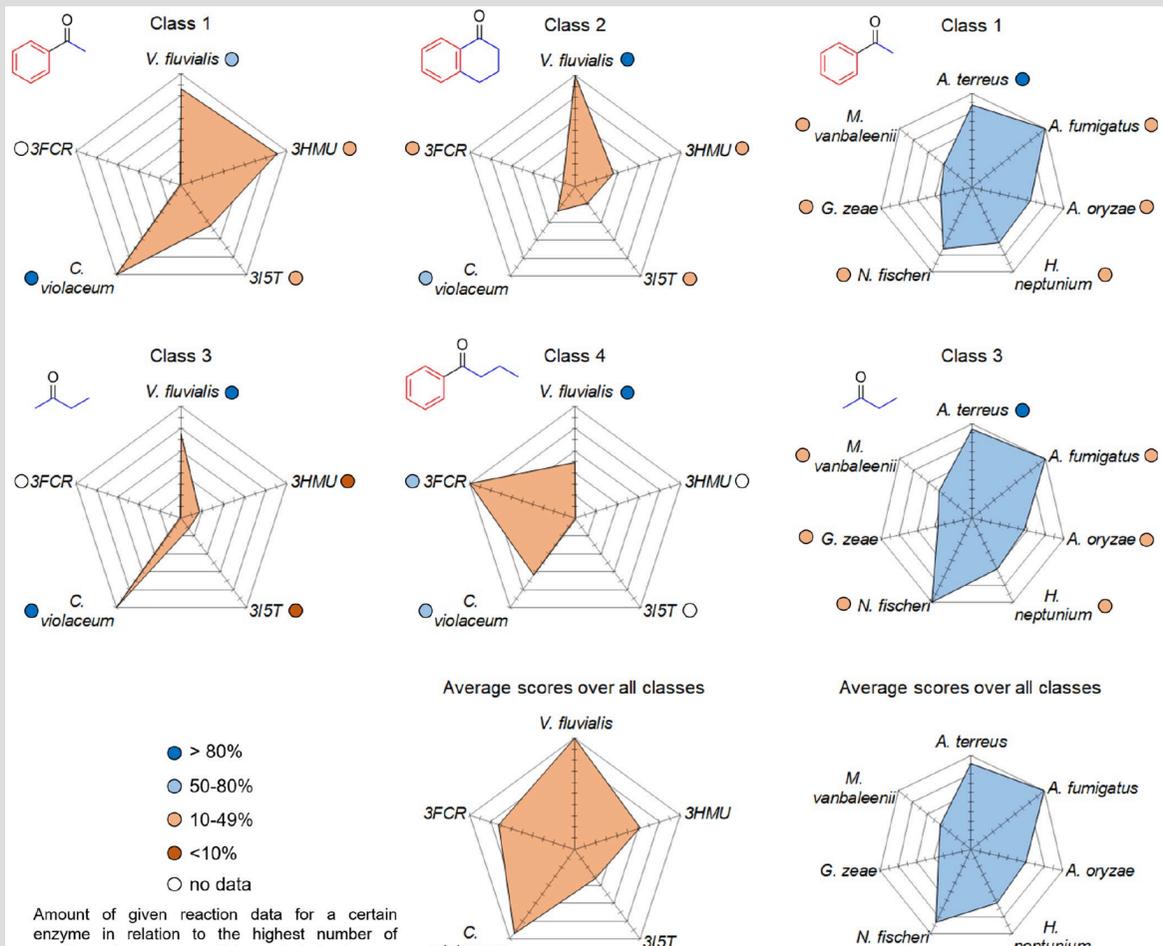




uni greifswald protein screening platform

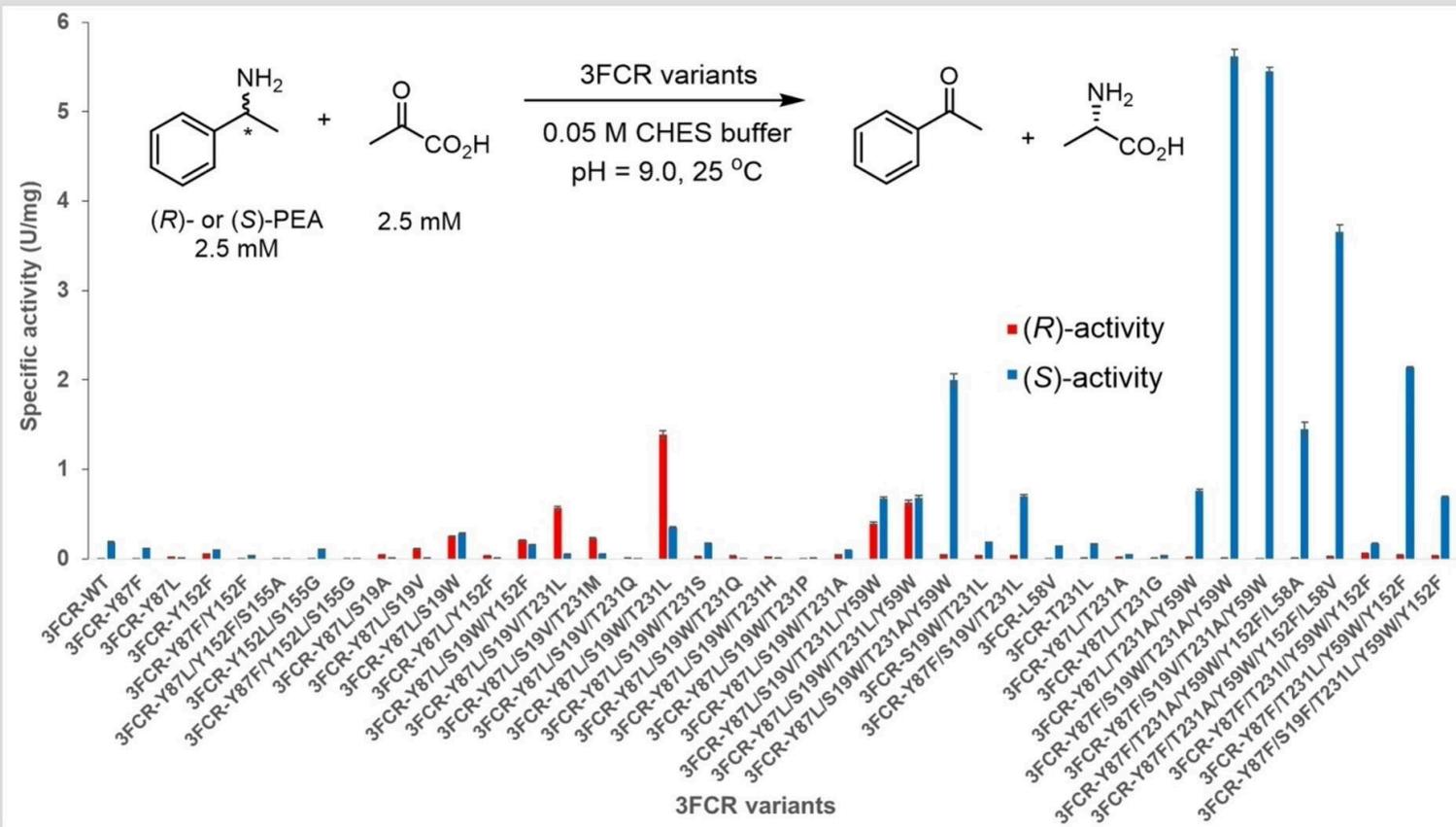


protein engineering



Source: Calvelage, S. et al.; A Systematic Analysis of the Substrate Scope of (S)- and (R)-Selective Amine Transaminases. *Advanced Synthesis & Catalysis* 2017, 359 (23), 4235-4243. <https://doi.org/10.1002/adsc.201701079>.

machine learning



Source: Structure- and Data-Driven Protein Engineering of Transaminases for Improving Activity and Stereoselectivity Yu-Fei Ao et al. Angewandte Chemie 2023. <https://doi.org/10.1002/anie.202301660>

motivation: why we need full
automation of metadata
acquisition and semantic data ?

- meaning of the data is well defined

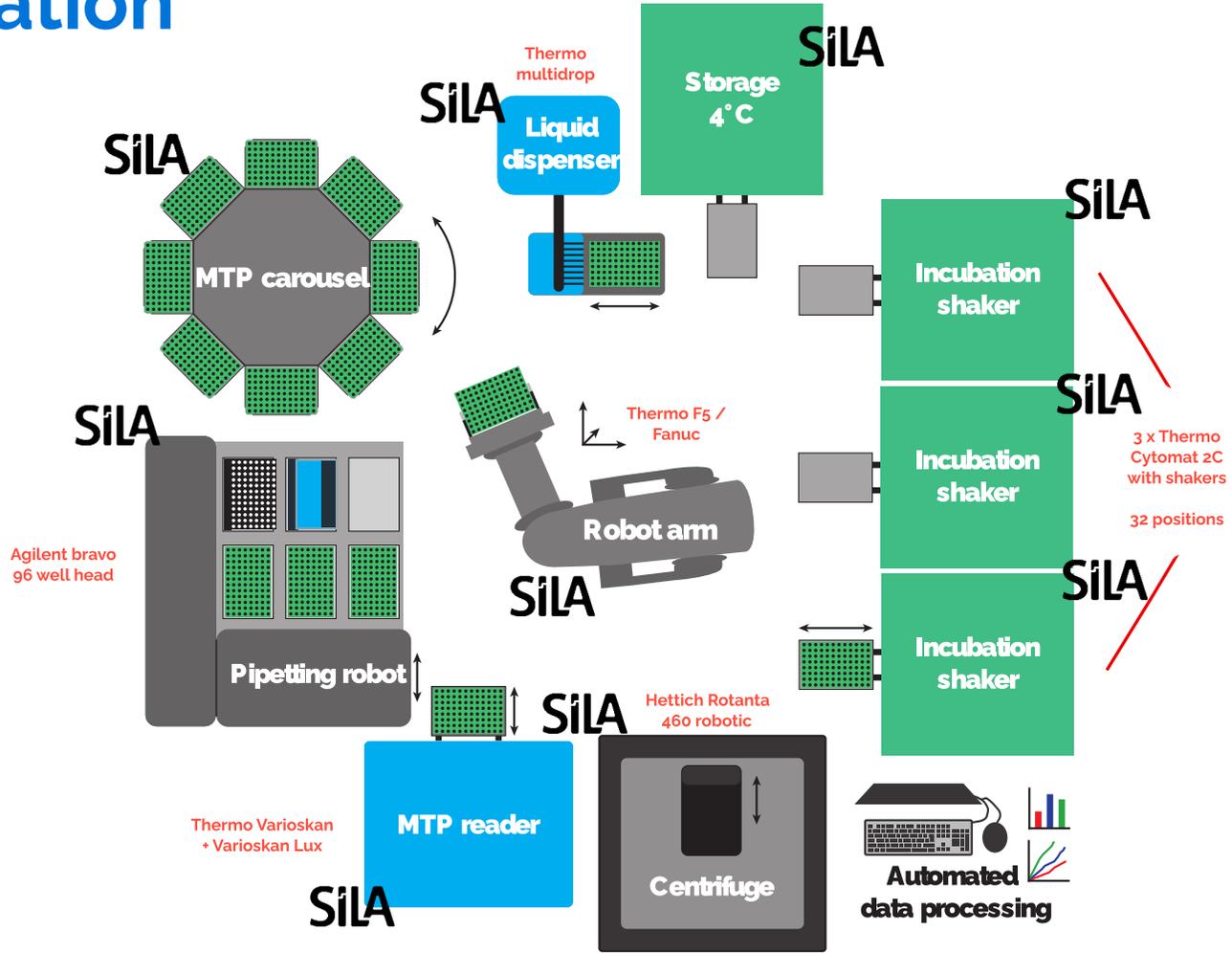
- meaning of the data is well defined
- machine readable

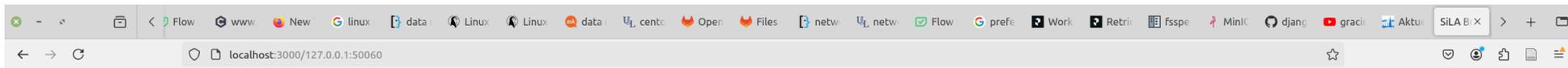
- meaning of the data is well defined
- machine readable
- machine actionable / “understandable”

- meaning of the data is well defined
- machine readable
- machine actionable / “understandable”
- machine reasoning (inference) possible (reasoners)

https://en.wikipedia.org/wiki/FAIR_data

standardisation of communication





Servers > 127.0.0.1:50060



gitlab.com/unitelabs

SiLA Service org.silastandard/core/SiLAService/v1

Device Info Provider de.unigreifswald.biochemie/device/DeviceInfoProvider/v1

Light Intensity Controller de.unigreifswald/instruments/LightIntensityController/v1

Logging Service de.unigreifswald/infrastructure/LoggingService/v0

Simulation Controller org.silastandard/core/SimulationController/v1

Temperature Controller de.unigreifswald/instruments/TemperatureController/v1

This is a simple example of a generic Feature for controlling and retrieving the temperature. A new target temperature can be set anytime with the 'Control Temperature' Command. The temperature range has been limited to prevent major damages of a device. In case the first target temperature has not been reached, a ControlInterrupted Error should be thrown.

Current Temperature	🔍 ^
Target Temperature	🔍 ^
Current Temperature JSONLD	🔍 ^
Target Temperature JSONLD	🔍 ^
Metadata	🔍 ^
Control Temperature	🔍 ^
Defined Exception Error	^

Oct. 29th | Prefect - dbt Showcase: Scaling Data Workflows

PRODUCT SOLUTIONS RESOURCES PRICING

16,030 SIGNUP / LOGIN BOOK A DEMO

```
flow.py
1 from prefect import flow, task
2
3
4 @task(log_prints=True)
5 def say_hello(name: str):
6     print(f"Hello {name}!")
7
8
9 @flow
10 def hello_universe(names: list[str]):
11     for name in names:
12         say_hello(name)
13
14
15 if __name__ == "__main__":
16     # create your first deployment to automate your flow
17     hello_universe.serve(name="your-first-deployment")
```

Acme Co.

Dashboard

Flow Runs

Artifacts

Event Feed

Flows

Blocks

Variables

Work Pools

Task-Run Concurrency

Automations

FLOW RUNS

72

TASK RUNS

1.2k

EVENTS

7.4k

WORK POOLS

Pool Name	Polled	Avg. Latency	Work Queue	Completes
azure-worker-b	8s	2.2s	3	89%

demo : new LARAsuite
orchestrator

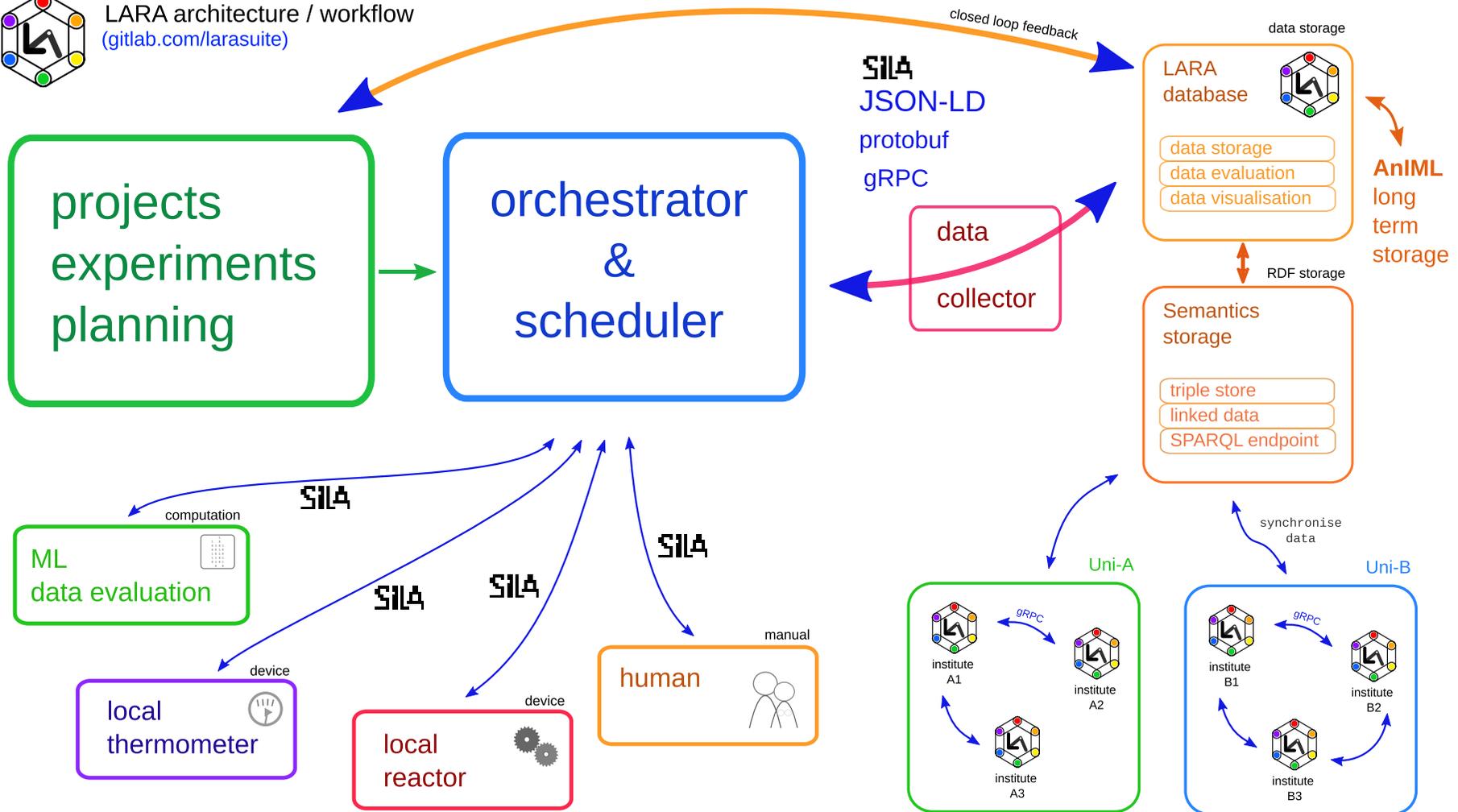
<https://gitlab.com/larasuite>

The screenshot displays the Prefect interface for a workflow run named 'papaya-dingo'. At the top, it indicates the run is 'Completed' on 2024/10/08 at 08:58:22 PM, with 7s duration and 12 task runs. The flow is identified as 'science-robotic-process'. A Gantt chart below shows the execution timeline of tasks: 'move-2d2' (20:58:22-20:58:23), 'move-97c' (20:58:23-20:58:24), 'move-71b' (20:58:26-20:58:27), and 'move-3f7' (20:58:28-20:58:29). Other tasks include 'get_url-633', 'get_url-7e2', 'get_url-978', 'add_reagent-f46', 'centrifuge-d90', 'get_url-d7f', 'incubate-fa3', and 'get_url-915'. Below the Gantt chart, a 'Logs' tab is active, showing a list of log entries with timestamps and messages such as 'Created task run 'move-2d2' for task 'move'', 'Moving PCR plate to pipetting_robot', and 'Finished in state Completed()'. The interface also includes a sidebar with navigation options like Dashboard, Runs, Flows, and Deployments, and a bottom navigation bar with 'Ready to scale? Upgrade', 'Join the Community', and 'Settings'.

LARAsuite



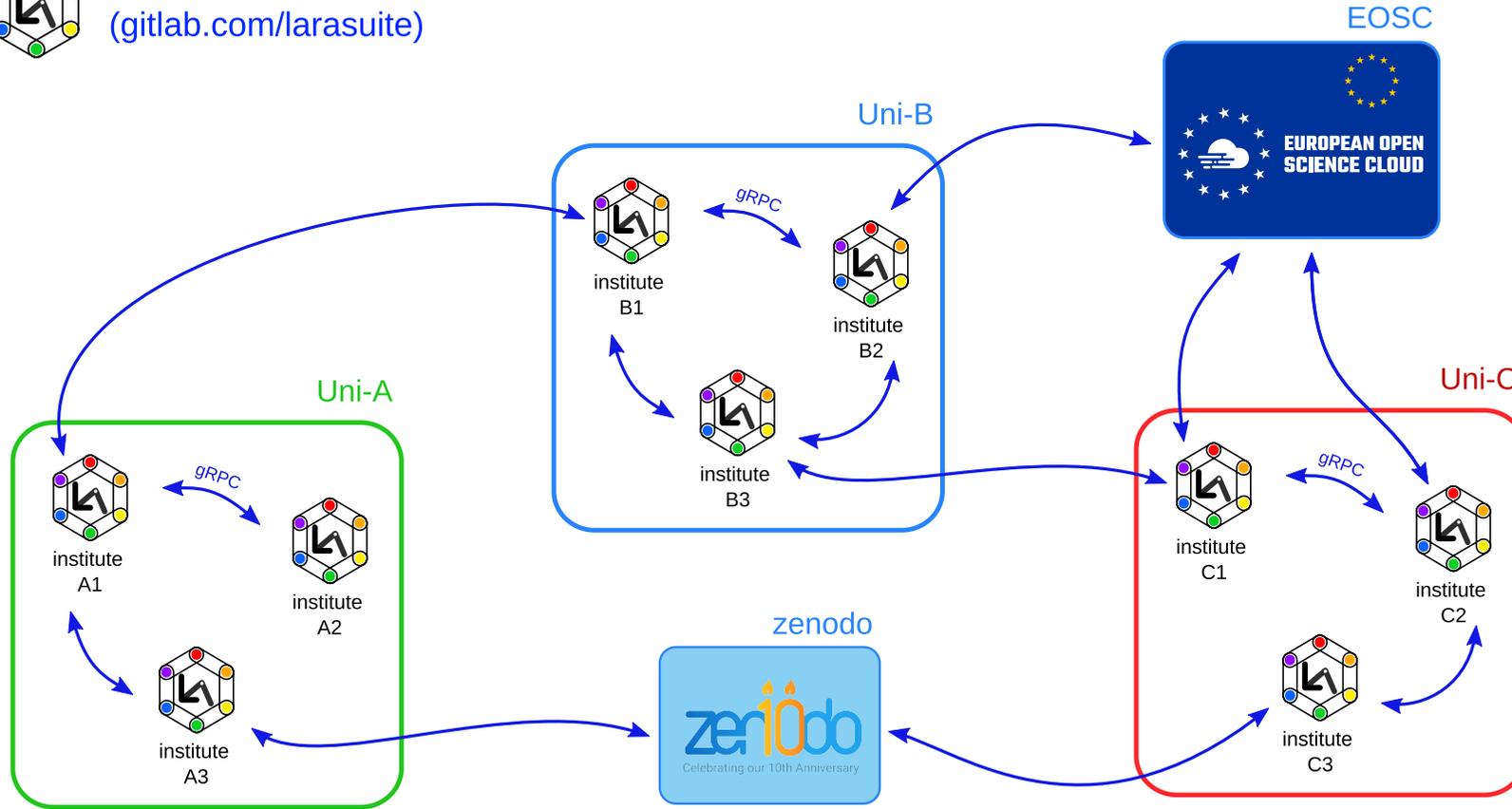
LARA architecture / workflow
(gitlab.com/larasuite)



LARA connectivity



LARA network
(gitlab.com/larasuite)

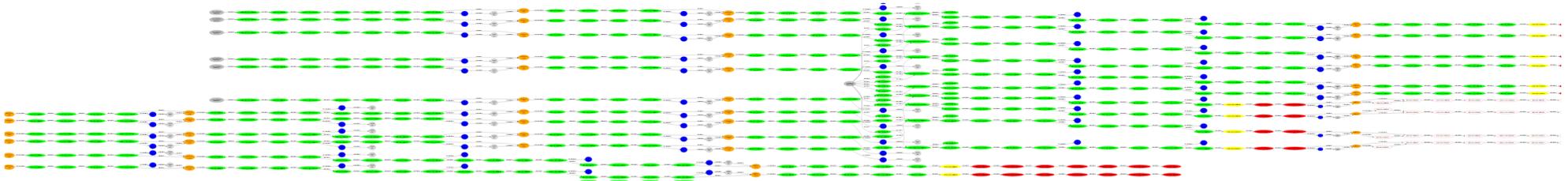


```
incubation_duration = 120
cont1 = self.containers[0]
# move to incubator
self.robot_arm.move(cont1, source_loc=cont1.pos, target_loc=self.incubator)
# incubate
self.incubator.incubate(cont1, duration=incubation_duration, temperature=310)
# move to reader
self.robot_arm.move(cont1, source_loc=cont1.pos, target_loc=self.reader)
# make absorbance measurement
self.reader.single_read(cont1, wavelengths=[600, 660], temperature=305)
# move back to hotel
self.robot_arm.move(cont1, source_loc=cont1.pos, target_loc=self.hotel)
```

<https://gitlab.com/opensourcelab/pythonlab>



XFG
Emergency STOP



labDataReader

- generic reader of proprietary data (e.g. HPLC, plate readers)
- primary output : pandas data frame and *JSON-LD* (metadata)

<https://gitlab.com/opensourcelab/scientificdata/labDataReader>

SciDat

- packing tabular data / data frames into *parquet* files, including *JSON-LD* metadata

<https://gitlab.com/opensourcelab/scientificdata/scidat>

vocabulary and ontology developments

developments

- vocabularies - controlled vocabularies
(example: github.com/nfdi4cat/voc4cat)

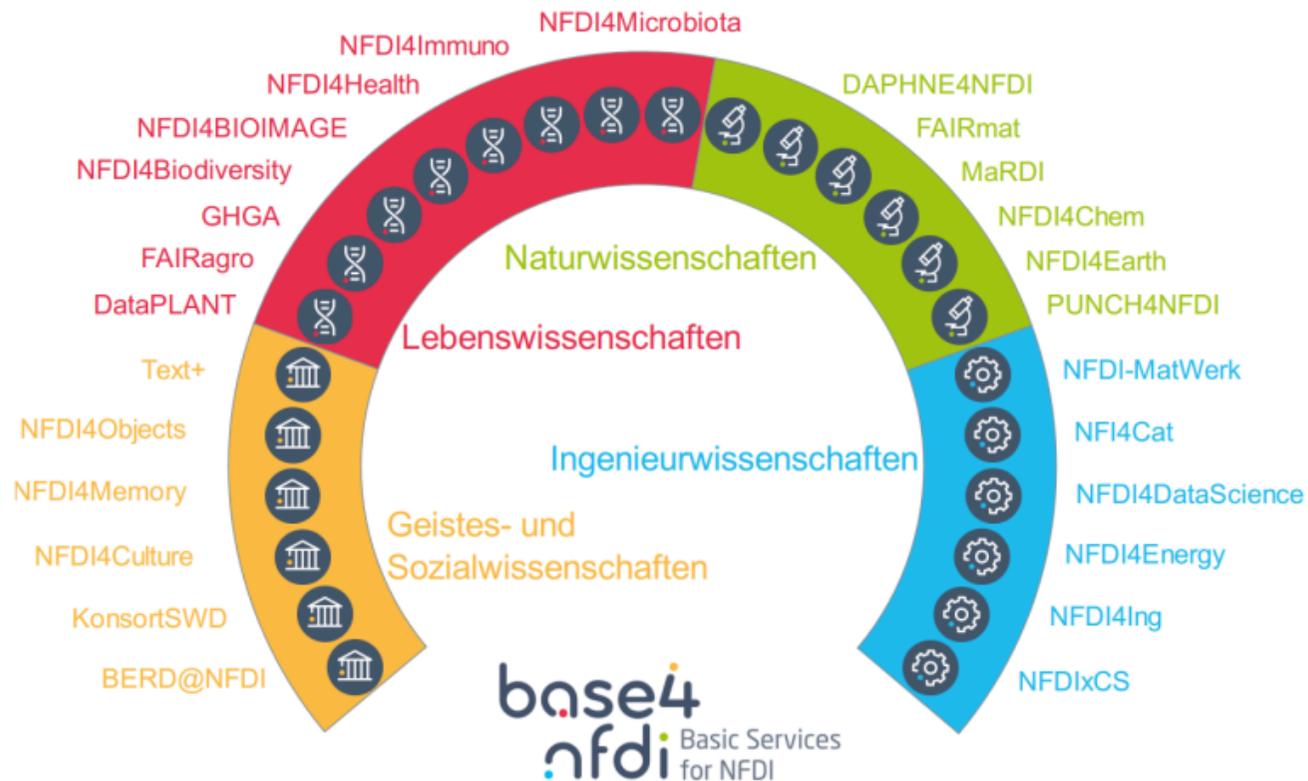
developments

- vocabularies - controlled vocabularies
(example: github.com/nfdi4cat/voc4cat)
- taxonomies - hierarchical vocabularies

developments

- vocabularies - controlled vocabularies
(example: github.com/nfdi4cat/voc4cat)
- taxonomies - hierarchical vocabularies
- ontologies - formal representation of knowledge
with reasoning capabilities

vocabulary and ontology developments



<https://forschungsdaten-thueringen.de/nfdi-en.html>

and now for something
completely different

**GBA software / data
workgroup**

collection of atomic lab operations

- common workflow language ontology with a mapping infrastructure to translate this into execution workflows
- collection of competency questions for the ontology developments

<https://gitlab.com/opensourcelab/scientificdata/ontologies/openscienceontology/experimental-workflows>

semantically enriched data

- conversion of proprietary data to *JSON-LD* format

<https://gitlab.com/opensourcecelab/scientificdata/labdatareader>

<https://gitlab.com/opensourcecelab/scientificdata/scidat>

open source workflow orchestration environment

- for the members, who do not have software development skills

<https://gitlab.com/opensourcelab>

- if you have more experiance, more ideas, better software development skills and infrastructures, please support us ! (or take over ;)

acknowledgements

- Stefan Maak
- Uwe Bornscheuer's group (Univ. Greifswald)

project partners

- Stefan Born (TU Berlin)
- Peter Neubauer's group (TU Berlin)
- Johannes Kabisch's group and associates (Uni Trondheim)
- Egon Heuson (Uni Lille)
- Lukas Bromig and Julian Willand (unitelabs)

KIWI-biolab team, NFDI4Cat teams, SiLA team & AnIML team

This work was supported by the German Federal Ministry of Education and Research through the Program "International Future Labs for Artificial Intelligence" (Grant number 01DD20002A) as well as the NFDI4Cat grant

We are grateful to the Deutsche Forschungsgemeinschaft (DFG, INST 292/118-1 FUGG) and the federal state Mecklenburg-Vorpommern for financing the robotic platform.

presentations

