



LARAsuite

full open source automation - tools for NFID4Cat

mark doerr, stefan maak & stefan born

university greifswald / tu-berlin

2024-10-23

LARA vision: a general purpose
open source Research Data
Management / Automation
Infrastructure

process planning

LARA

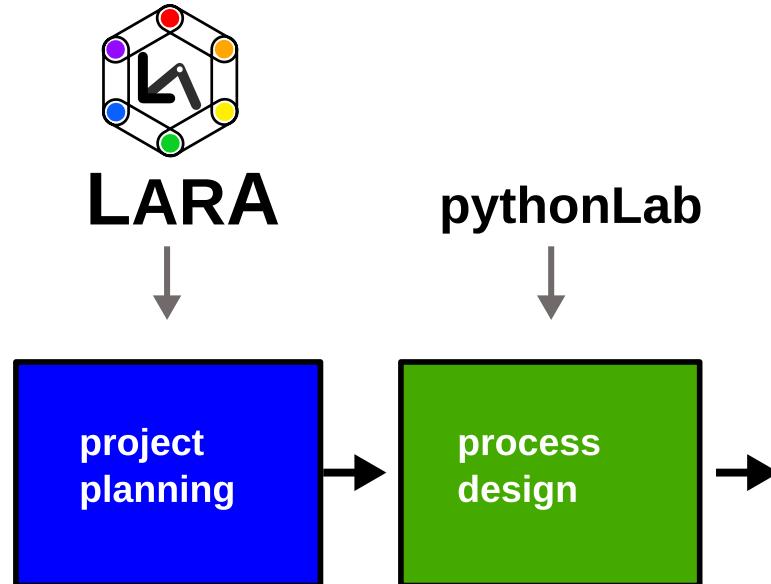


LARA



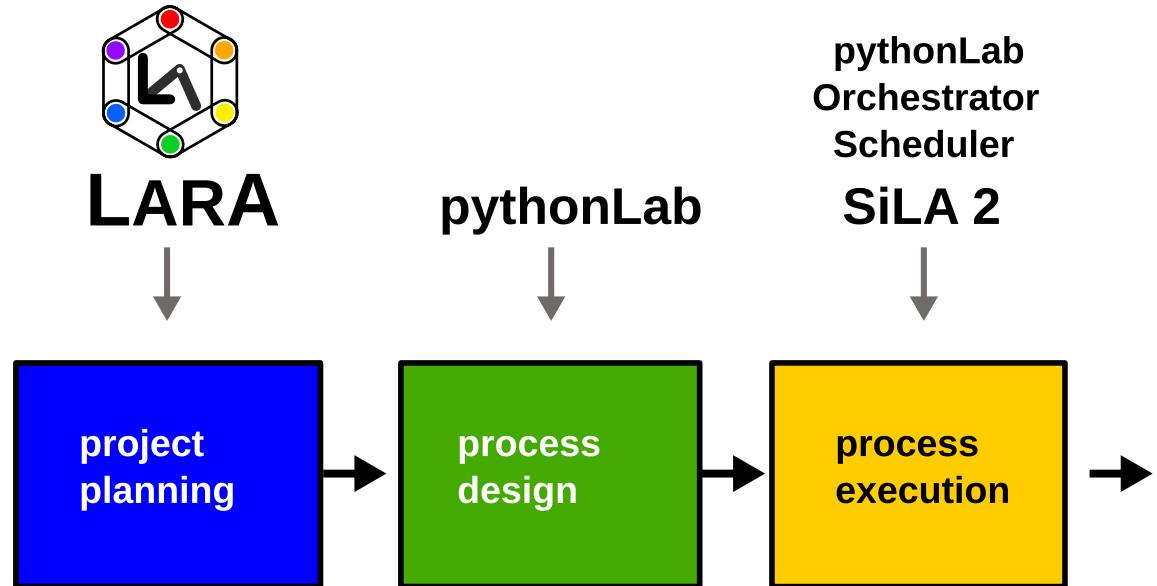
process design

LARA



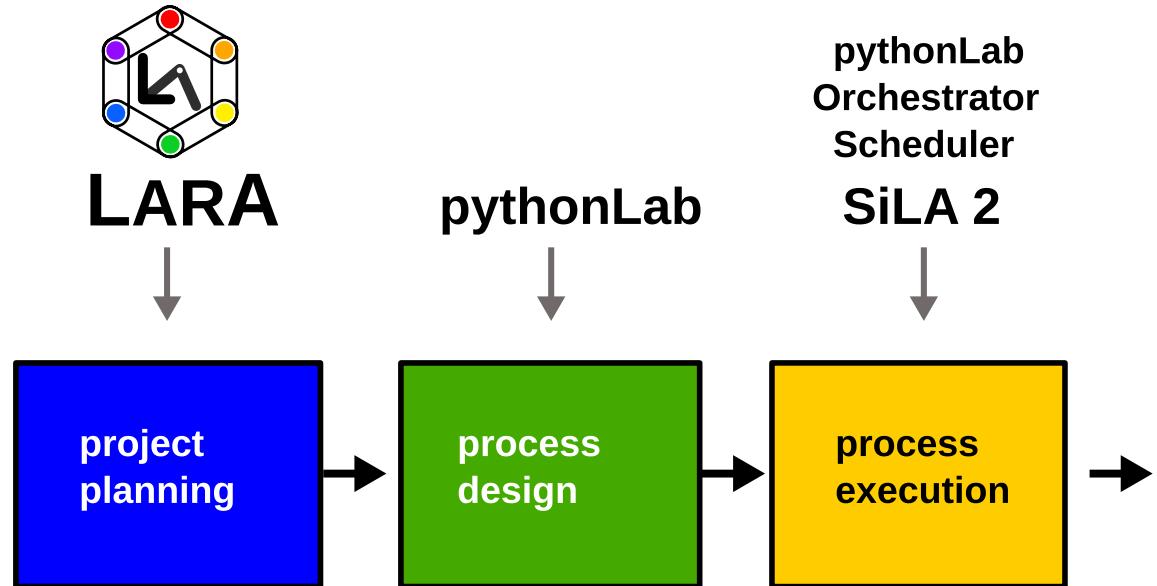
process execution

LARA



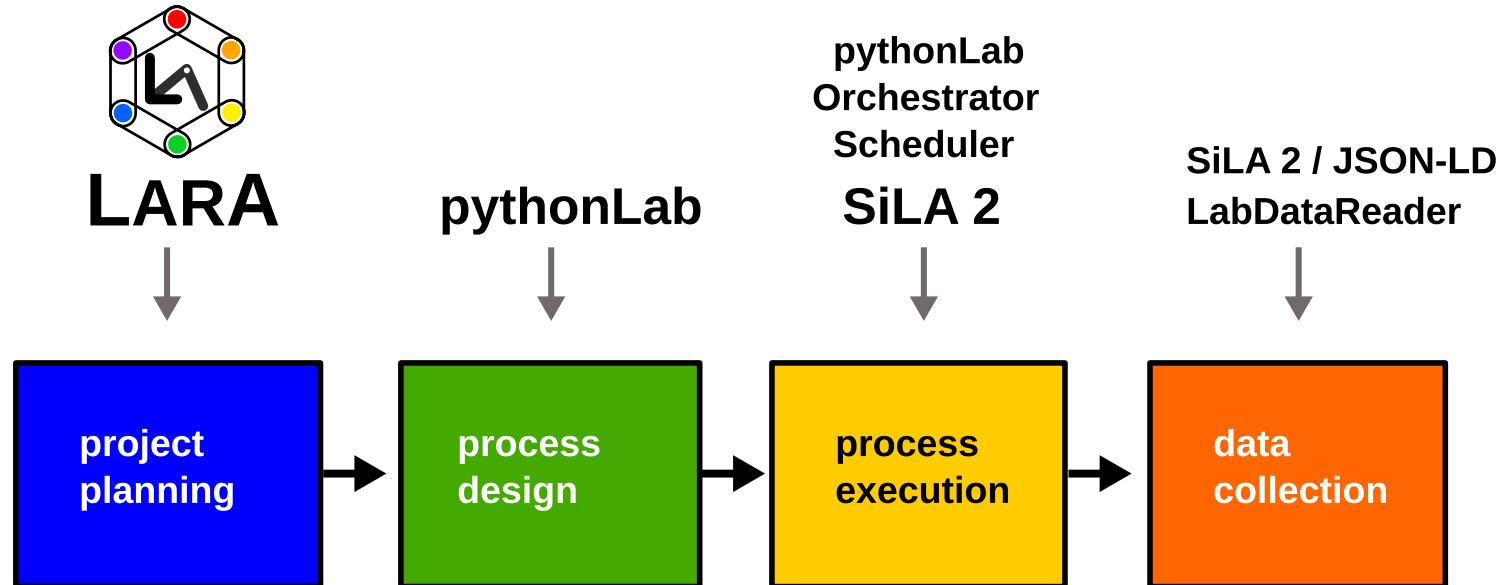
process execution

LARA



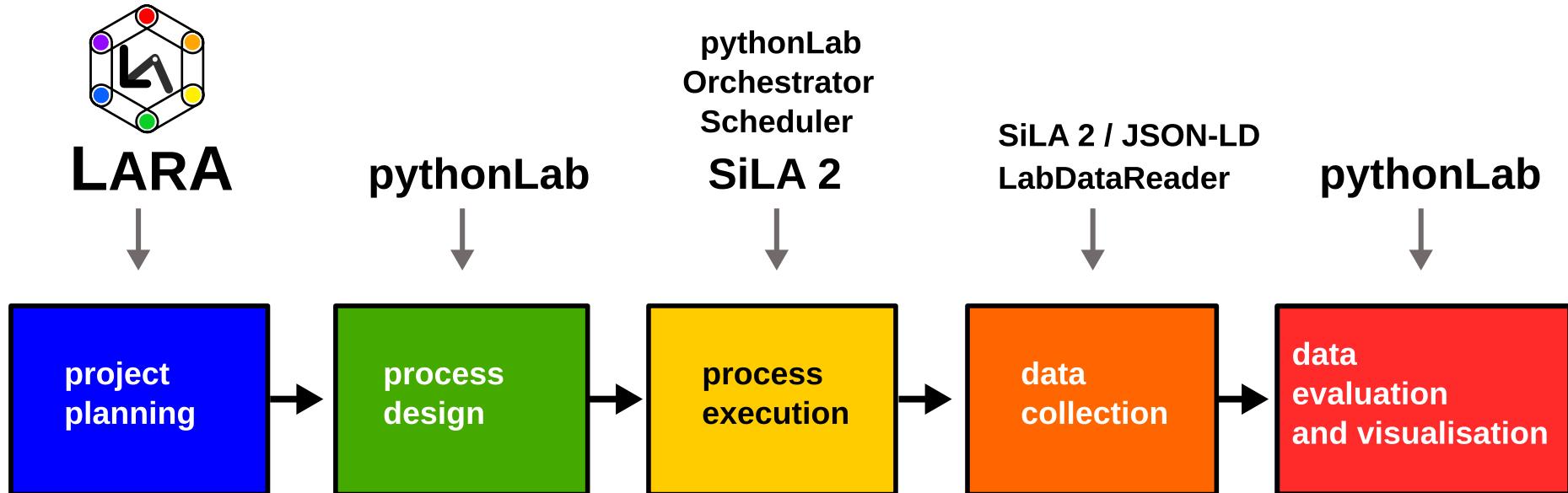
data collection

LARA



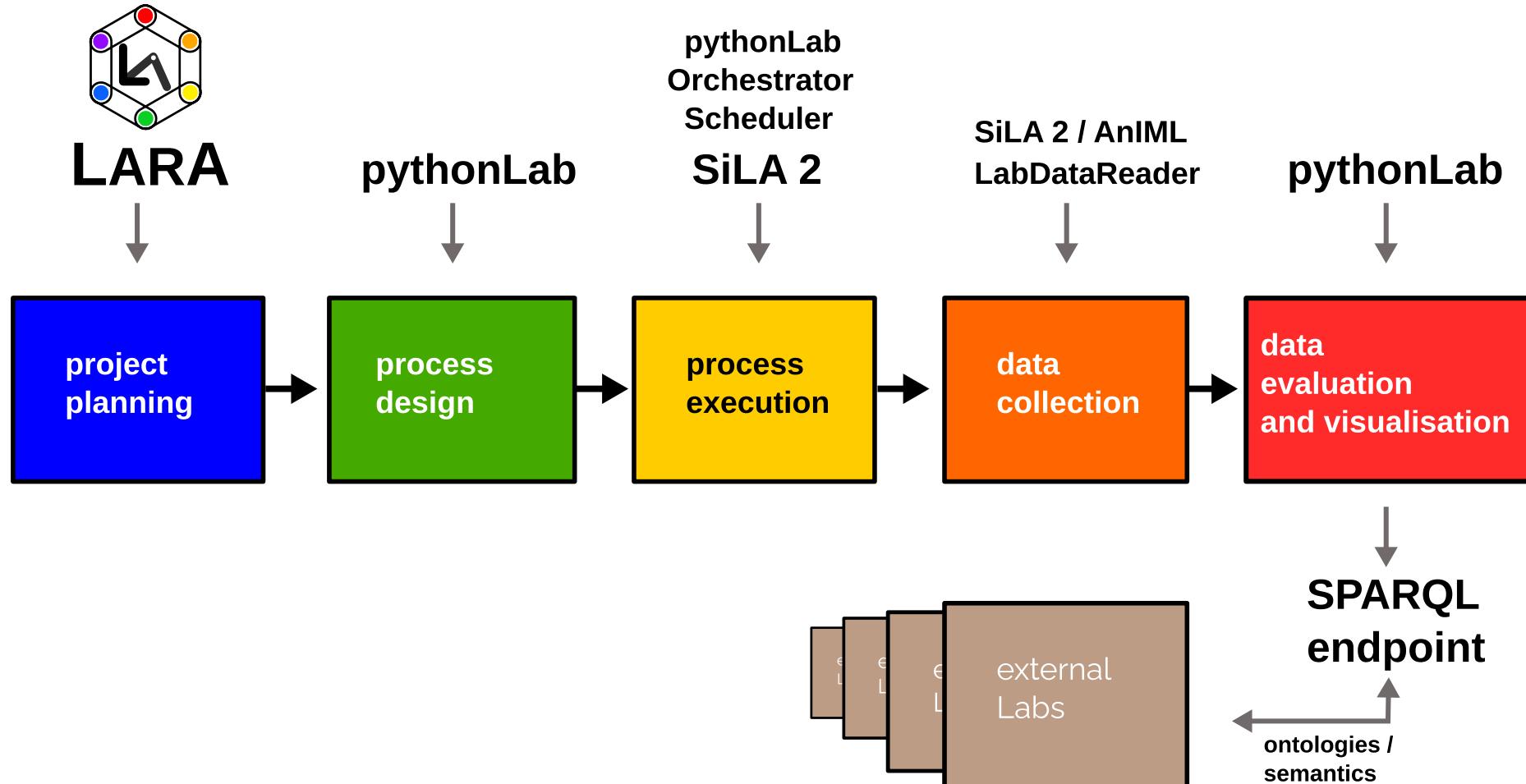
data evaluation / visualisation

LARA



querying of data

LARA

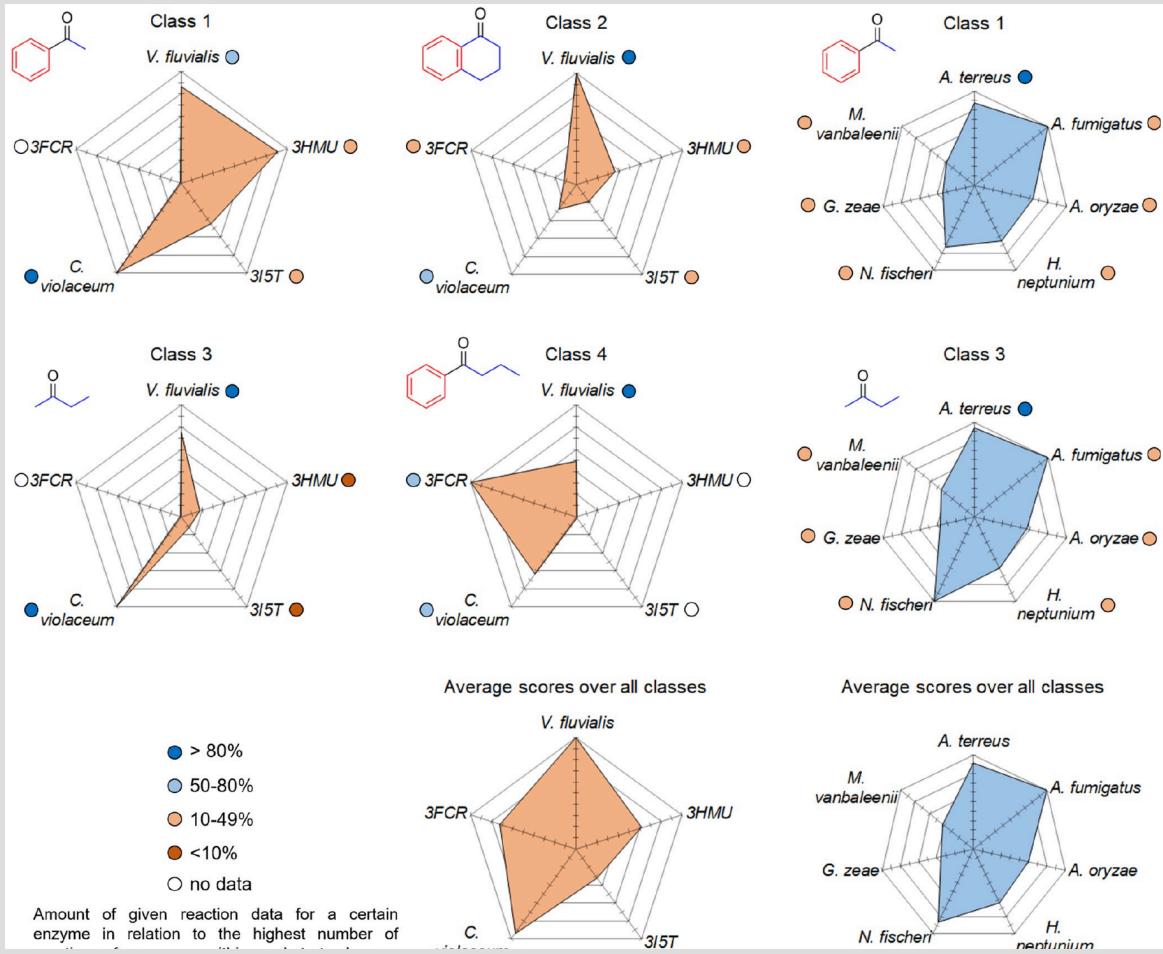


uni greifswald protein screening platform

LARA

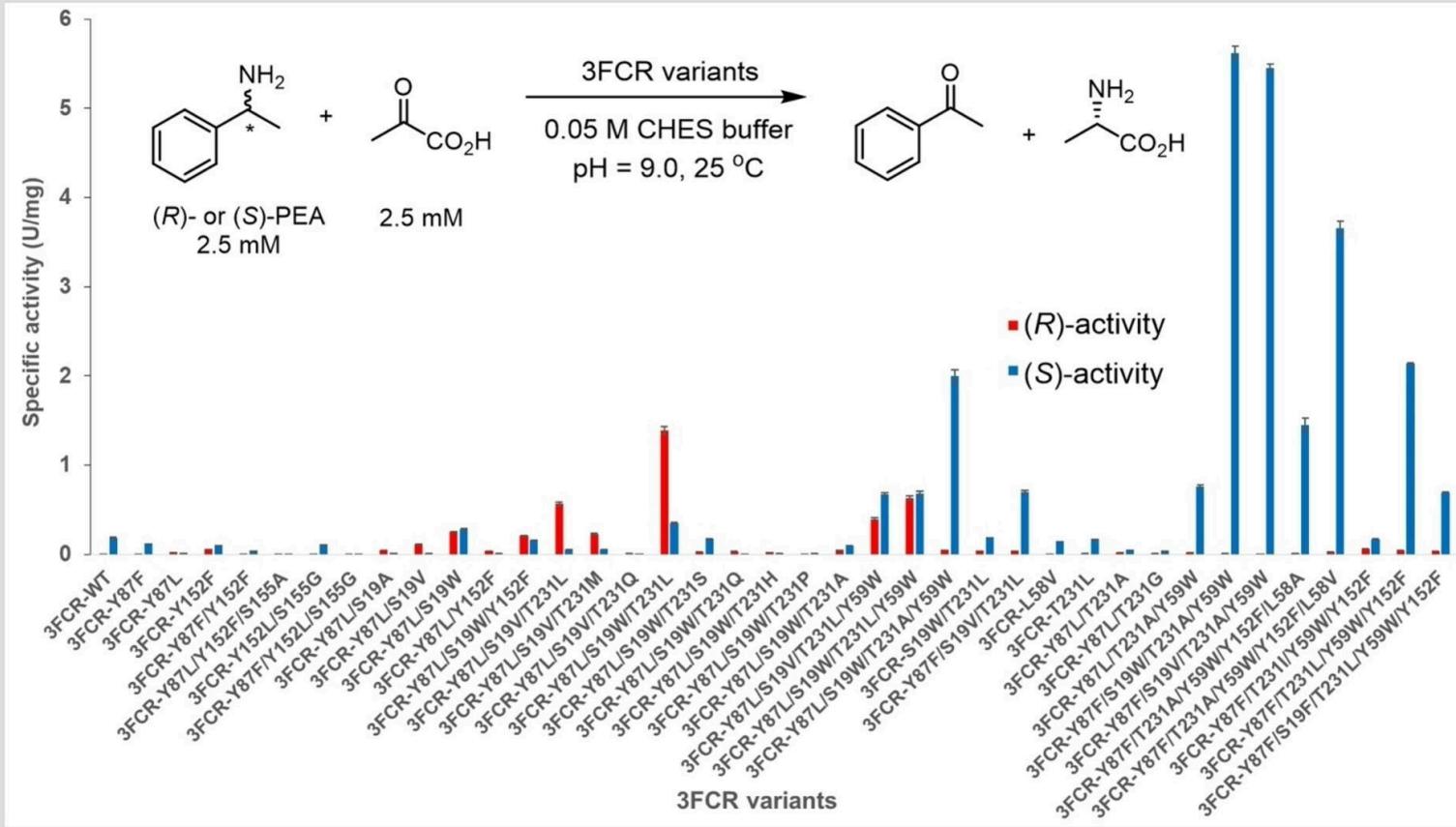


protein engineering



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
aquisition and semantic data ?

- meaning of the data is well defined

- meaning of the data is well defined
- machine readable

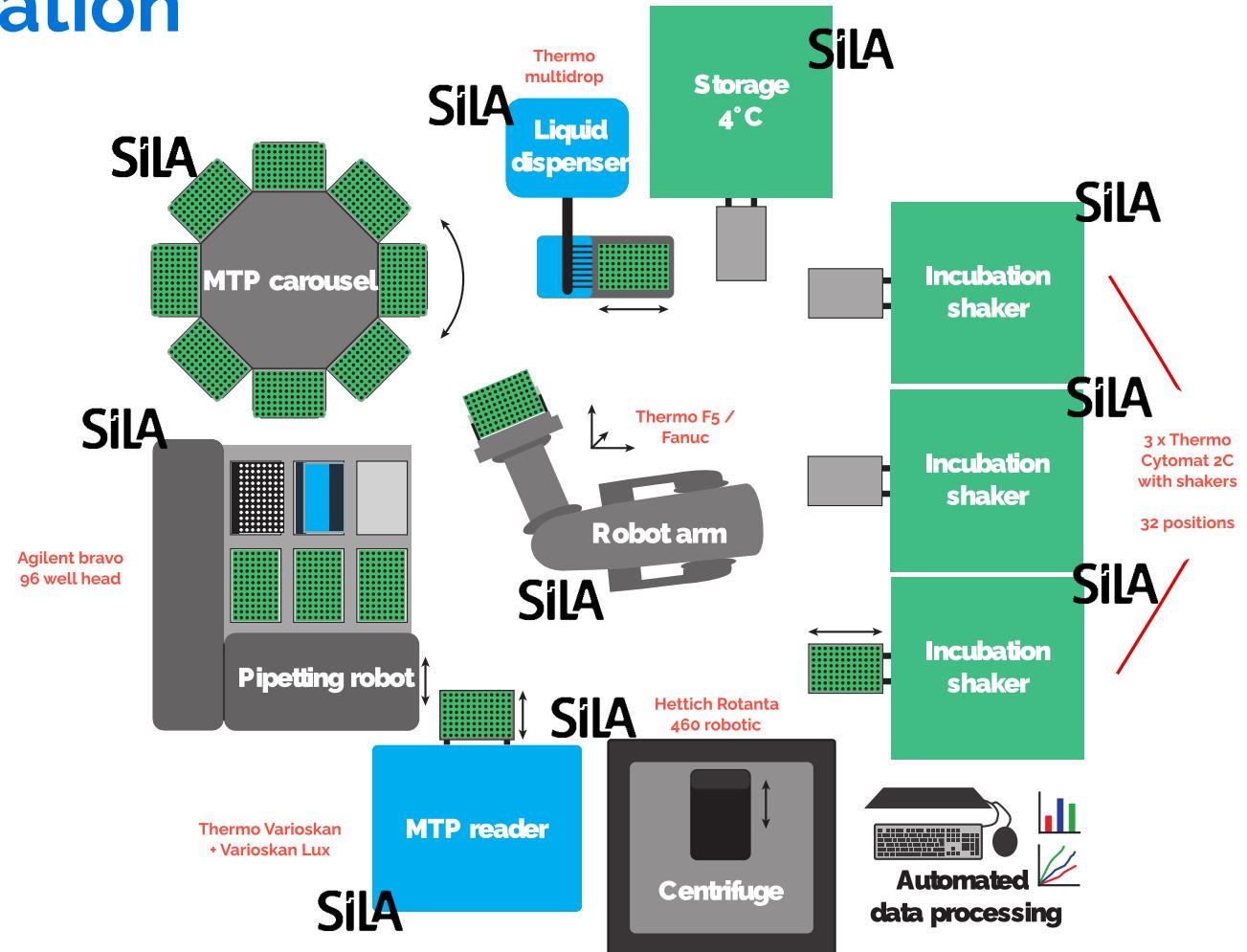
- meaning of the data is well defined
- machine readable
- machine actionable / “understandable”

- 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

LARA



generic SiLA browser (unitelabs)

LARA

The screenshot shows a browser window displaying the SiLA service catalog for a device at `localhost:3000/127.0.0.1:50060`. The page lists various SiLA services and their corresponding provider details:

- 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

A detailed description for the Temperature Controller service is provided:

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.

The page also displays several control buttons for the Temperature Controller:

- Current Temperature
- Target Temperature
- Current Temperature JSONLD
- Target Temperature JSONLD
- Metadata
- Control Temperature (highlighted with an orange border)
- Defined Execution Error

prefect (www.prefect.io)

LARA

The screenshot displays the Prefect website interface. At the top, there is a navigation bar with links for PRODUCT, SOLUTIONS, RESOURCES, and PRICING. On the right side of the header, there is a user icon, a star rating of 16,030, a SIGNUP / LOGIN button, and a BOOK A DEMO button.

The main content area features a code editor window titled "flow.py" containing Python code:

```
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")
```

Below the code editor is a dashboard section with various metrics and charts. The dashboard includes a sidebar with navigation links such as Dashboard, Flow Runs, Artifacts, Event Feed, Flows, Blocks, Variables, Work Pools, Task Run Concurrency, and Automations. The main dashboard area shows the following data:

- FLOW RUNS:** 72
- TASK RUNS:** 1.2k (Up 6.8%)
- EVENTS:** 7.4k (Up 8.2%)
- WORK POOLS:** azure-push (Pooled: 8s, Avg. Latency: 2.2s, Work Queues: 3, Completes: 89%+), azure-worker-b (Pooled: 1s, Avg. Latency: 0.5s, Work Queues: 1, Completes: 100%)

demo : new LARAsuite
orchestrator

<https://gitlab.com/larasuite>

prefect 3 robotic workflow

LARA

Runs / papaya-dingo

Completed 2024/10/08 08:58:22 PM 7s 12 Task runs

Flow science-robotic-process

Dashboard

- Runs
- Flows
- Deployments
- Work Pools
- Blocks
- Variables
- Automations
- Event Feed
- Notifications
- Concurrency

Logs Task Runs Subflow Runs Artifacts Details Parameters Job Variables

Level: all Oldest to newest

Oct 8th, 2024

INFO Created task run 'move-2d2' for task 'move'

INFO Moving PCR plate to pipetting_robot

INFO Created task run 'get_url-633' for task 'get_url'

INFO Finished in state Completed()

INFO PrefectHQ/prefect repository statistics 🤖:

INFO Stars 🌟 : 16009

INFO Forks ⌂ : 1568

INFO Finished in state Completed()

INFO Created task run 'add_reagent-f46' for task 'add_reagent'

INFO Adding 10µl of Teal

Ready to scale? Upgrade

Join the Community

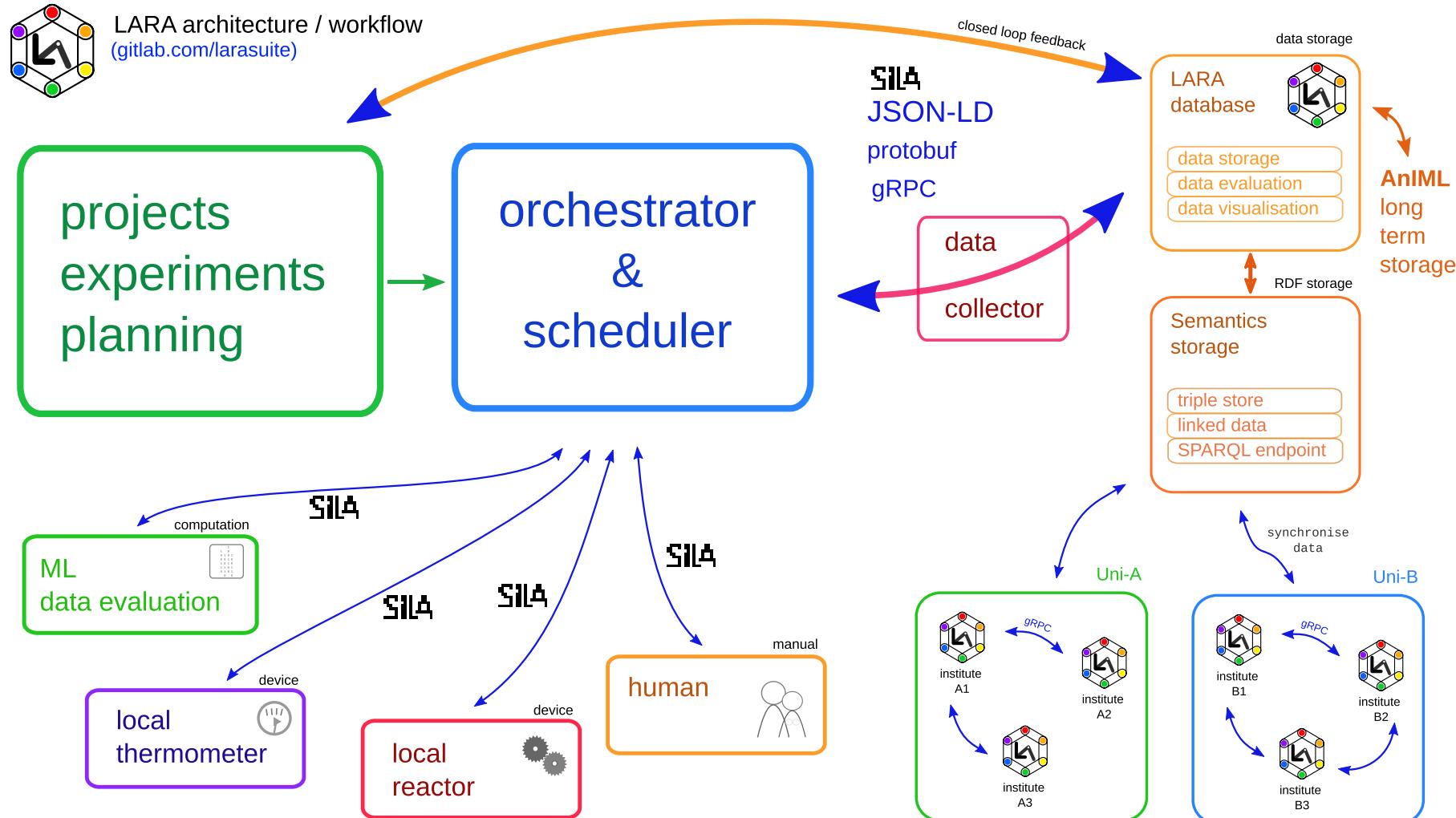
Settings

08:58:22 PM move-2d2 prefect.task_runs
08:58:22 PM move-2d2 prefect.task_runs
08:58:22 PM get_url-633 prefect.task_runs
08:58:23 PM move-2d2 prefect.task_runs
08:58:23 PM add_reagent-f46 prefect.task_runs
08:58:23 PM add_reagent-f46 prefect.task_runs

LARAsuite

LARAsuite overview

LARA

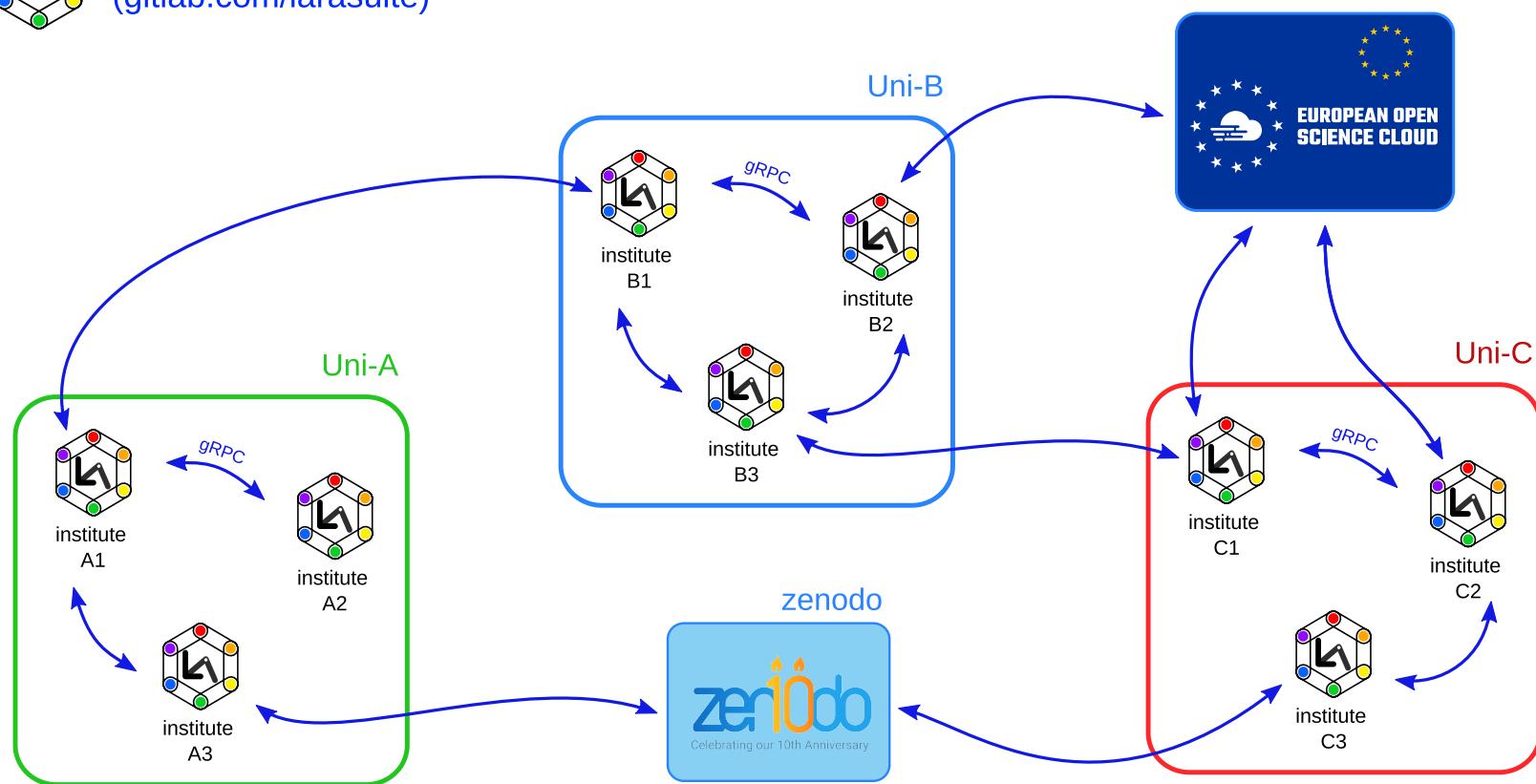


LARA connectivity

LARA



LARA network
(gitlab.com/larasuite)

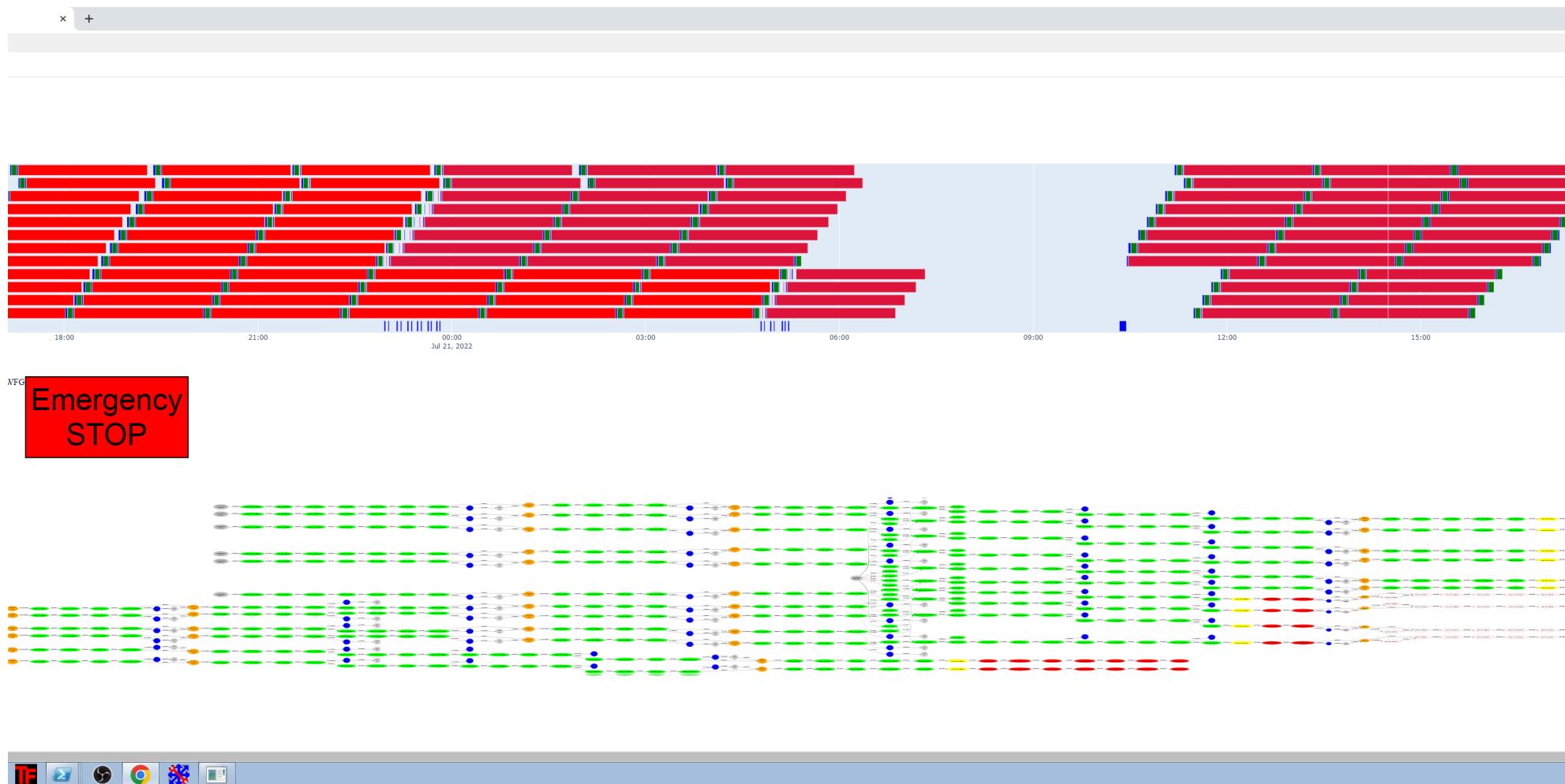


```
incubation_duration = 120
cont1 = self.containers[0]
# move to incubator
self.robot_arm.move(cont1, source_loc=cont1.pos, target_loc=self.incubator)
# incubate
self.incubator.incubate(cont1, duration=incubation_duration, temperature=310)
# move to reader
self.robot_arm.move(cont1, source_loc=cont1.pos, 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>

LARA scheduler (Stefan Maak)

LARA



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

vocabulary and ontology developments

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

vocabulary and ontology developments

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

vocabulary and ontology developments

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

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

vocabulary and ontology developments



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

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.

