



**HAMILTON**  **N**

# Python in Lab Automation

Building on solid ground

OST Rapperswil, 17 October 2024

J.Rast (@jrast on )

# Outline

- 1 How Hamilton got started with Python
- 2 ... and where we are today
- 3 A glimpse into the future



# How Hamilton got started with Python

# Providing Solutions since 1950

**HAMILTON** 

**HAMILTON**   
**MEDICAL**

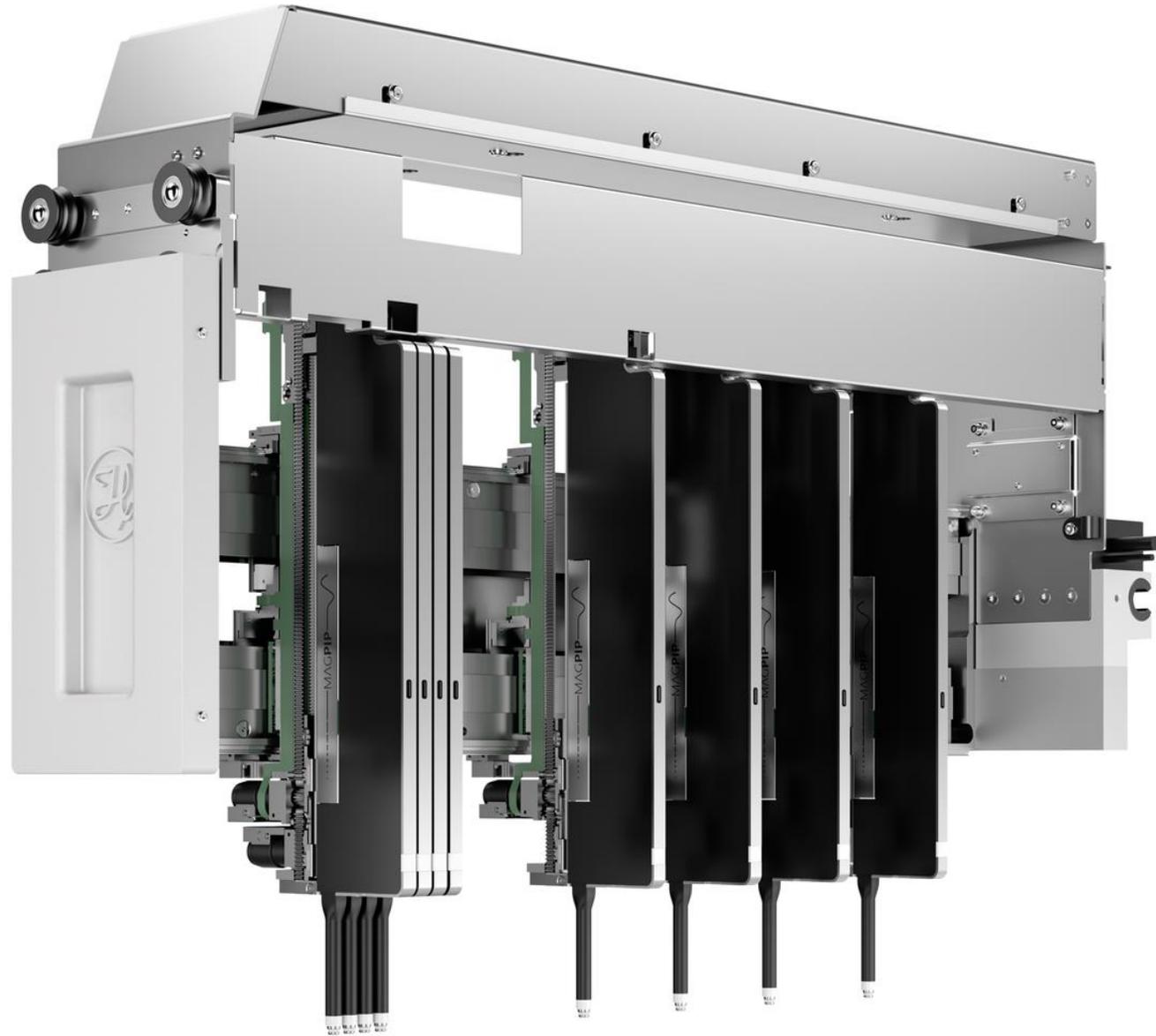


Innovative pioneer in the fields of **automated liquid handling, sample management, process sensors, laboratory equipment and ventilators.**

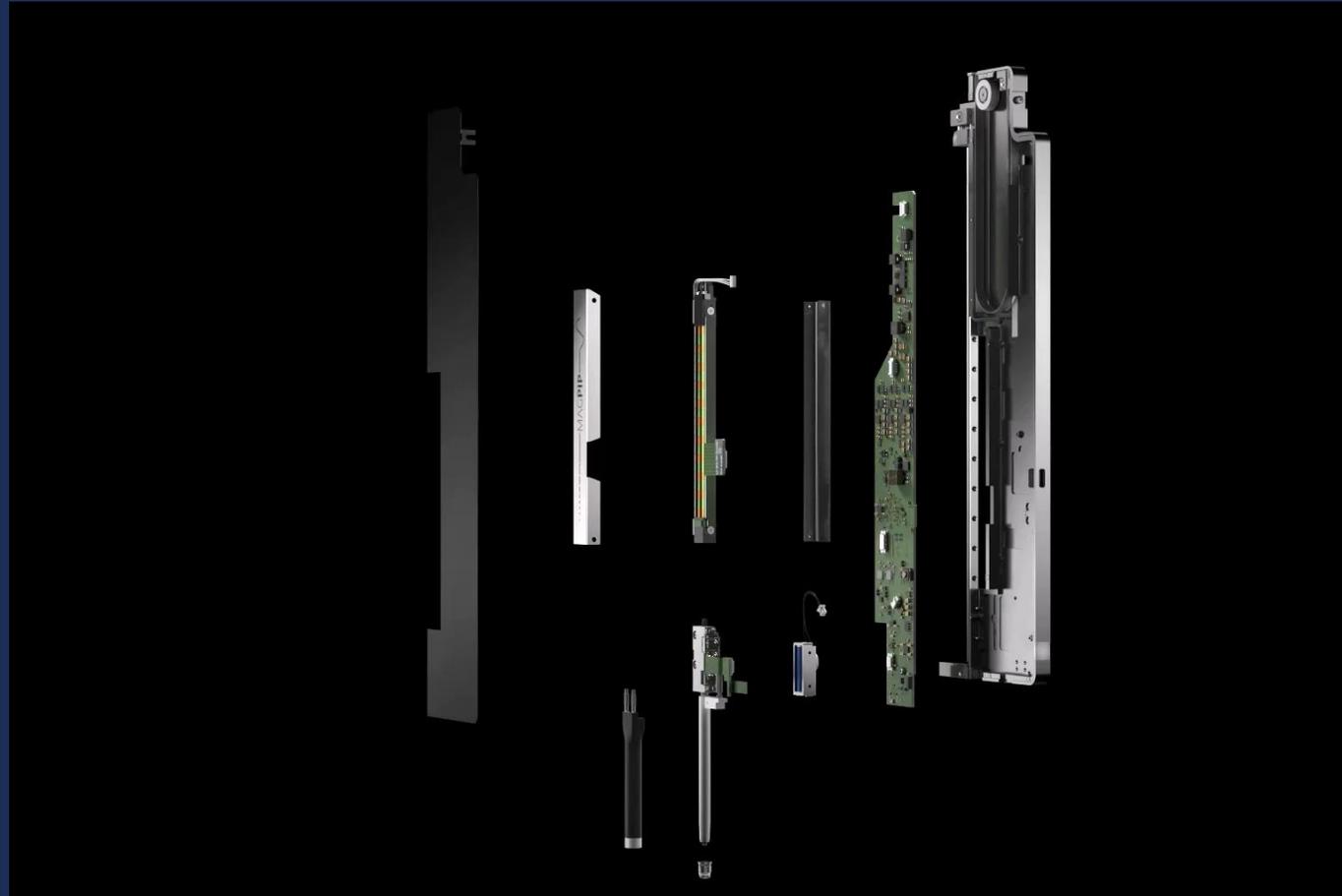
Hospitals, laboratories, Biobanks and other institutions across the globe rely on us, our products and services.

**HAMILTON** 



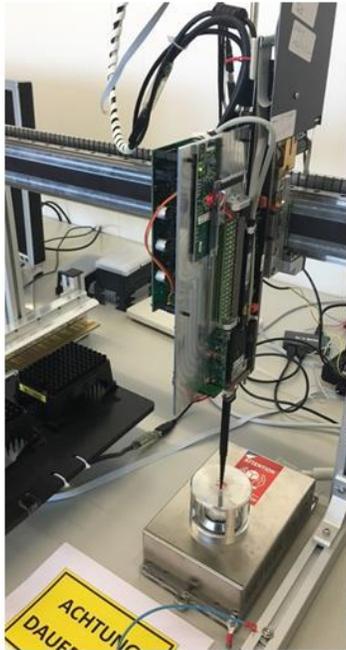


# How we got started with Python

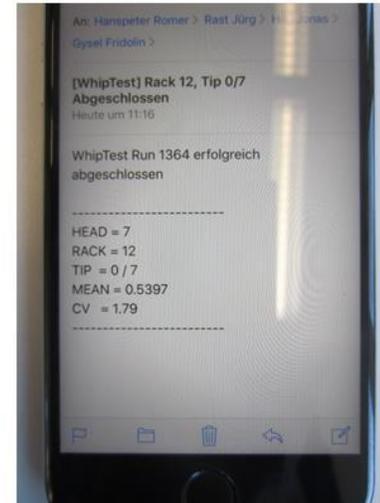
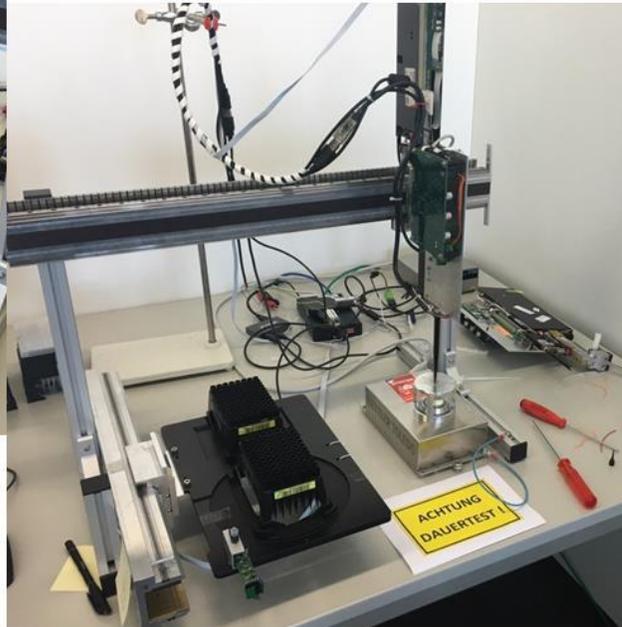


Want to find out more? Visit <https://www.youtube.com/watch?v=DIP13p4rb8w>

# How we got started with Python



Durchsatz 6 Tips a 100 Schuss / 1h  
14'400 Schuss / 24h  
Messdauer Waage 4s / Schuss



Automatischer Report  
nach 100 Aliquotes/Tip  
per Mail

Datenerfassung aller  
relevanten Parameter  
und Messgrößen in  
Datenbank

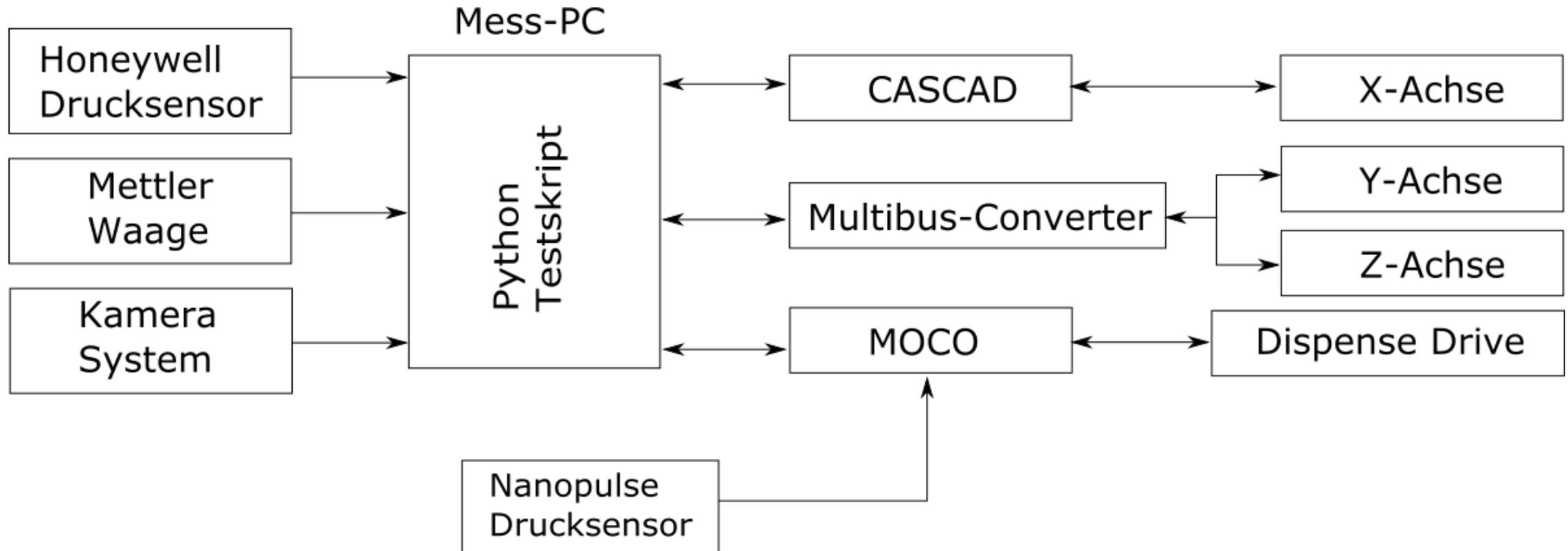
Engineer

Because badass  
miracle worker isn't a  
job title

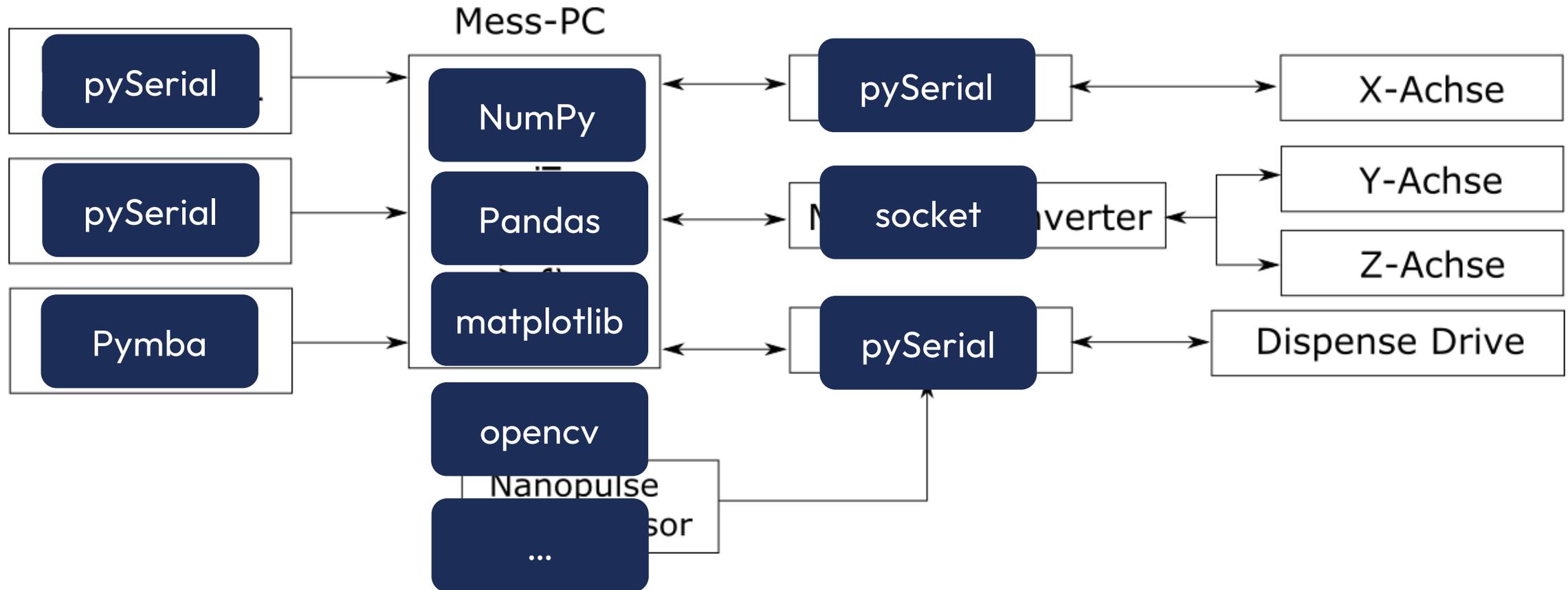


HAMILTON

# How we got started with Python



# How we got started with Python



# How we got started with Python

```
print("Writing Moco Config")
moco.read(cmds.SERR) # Dummy Read System Error
for cmd in cmds.writable():
    if hasattr(cmd.write, 'default') and cmd.write.default is not None:
        response = moco.write(cmd.number, cmd.write.default)
        if response.error:
            print("Failed Writing {} = {} to Moco".format(cmd.name, cmd.write.default))

print("Moco Alignement")
moco.action(cmds.STEP, -50000)
moco.wait_ready()
moco.action(cmds.ALGN, 0)
moco.wait_ready()
if moco.read(cmds.DERR).data > 21:
    print("Moco Alignement Failed")

print("Moco Init")
moco.action(cmds.IPOS, 200000)
moco.wait_ready()
if moco.read(cmds.DERR).data > 0:
    print("Moco Init failed")

print("Move to first Position")
moco.action(cmds.TPOS, -1000)
moco.wait_ready()
if moco.read(cmds.DERR).data > 0:
    print("Moco Move failed")
```

# How we got started with Python

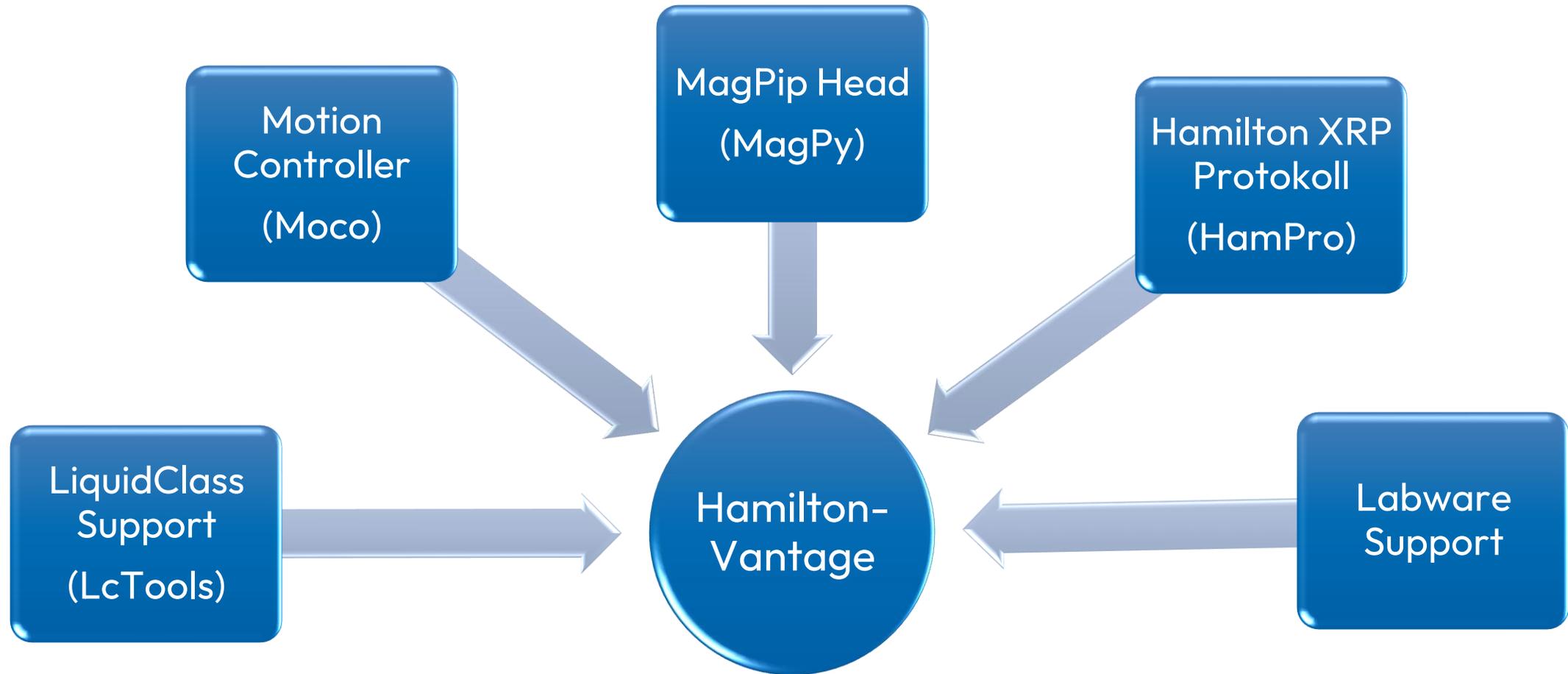
## Retrospective

- It was a mess:  
Code was changed fast and written mostly as «spaghetti code»
- It was a bit like open-heart surgery:  
If something broke, the whole system came to a halt.
- But we were fast!



... and where we are today

# ... and where we are today



# ... and where we are today

- Parallel Sampling of ~60 internal signals in FW with up to 20kHz Sampling Rate
- Transfer via USB using `python-libusb1` in real-time
- Context-Manager to start / stop the acquisition even when something goes wrong.
- Data Acquisition directly into a `Pandas` DataFrame
- Optional live data processing with up to 1kHz



```
with ana.start(fs=0.5) as df:
    arm.aspirate_clld_lw_list_lc(
        asp_source,
        target_volume,
        clld_sens=2,
        liquid_cls=liquid_cls,
        z_cmd_start=Z_USB,
        z_cmd_end=Z_USB,
        # Set Liquid Tracking enabled
        ct=[CONTAINER_ID] * 8,
        ze=[1] * 8,
    )

    response = arm.send_cmd("WMVL")
    liquid_level = arm.units.Z.arm2mm(response.vl[channel_idx])
    print(f"Liquid Level: {liquid_level:.1f}mm")
    data["liquid_levels"].append(liquid_level)

    weight_after_asp = helpers.get_weight()
    data["weights_after_asp"].append(weight_after_asp)
```

# ... and where we are today

- Working with abstract / logical positions
- Geometry modeled with NumPy
- Visualisation with vtk (work in progress ...)

```
with ana.start(fs=0.5) as df:
    arm.aspirate_clld_lw_list_lc(
        asp_source,
        target_volume,
        clld_sens=2,
        liquid_cls=liquid_cls,
        z_cmd_start=Z_USB,
        z_cmd_end=Z_USB,
        # Set Liquid Tracking enabled
        ct=[CONTAINER_ID] * 8,
        ze=[1] * 8,
    )

    response = arm.send_cmd("WMVL")
    liquid_level = arm.units.Z.arm2mm(response.vl[channel_idx])
    print(f"Liquid Level: {liquid_level:.1f}mm")
    data["liquid_levels"].append(liquid_level)

    weight_after_asp = helpers.get_weight()
    data["weights_after_asp"].append(weight_after_asp)
```

# ... and where we are today

- Complex behavior abstracted away into “Liquid Classes”
- Access to “Liquid Database” of the user software through `pyodbc`
- Access to low level functionality still retained

```
with ana.start(fs=0.5) as df:
    arm.aspirate_clld_lw_list_lc(
        asp_source,
        target_volume,
        clld_sens=2,
        liquid_cls=liquid_cls,
        z_cmd_start=Z_USB,
        z_cmd_end=Z_USB,
        # Set Liquid Tracking enabled
        ct=[CONTAINER_ID] * 8,
        ze=[1] * 8,
    )

    response = arm.send_cmd("WMVL")
    liquid_level = arm.units.Z.arm2mm(response.vl[channel_idx])
    print(f"Liquid Level: {liquid_level:.1f}mm")
    data["liquid_levels"].append(liquid_level)

    weight_after_asp = helpers.get_weight()
    data["weights_after_asp"].append(weight_after_asp)
```

# ... and where we are today

- Clean libraries for our devices
- Complexity is abstracted away
- Even non-programmers can at least understand, or reason about, what a script is doing
- With CoPilot & Co. they will soon be able to write their own scripts

# A glimpse into the future

# A glimpse into the future

## The Use of AI-Robotic Systems for Scientific Discovery

Article | [Open access](#) | Published: 20 December 2023

### Autonomous chemical research with large language models

[Daniil A. Boiko](#), [Robert MacKnight](#), [Ben Kline](#) & [Gabe Gomes](#) 

[Nature](#) **624**, 570–578 (2023) | [Cite this article](#)

**132k** Accesses | **109** Citations | **940** Altmetric | [Metrics](#)

Article | Published: 08 July 2020

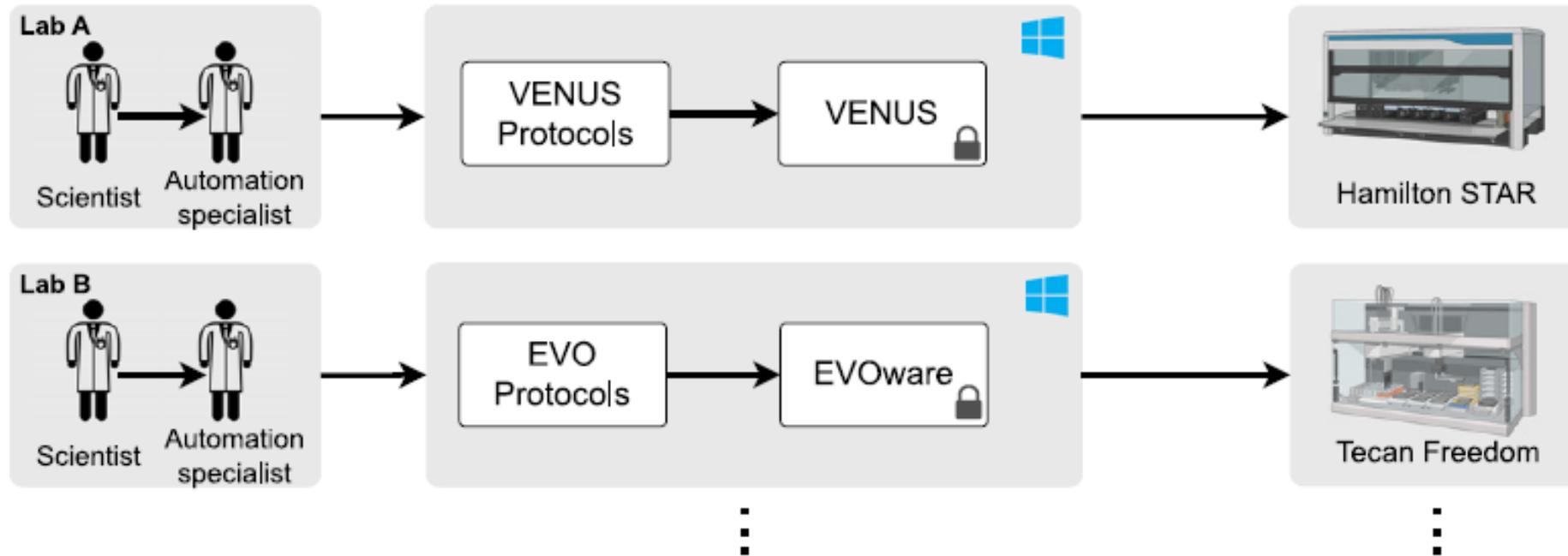
### A mobile robotic chemist

[Benjamin Burger](#), [Phillip M. Maffettone](#), [Vladimir V. Gusev](#), [Catherine M. Aitchison](#), [Yang Bai](#), [Xiaoyan Wang](#), [Xiaobo Li](#), [Ben M. Alston](#), [Buyi Li](#), [Rob Clowes](#), [Nicola Rankin](#), [Brandon Harris](#), [Reiner Sebastian Sprick](#) & [Andrew I. Cooper](#) 

[Nature](#) **583**, 237–241 (2020) | [Cite this article](#)

**89k** Accesses | **717** Citations | **1145** Altmetric | [Metrics](#)

# A glimpse into the future



# A glimpse into the future

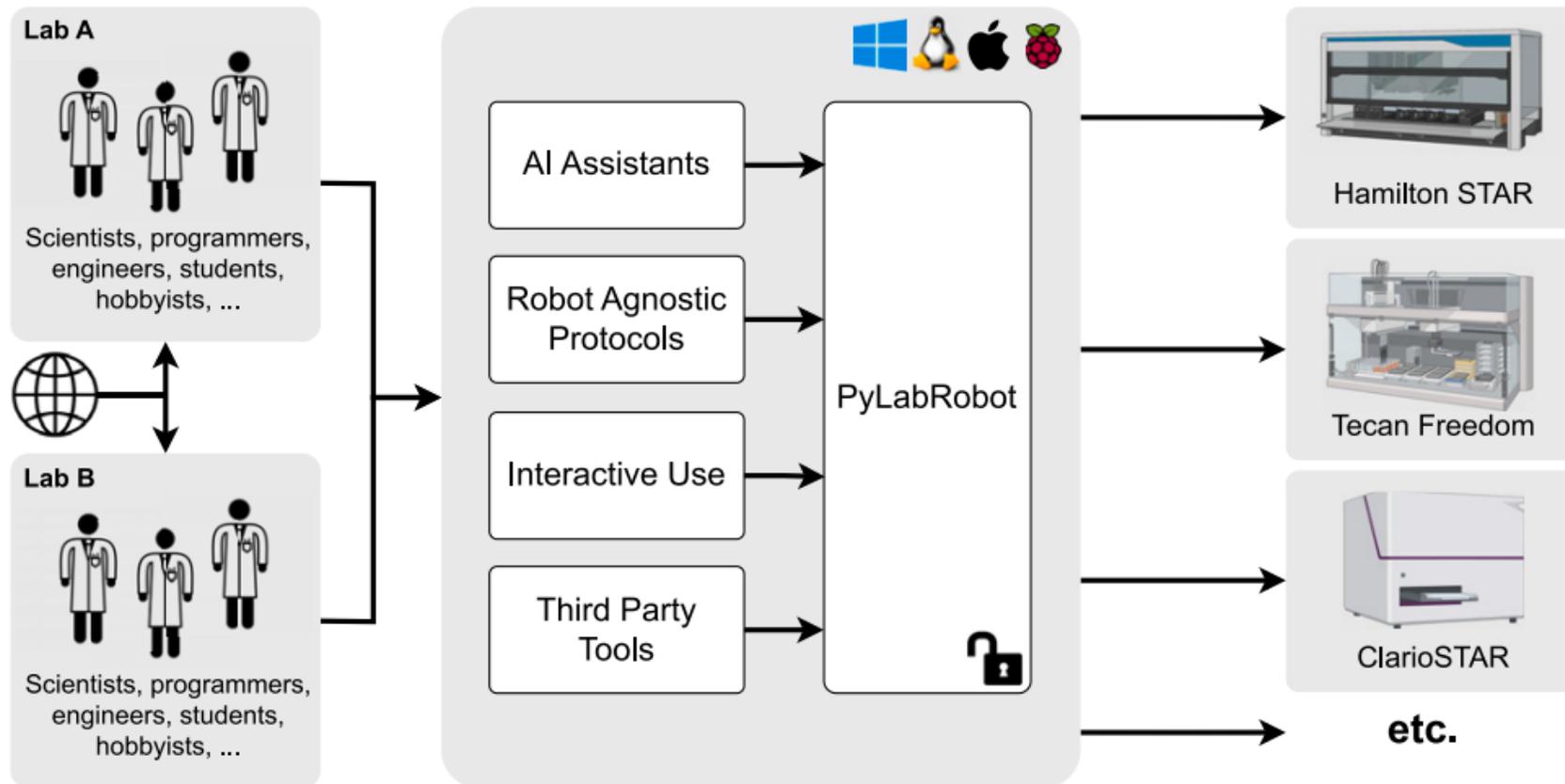
## PyLabRobot: An open-source, hardware-agnostic interface for liquid-handling robots and accessories

[Rick P. Wierenga](#)<sup>1,2</sup> · [Stefan M. Golas](#)<sup>2</sup> · [Wilson Ho](#)<sup>2,3</sup> · [Connor W. Coley](#)<sup>4</sup> · [Kevin M. Esvelt](#)<sup>2,5</sup> 

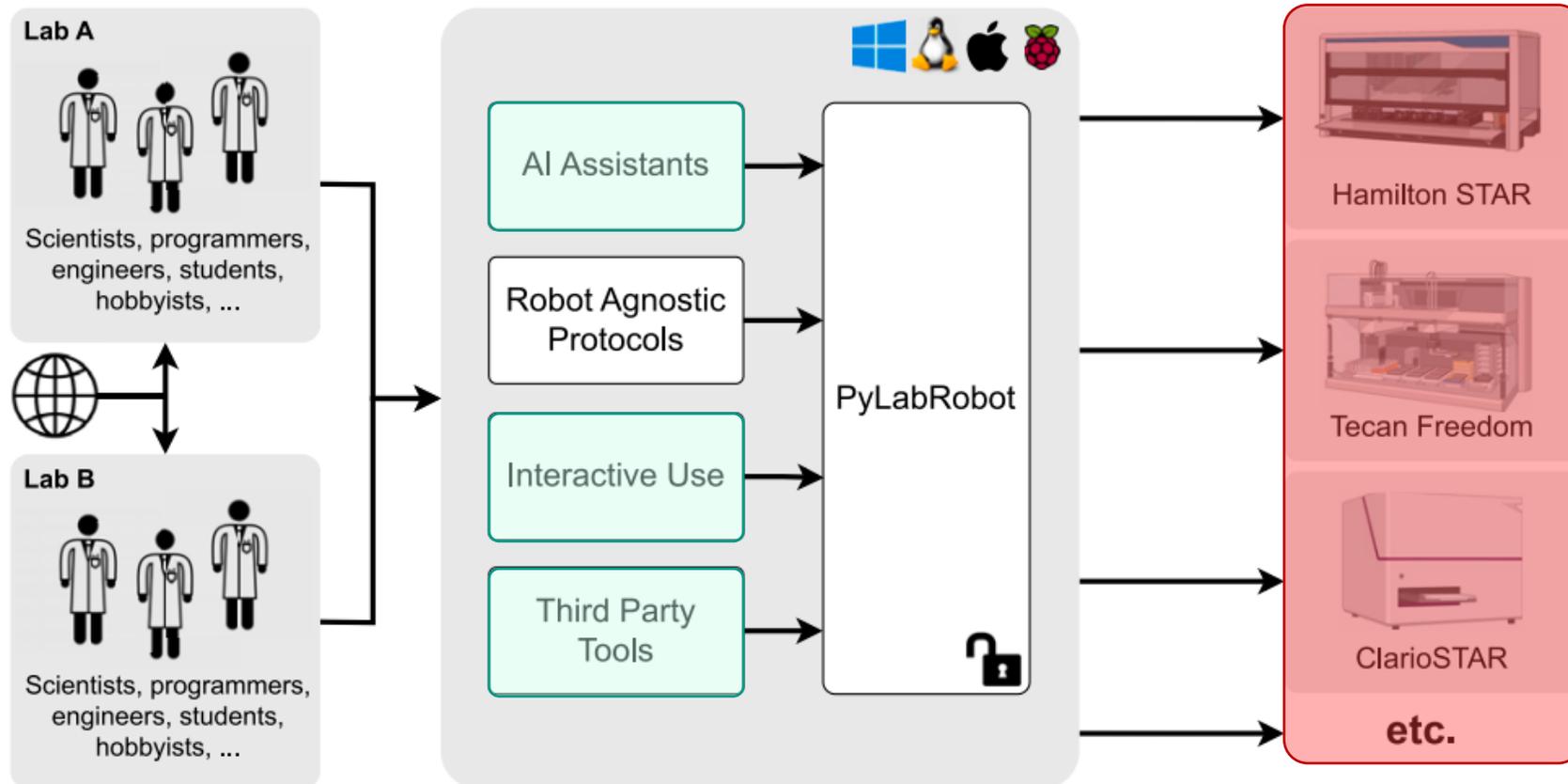
[Affiliations & Notes](#)  [Article Info](#) 

 [Download PDF](#)  [Cite](#)  [Share](#)  [Set Alert](#)  [Get Rights](#)  [Reprints](#)

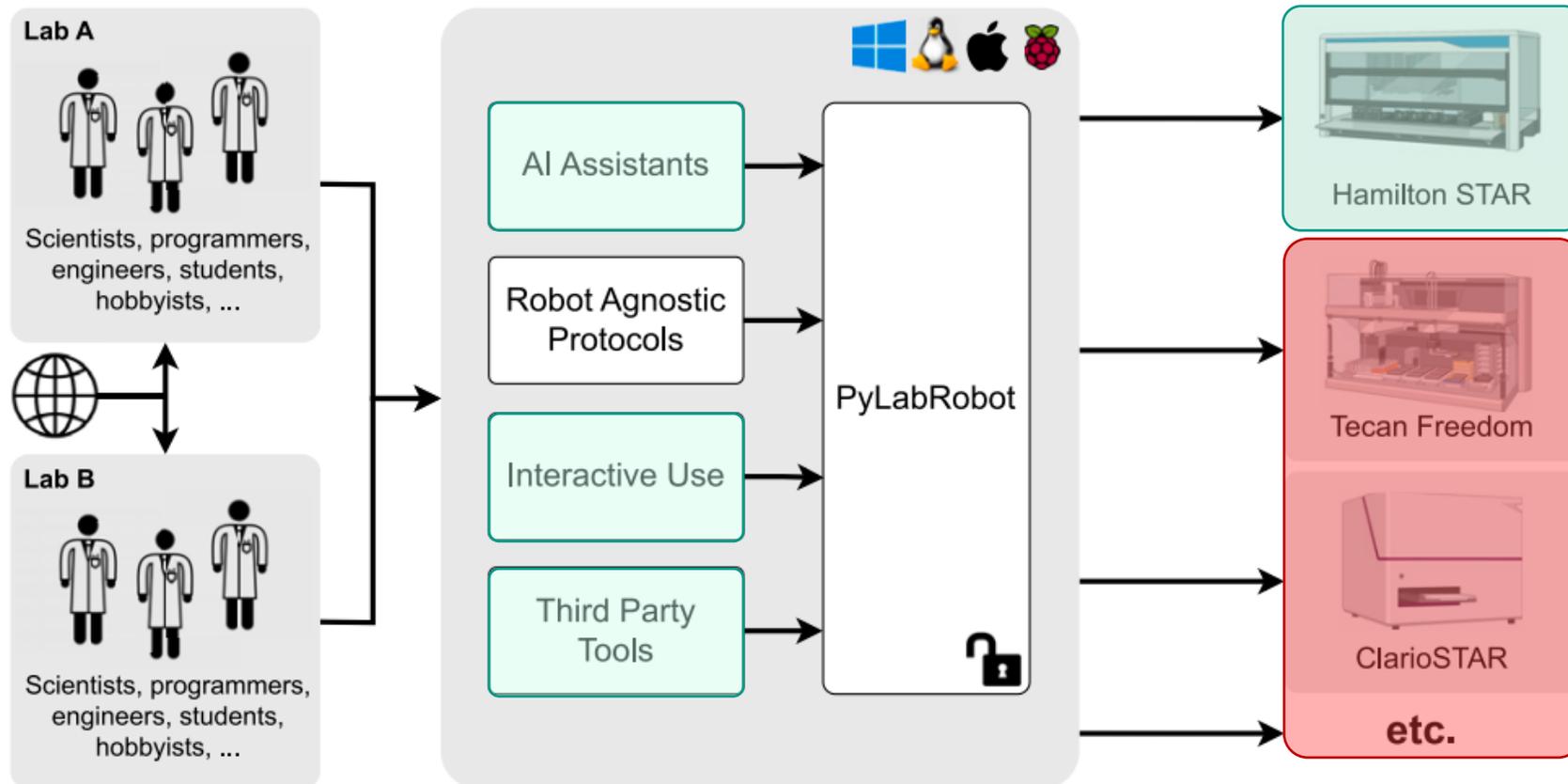
# A glimpse into the future



# A glimpse into the future



# A glimpse into the future







Do you have  
any questions ?

**Hamilton Americas & Pacific Rim**

4970 Energy Way  
Reno, Nevada 89502 USA  
Tel: +1-775-858-3000

**Hamilton Europe, Asia & Africa**

Via Crusch 8  
CH-7402 Bonaduz  
Tel: +41 58 610 10 10