



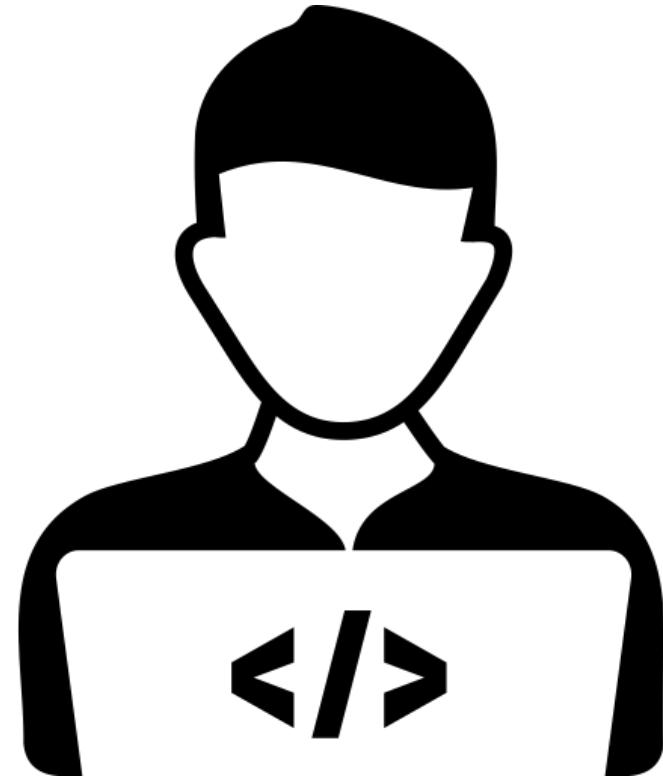
# Covalent

A Workflow Orchestration Platform for  
MLOps

# whoami?

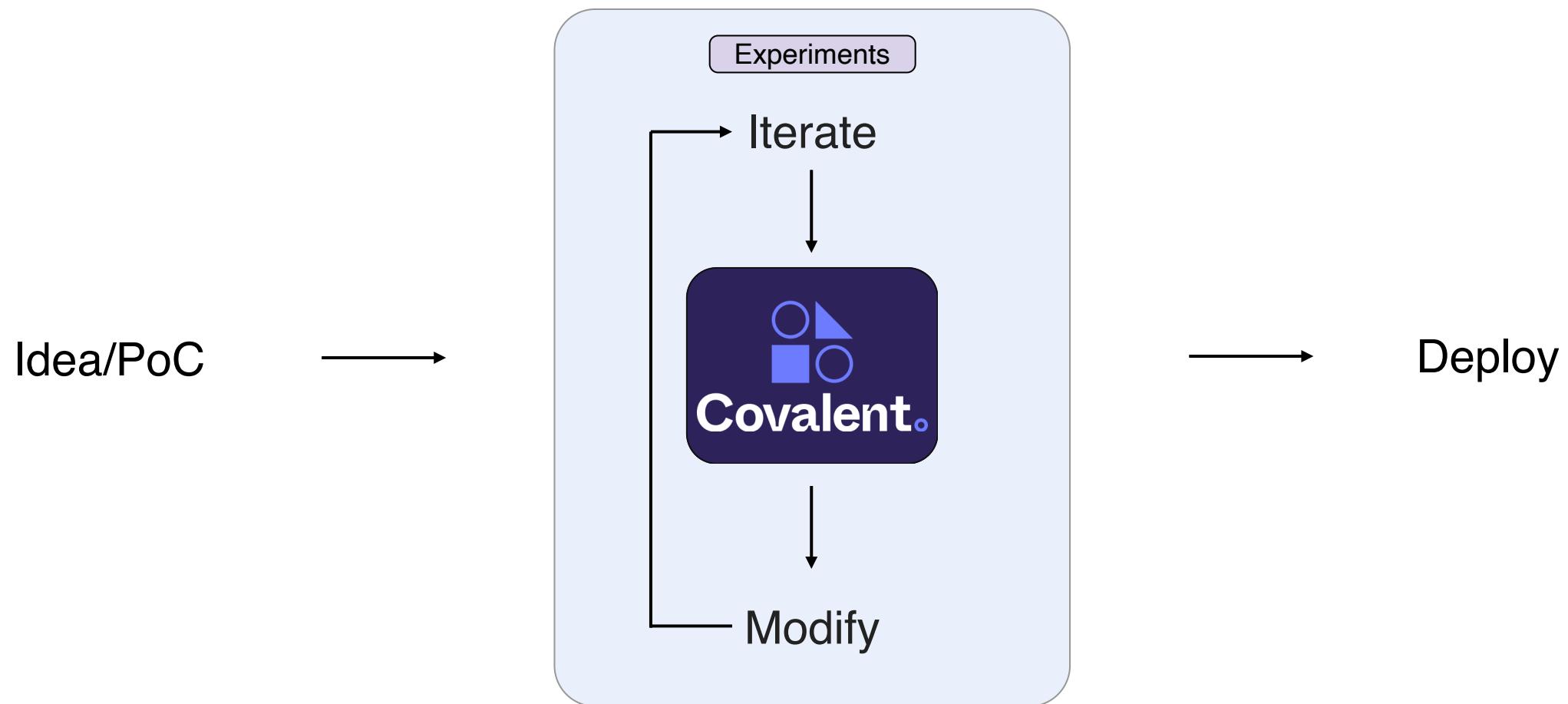
Sankalp Sanand

Software Engineer at Agnostiq



# Covalent.

*Make your complex workflows modular, scalable  
and reproducible.*



# Heterogenous Hardware



CPU



FPGA



GPU



TPU

# Challenges.

*Computational*



## Hardware-potpourri.

Hybrid experiments. Single experiment now contains CPU+GPU+QPU+TPU



## Interactive Experimentation.

Rapid prototyping and iterating with minimal setup overhead.



## Distributed computation.

Era of high-throughput calculations. Running **massive parallel jobs** at scale.



## Limited resource.

Long queue times, and waiting serially even in the case of **independent tasks!**

# Challenges.

*Experimental-Organization*



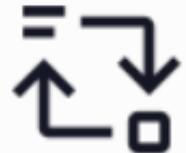
## Manageability.

- Organize 1000s of experiments
- Experimental versioning of multiple runs
- Input/parameters logging for each run
- Checkpointing costly computations
- Job failure management
- Real time monitoring



## Reproducibility.

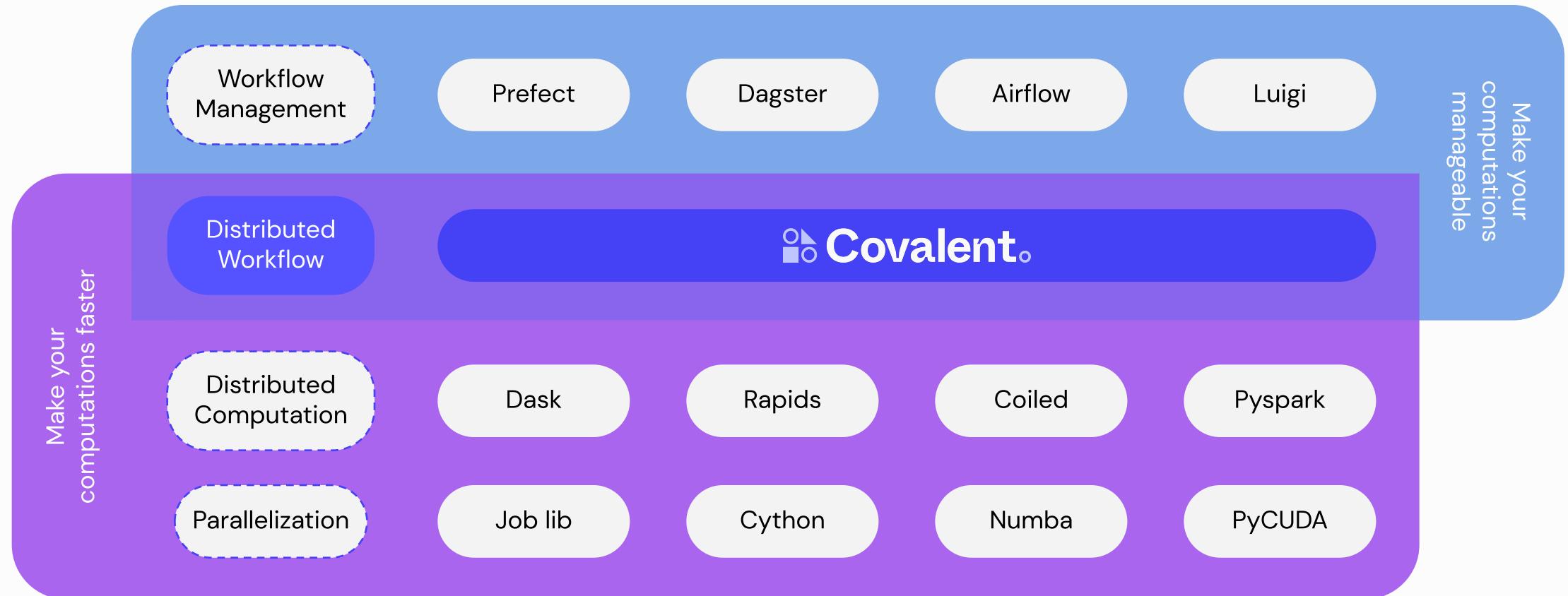
- Environment saving/caching
- Hardware metadata caching
- Inputs/parameters logging
- Experiment dependant device setup



## Shareability.

- Experiment organization
- Clear and intuitive code structure
- Reproducible experiment parameters and setup

# Where Covalent fits in the stack.



# 4. Real time monitoring

## Visual overview

*Visualize and share your workflow to transfer knowledge as fast as possible*

## Status/Error updates

*Get real time Updates on errors and completion*

## Checkpoints

*Stores anything and everything automatically without a single line of code*

## Meta-data

*Maintains details from environment to hardwares used*

## Parameter

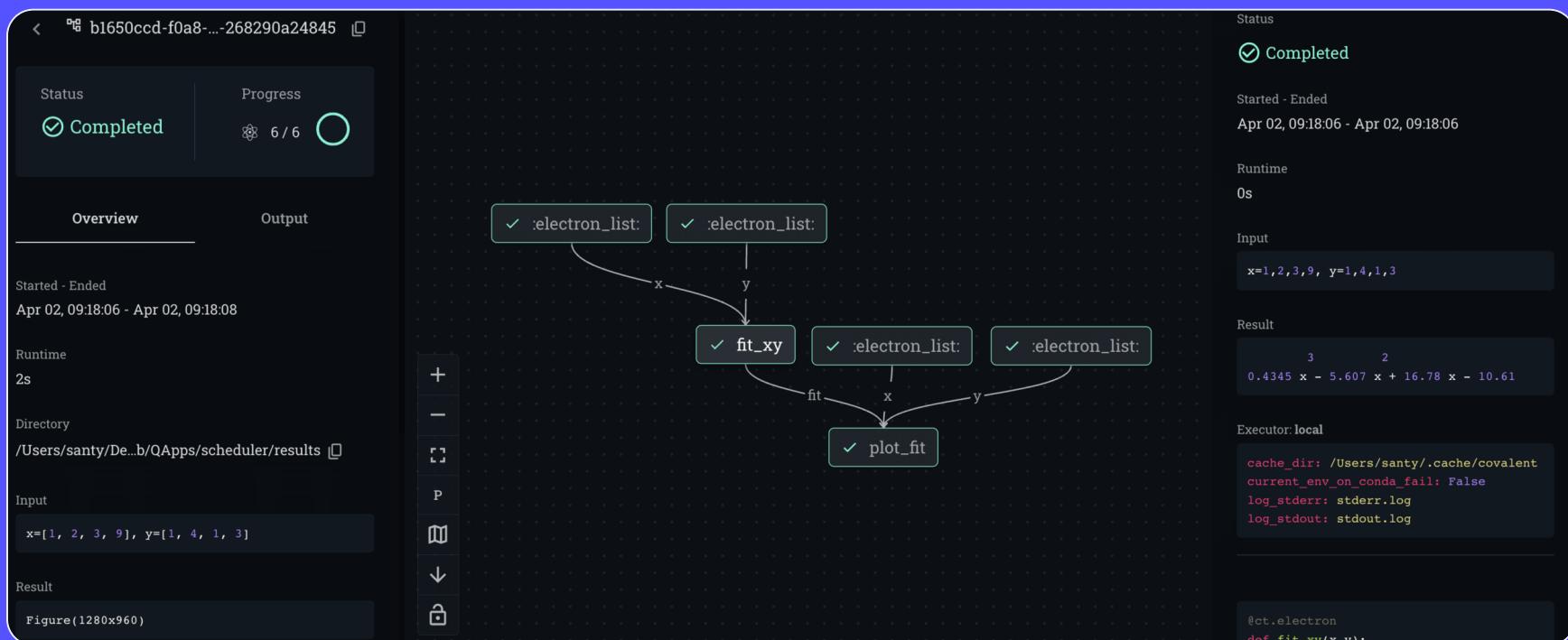
*Never forget the hyper-parameters that worked*

## Interactive

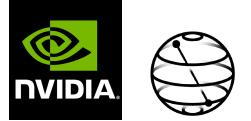
*Start/stop/manage your HPC jobs right from the UI*

**Beautiful and interactive UI**

*bring your workflows to life!*



# Hardware resources



## Cloud resources

- SLURM Executor
- AWS Fargate Executor
- AWS Batch Executor
- Azure Executor Coming soon
- GCP Executor Coming soon
- Kubernetes Executor Coming soon

## Other resources

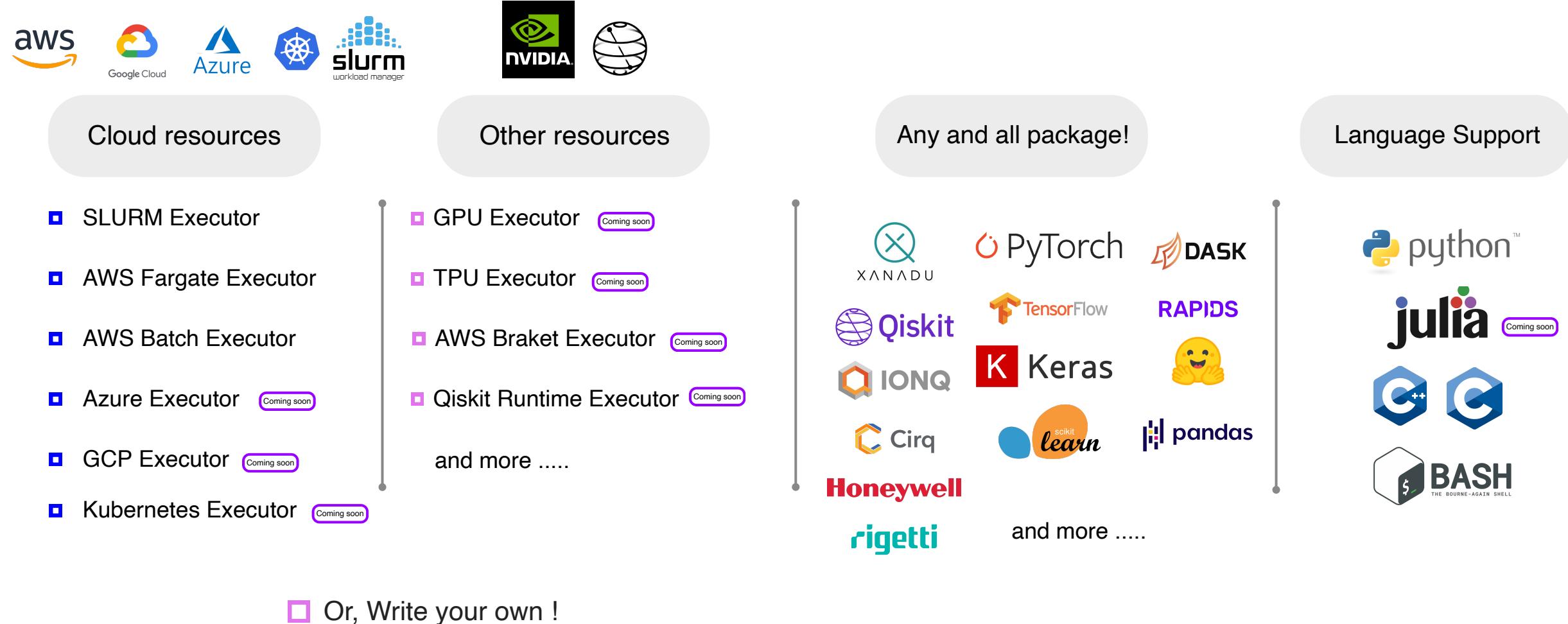
- GPU Executor Coming soon
- TPU Executor Coming soon
- AWS Braket Executor Coming soon
- Qiskit Runtime Executor Coming soon

and more .....

```
1 import covalent as ct
2 from covalent.executor import FargateExecutor,SlurmExecutor
3
4 slurm=SlurmExecutor(options ={"cpus-per-task": 8},
5                      conda_env="cova")
6 aws_fargate=FargateExecutor(vcpu=1, memory=2)
7
8
9 @ct.electron(executor=slurm)
10 def fit_xy(x,y):
11     z = np.polyfit(x, y, 3)
12     return np.poly1d(z)
13
14 @ct.electron(executor=aws_fargate)
15 def square(x):
16     return x**2
```

□ Or, Write your own !

# Flexible with language.



# Demo



# Summary.

 [github.com/AgnostiqHQ/covalent](https://github.com/AgnostiqHQ/covalent)



Drop us a  and contribute !

<i>Pythonic workflows</i>	<i>Automatic checkpointing</i>	<i>Multiple Language support</i>	<i>Little-to-no overhead</i>	<i>Customizable</i>
<i>Reproducibility</i>	<i>Code locally, run anywhere</i>	 <b>Covalent.</b> Its Open-Source !	<i>Intuitive User-interface</i>	<i>Natively hybrid workflows</i>
<i>Native parallelization</i>				<i>variety of executors</i>
<i>Code isolation</i>	<i>Parameter caching</i>	<i>Cloud Agnostic</i>	<i>Interactive jobs</i>	<i>Start locally and scale</i>