



Evaluating Kubernetes batch scheduling systems for containerized declarative data analyses

CERN openlab Summer Student Lightning Talks

Author:

Xavier Tintin

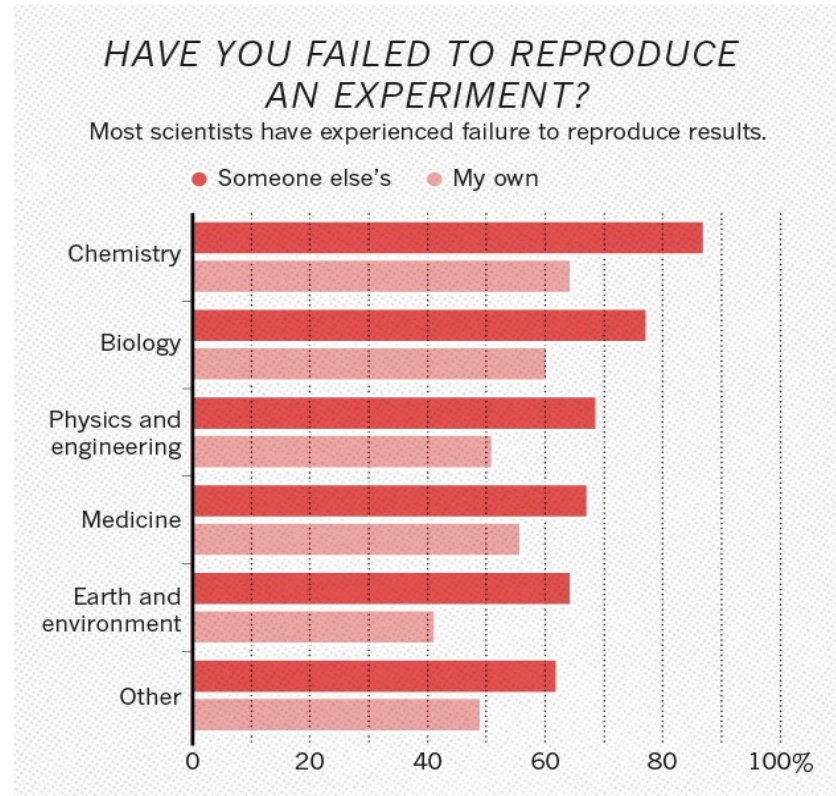
Supervisors:

Marco Donadoni

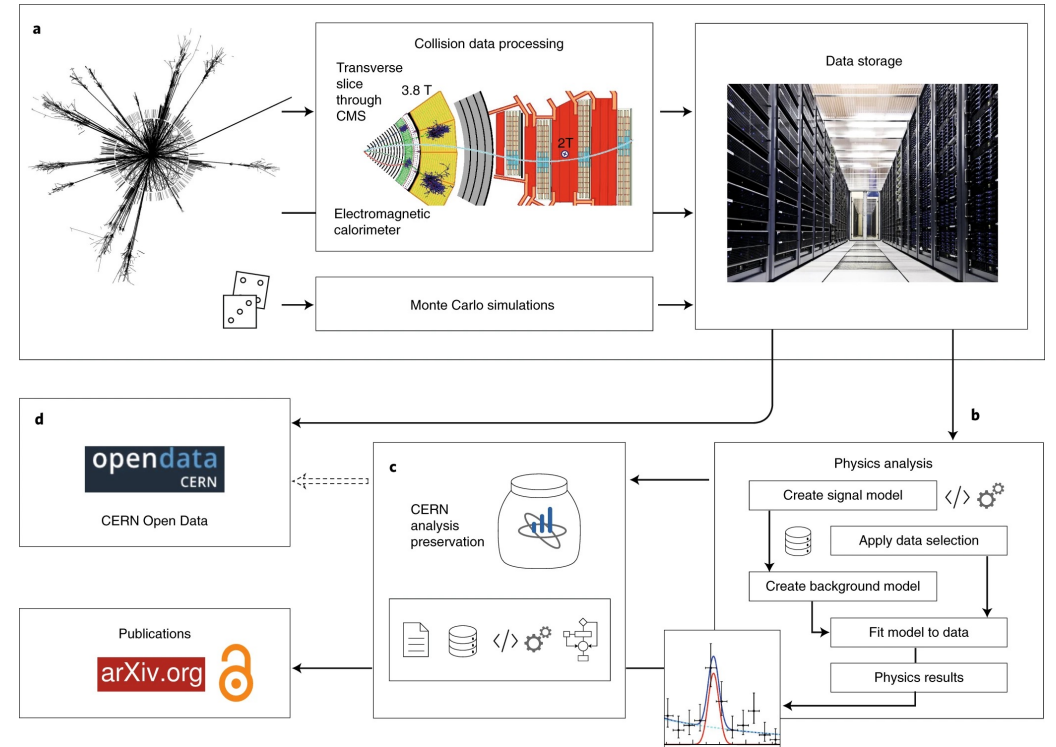
Tibor Šimko

Reproducible and reusable science

When testing new theories, building on previous data analyses is necessary, but **very** challenging



<https://doi.org/10.1038/533452a>

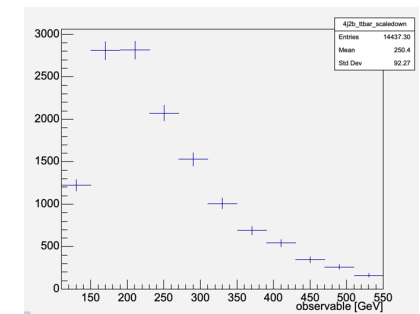
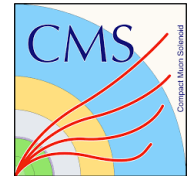
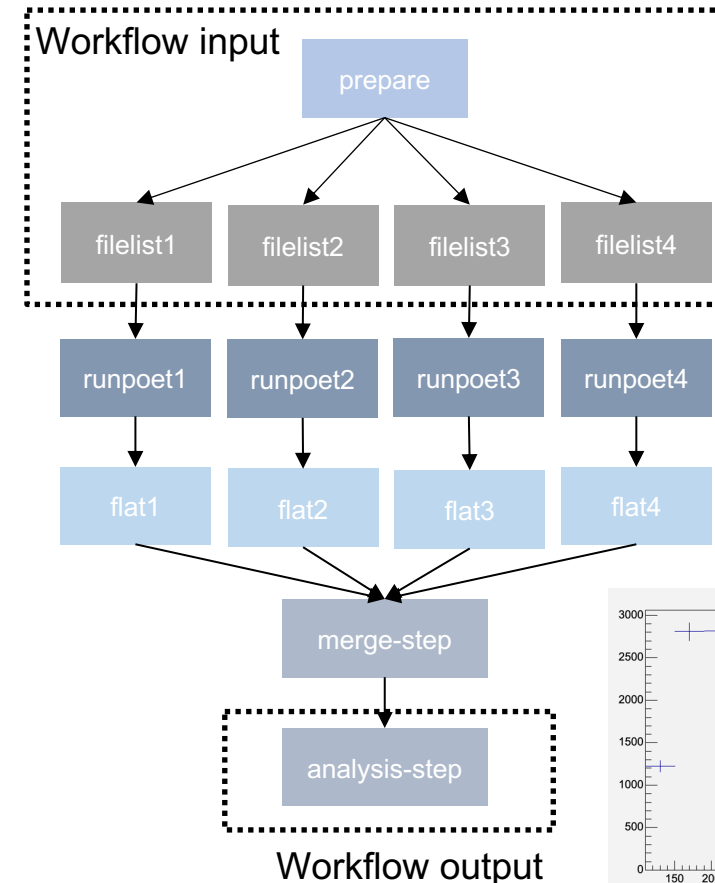


Scientific workflows represent the **complex** flow of data that through various steps of collection, transformation, and analysis produces published results.

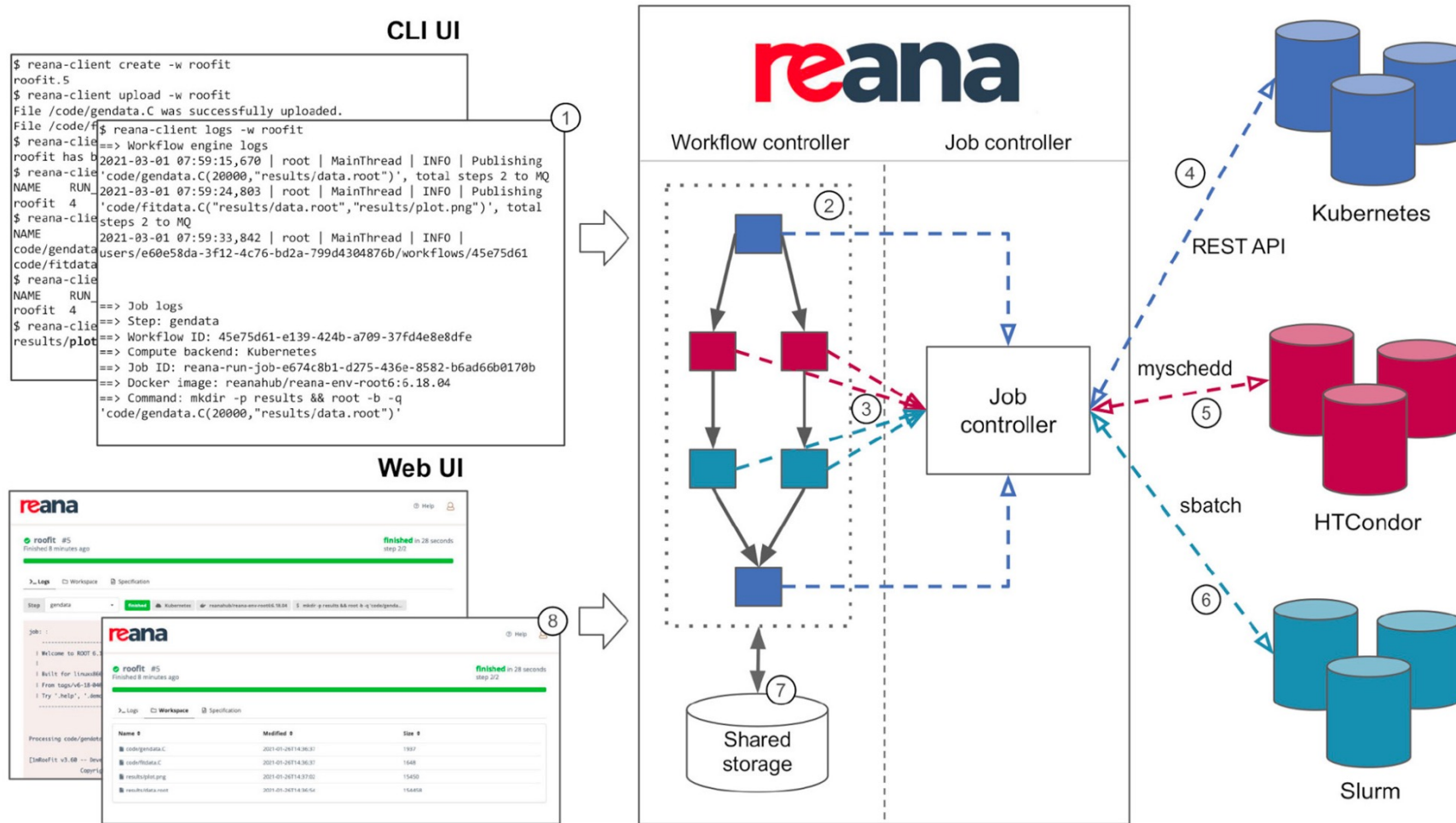
Declarative HEP analysis workflows

Workflows

- A workflow is a recipe explaining how to compute results from the input data
- What is needed?
 - Input data
 - Code
 - Computing environment
 - Workflow steps
- Each step can use different container images
- Steps can be described by means of declarative workflow languages

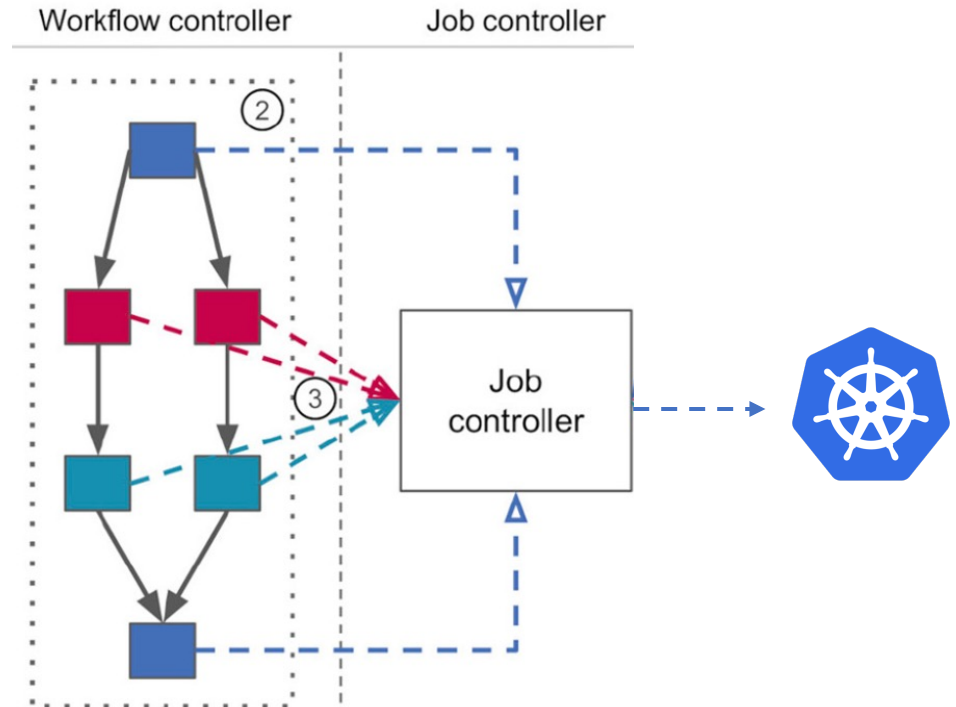


REANA: reusable analyses



Current solution: Kubernetes API

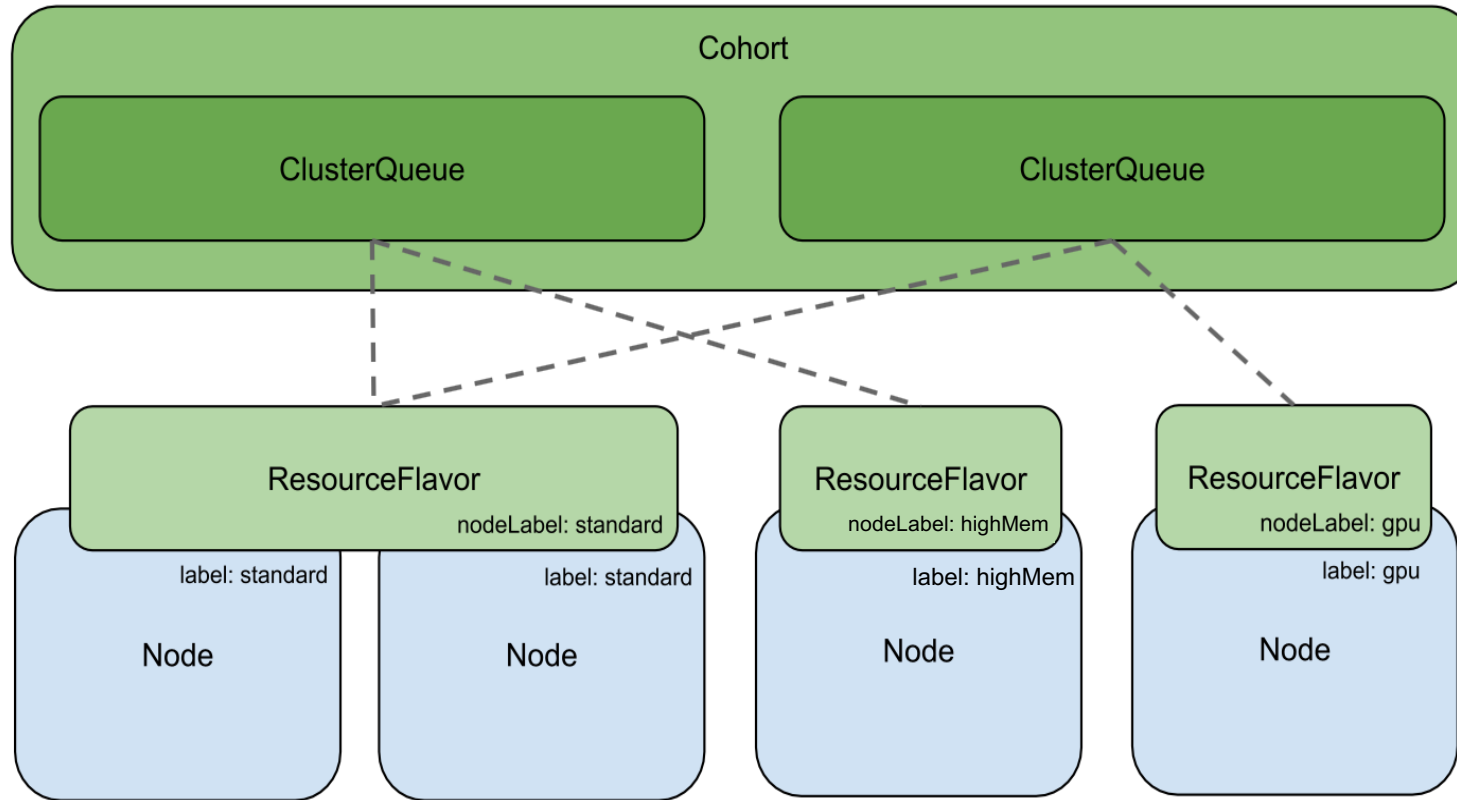
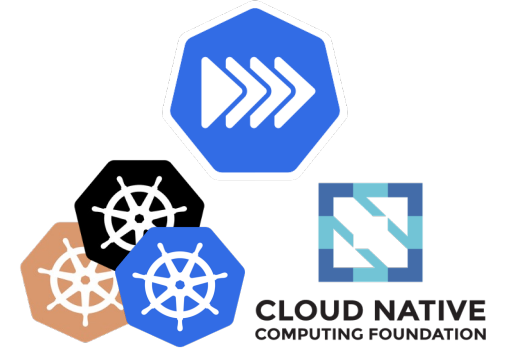
```
class KubernetesJobManager(JobManager):  
  
    @JobManager.execution_hook  
    def execute(self):  
        """Execute a job in Kubernetes."""  
        backend_job_id = build_unique_component_name("run-job")  
        self.job = {  
            "kind": "Job",  
            "apiVersion": "batch/v1",  
            "metadata": {  
                "name": backend_job_id,  
                "namespace": REANA_RUNTIME_KUBERNETES_NAMESPACE,  
            },  
            "spec": {  
                "backoffLimit": KubernetesJobManager.MAX_NUM_JOB_RESTARTS,  
                "autoSelector": True,  
                "template": {  
                    "metadata": {  
                        "name": backend_job_id,  
                        "labels": {"reana-run-job-workflow-uuid": self.workflow_uuid},  
                    },  
                },  
            },  
        }
```



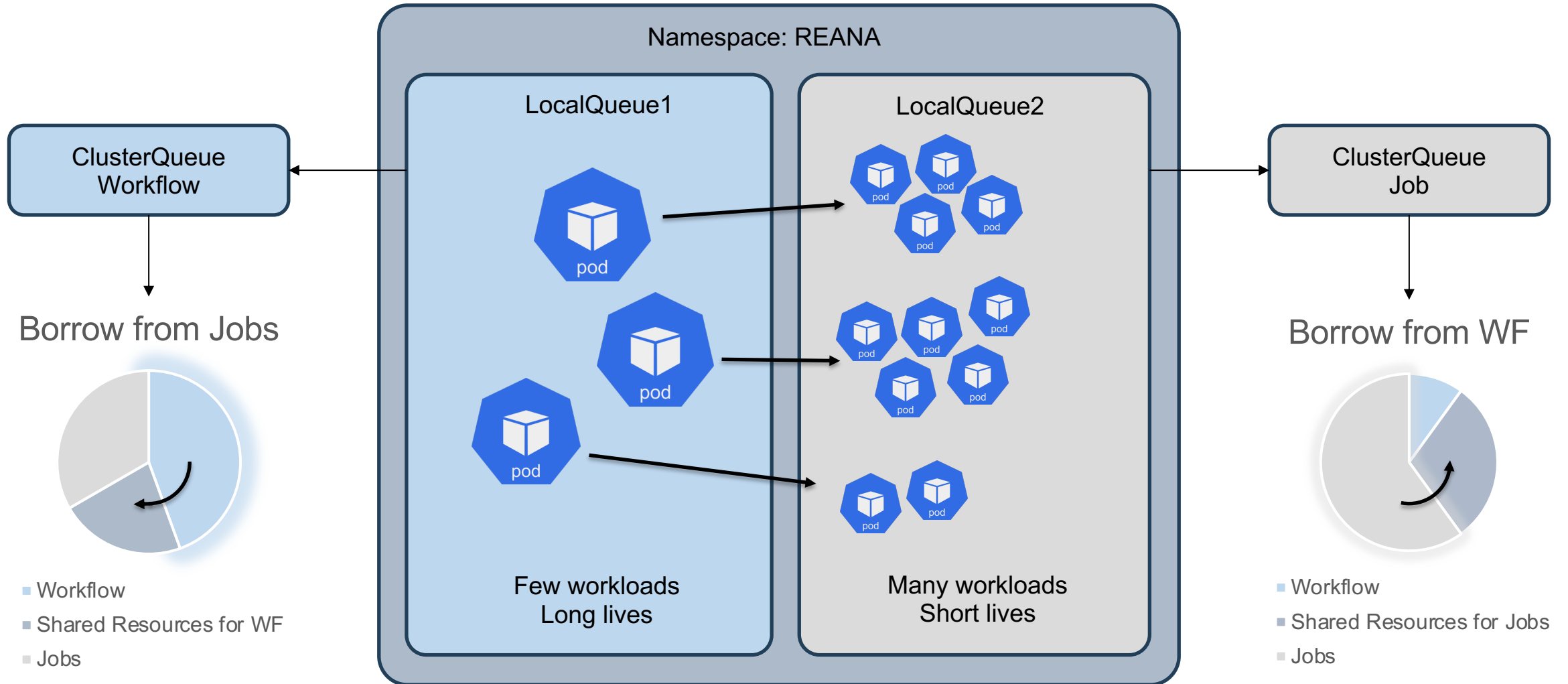
REANA job controller uses native Kubernetes Job API and necessitated custom developments for efficient and fair workflow scheduling.

New kids on the block?

Kueue: Kubernetes-native batch scheduling system



Kueue evaluation setup



Typical evaluation experiment

```

apiVersion: batch/v1
kind: Job
metadata:
  namespace: reana # Job under reana namespace
  generateName: sample-job-
  annotations:
    kueue.x-k8s.io/queue-name: lq-jobs # Point to the LocalQueue
spec:
  parallelism: 1 # This Job will have 1 replica running at the same time
  completions: 1 # This Job requires 1 completion
  suspend: true # Set to true to allow Kueue to control the Job when it starts
  template:
    spec:
      containers:
      - name: dummy-job
        image: gcr.io/k8s-staging-perf-tests/sleep:latest
        args: ["5s"] # Sleep for 5 seconds
        resources:
          requests:
            cpu: "4m"
            memory: "150M"
          limits:
            cpu: "4m"
            memory: "150M"
        restartPolicy: Never
  
```

Context: default
Cluster: reana-summer
User: admin
K9s Rev: v0.27.4
K8s Rev: v1.25.3
CPU: 5%
MEM: 47%

<0> all <a> Attach <l> Logs
<1> reana <ctrl-d> Delete <p> Logs Prev
<2> default <d> Describe <shift-f> Port-Forw
 <e> Edit <s> Shell
 <?> Help <n> Show Node
 <ctrl-k> Kill <f> Show Port

Pods(reana)[433]

NAME ↑	PF	READY	RESTARTS	STATUS	CPU	MEM	%CPU/R	%CPU/L	%MEM/R
reana-run-batch-39b55b1e-58af-412c-8648-6ca455d7be60-gqfdx	●	2/2	0	Running	0	0	n/a	n/a	n/a
reana-run-batch-40a4572a-32c9-47fd-abb9-2b32c8a2774c-5mf2j	●	2/2	0	Running	0	0	n/a	n/a	n/a
reana-run-batch-46d81a79-de2f-451b-aa31-139d5c0bf55b-4tsw7	●	2/2	0	Running	0	0	n/a	n/a	n/a
reana-run-batch-48cdb861-0e94-4121-83ac-ed2d8b195ff8-rr94l	●	2/2	0	Running	0	0	n/a	n/a	n/a
reana-run-batch-48d67496-1e91-4026-bf5a-3cda7d953650-4pvhr	●	2/2	0	Running	0	0	n/a	n/a	n/a

Node: All ▾

Network I/O pressure

Total usage

Cluster memory usage

Used: 336.50 GiB Total: 728.73 GiB

Cluster CPU usage (1m avg)

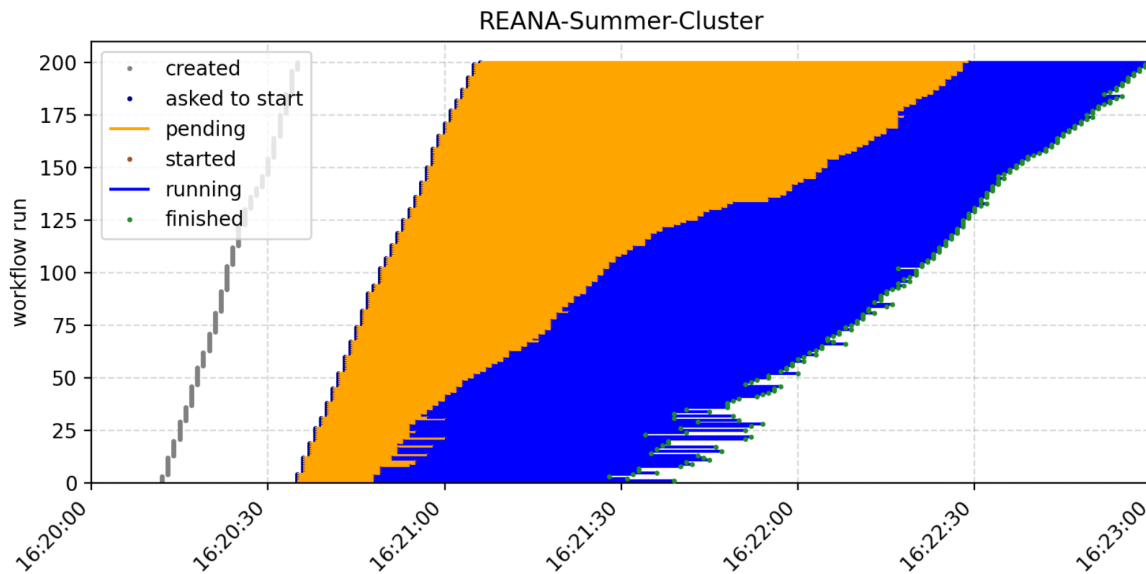
Used: 22.06 Total: 408.00

Cluster filesystem usage

Used: 982.86 GiB Total: 3.98 TiB

First results

reana



200 workflows running in parallel on 46 nodes

Kueue succeeded on several demo examples for thousands of submissions

Encountered a few corner cases when submitting a high number of jobs

- Kueue UUID job allocation issues
- Unresponsive when overloaded with 20k jobs

```
Error from server (InternalError): error when creating "Workflows/wf-999.yaml": Internal error occurred: failed calling webhook "mjob.kb.io": failed to call webhook: Post "https://kueue-webhook-service.kueue-system.svc:443/mutate-batch-v1-job?timeout=10s": dial tcp 10.254.242.140:443: connect: connection refused
Error from server (InternalError): error when creating "Workflows/wf-9990.yaml": Internal error occurred: failed calling webhook "mjob.kb.io": failed to call webhook: Post "https://kueue-webhook-service.kueue-system.svc:443/mutate-batch-v1-job?timeout=10s": dial tcp 10.254.242.140:443: connect: connection refused
Error from server (InternalError): error when creating "Workflows/wf-
```

Kueue is promising... but still maturing

Thank you Questions?

xavier.tintin@cern.ch

<https://www.linkedin.com/in/xavier-tintin/>

 [@reanahub](#)