



UNIVERSITY OF
NOTRE DAME



Abstracting container technologies and transfer mechanisms in the Scalable CyberInfrastructure for Artificial Intelligence and Likelihood Free Inference (SCAILFIN) project

Kenyi Hurtado, representing the SCAILFIN team



UNIVERSITY OF
NOTRE DAME



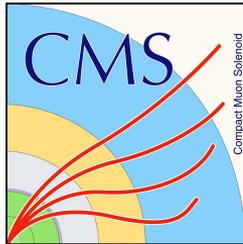
SCAILFIN

*Scalable CyberInfrastructure for Artificial Intelligence and
Likelihood Free Inference*

Motivation

Major Multi-User Research Facilities involve the comparison of data collected from experiments with “synthetic” data, produced from computationally-intensive simulations.

Comparisons of experimental data and predictions from simulations are abstractions of the specific data analysis techniques developed by the respective communities over several decades. E.g.:



ICECUBE
SOUTH POLE NEUTRINO OBSERVATORY



Motivation

Many of these data analysis tasks are often conducted manually or through *ad hoc scripts* that might not be well maintained, making reproducibility and reusability difficult. Many of these tasks do have a well-defined workflow that make automation possible, though.

REANA was created (in collaboration with DASPOS, DIANA and CERN) to address the reproducibility and reusability of the analysis pipeline.



Reproducible research data analysis platform

Motivation

In parallel:

Interest in leveraging Machine Learning (ML) and Artificial Intelligence (AI) techniques, to enhance the analysis of data from these facilities.

In particular, its application with emergent **Likelihood-Free Inference (LFI) techniques** when the predictions for the data are implicitly defined by the simulation, often leading to an intractable likelihood function. This can apply to analysis of data from LHC, LIGO, etc, but such Likelihood-Free algorithms have so far been **implemented mostly on individual machines and in ad hoc scripts because the training workflows are very complicated.**

Introduction

SCAILFIN: Scalable CyberInfrastructure for Artificial Intelligence and Likelihood Free Inference

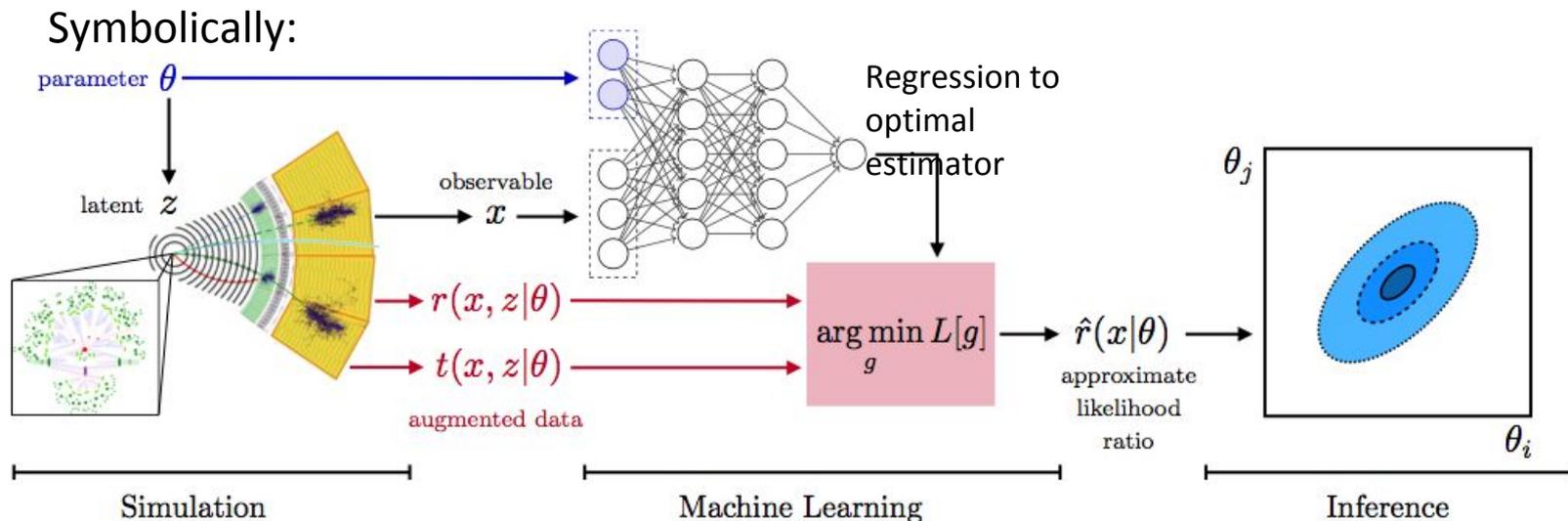
The SCAILFIN project aims to deploy artificial intelligence and likelihood-free inference techniques and software using scalable cyberinfrastructure (CI) that is developed to be integrated into existing CI elements, such as the **REANA** system, to work on **HPC facilities**.

PI's: Mark Neubauer, Dan Katz

Kyle Cranmer, Heiko Mueller

Mike Hildreth





Estimation of optimal estimator lends itself to ML methods:

- Training data derived from simulations
- Can be guided by optimal sampling based on phase space density of generator, sensitivity to physics under study

Today's Topic

SCAILFIN goals

- ...
- One of the goals of the project is **extending the REANA** platform to allow remote **submission** of workflows to **HPC facilities**.
 - General description of SCAILFIN components and functionality
 - Monday Presentation (Mike Hildreth)
<https://indico.cern.ch/event/773049/contributions/3474811/>
 - Focus on integrating different container technologies and file management into the job submission component
 - This talk
- ...

Background

SCAILFIN components

- **REANA** as the Cyber Infrastructure element to deploy AI and Likelihood-Free inference techniques.
- **VC3** (Virtual Clusters for Community Computation) in order to scale REANA to HPC resources.

More details: Monday Presentation (Mike Hildreth)

<https://indico.cern.ch/event/773049/contributions/3474811/>

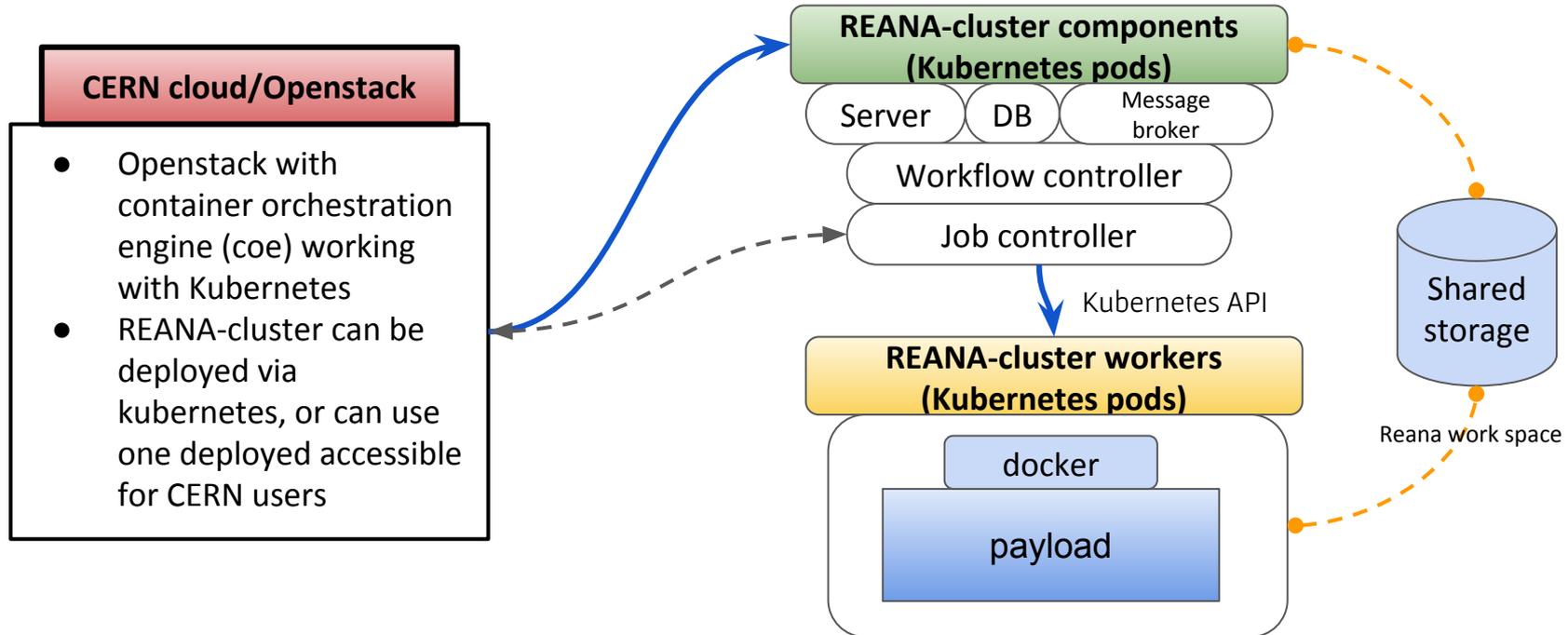
reana: Reproducible Research Data Analysis Platform

Components

- Two major components each consisting of many sub-components
 - **reana-client**: User facing component.
 - Accepts workflows and is used as interface to entire REANA system (for user).
 - **reana-cluster**: Workhorse.
 - Consists of many small pieces which handle workflows, dish out jobs, coordinates results, can be thought of as the job scheduler. Jobs are scheduled via Kubernetes.

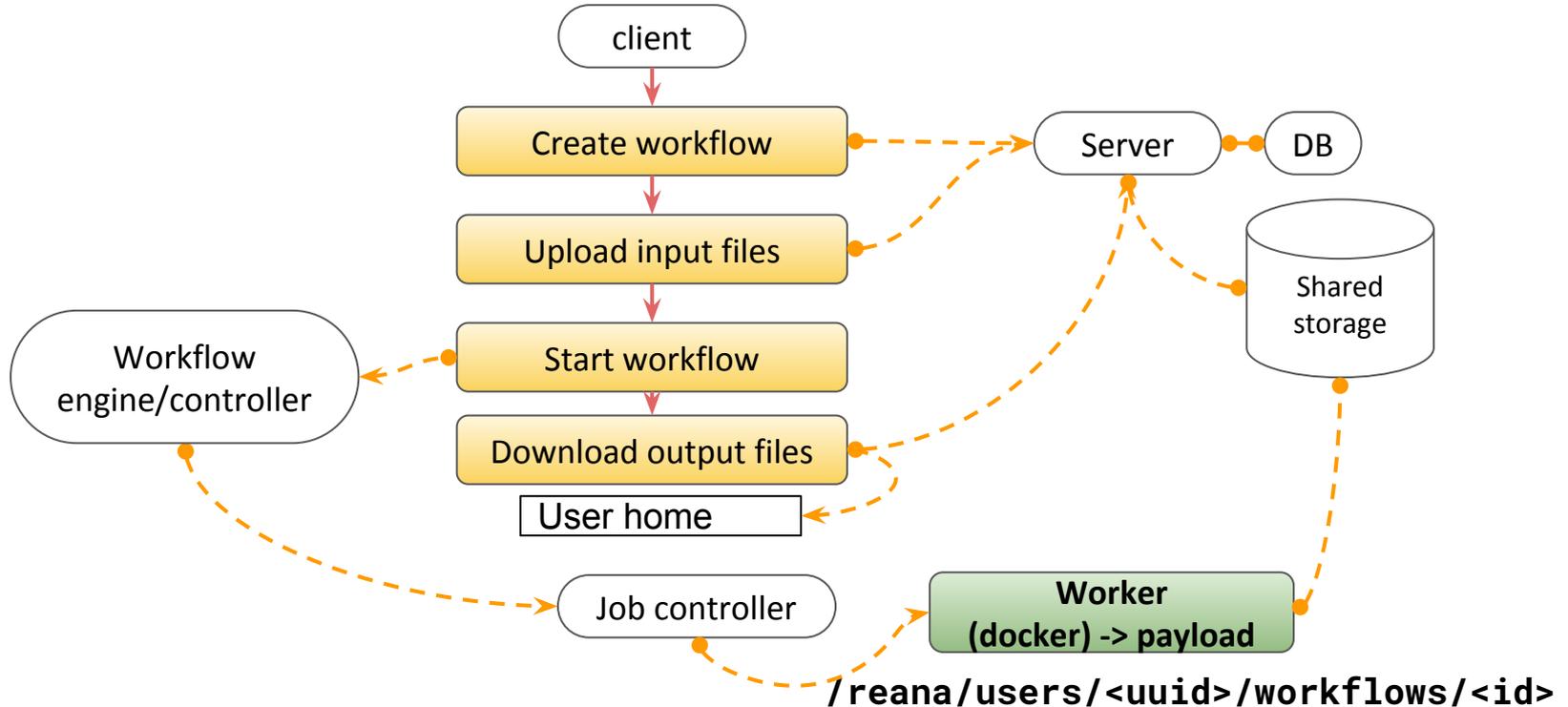
REANA cluster / Workforce Infrastructure

Standard kubernetes deployment

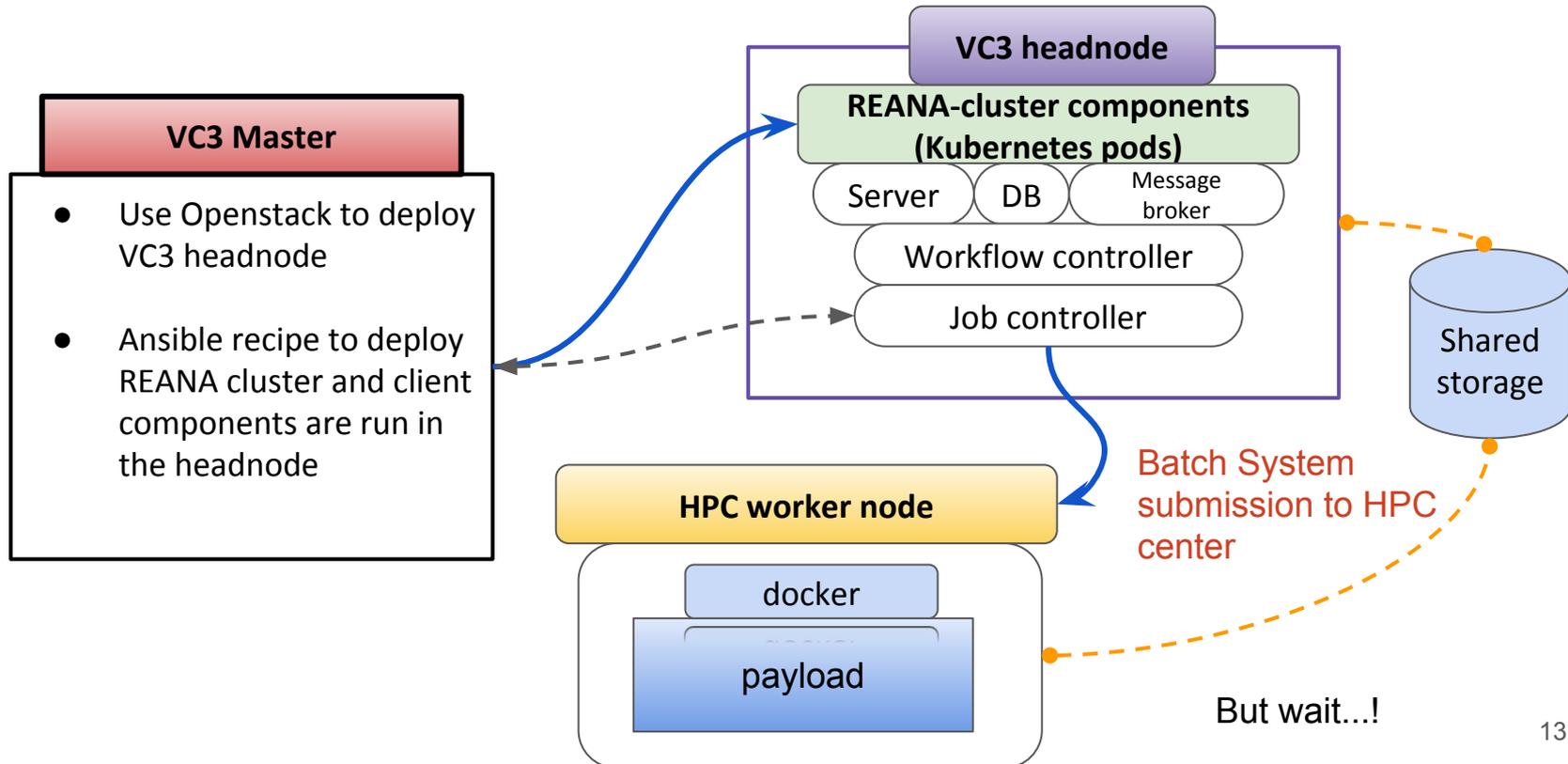


REANA client / User's perspective

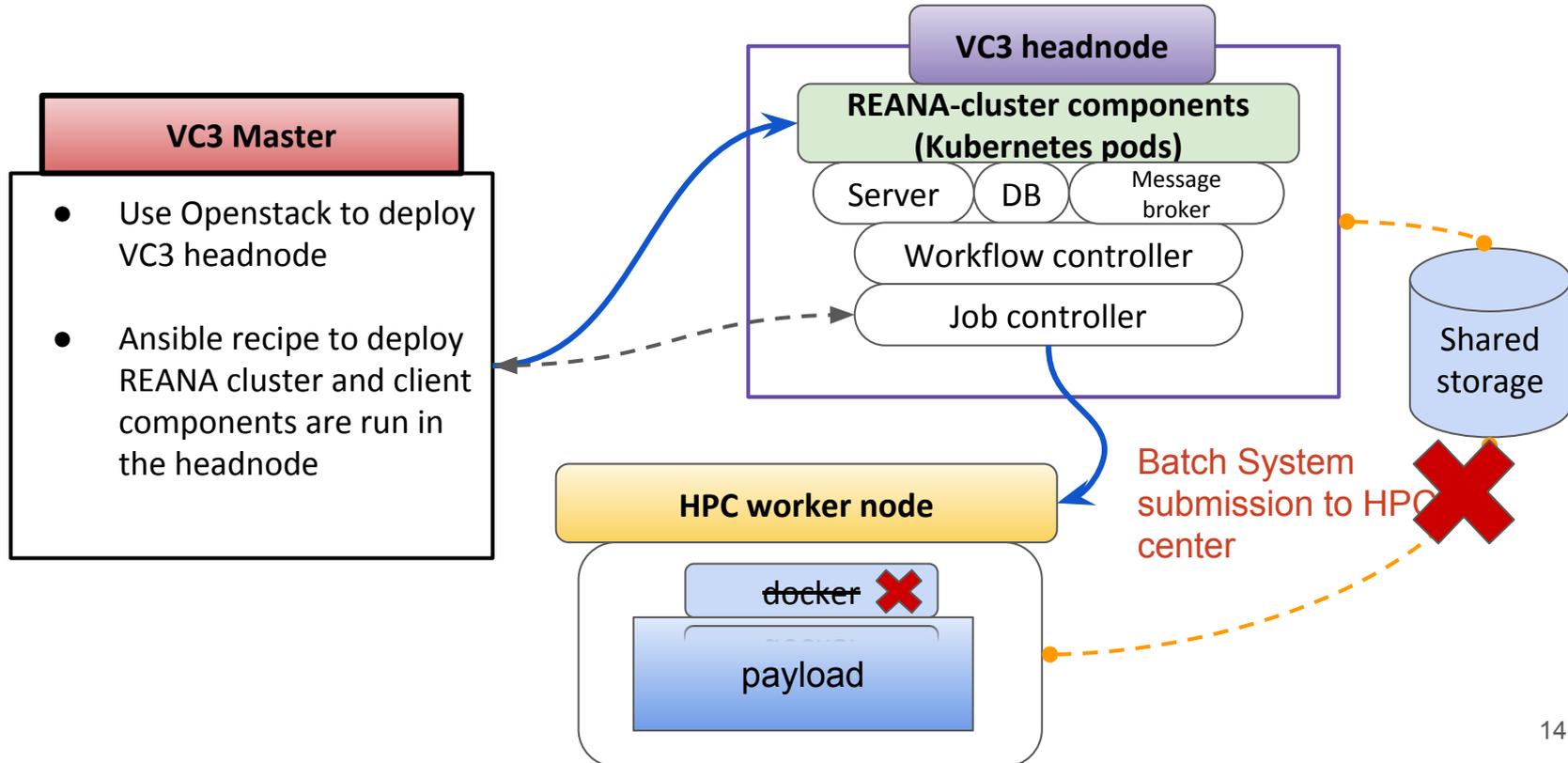
Starting a workflow



REANA cluster / Workforce Infrastructure SCALFIN deployment

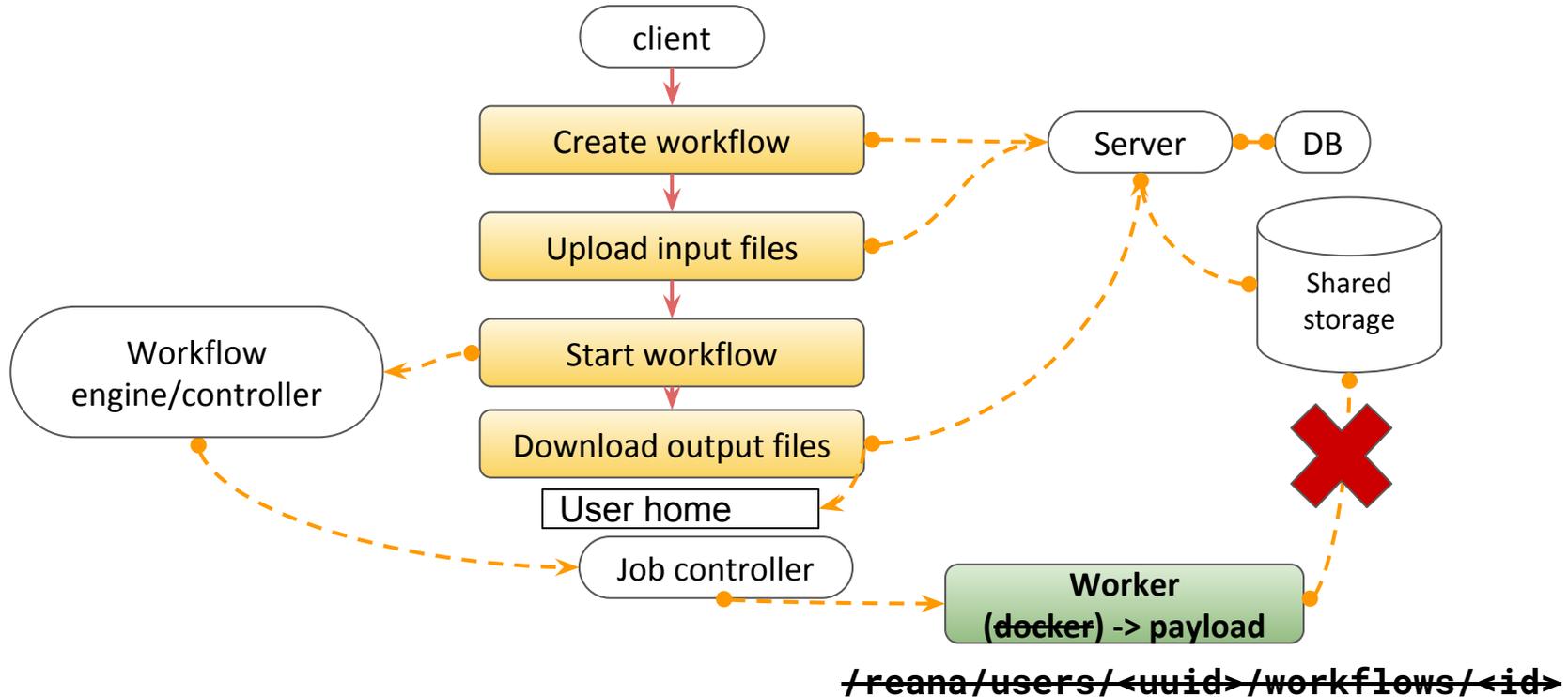


REANA cluster / Workforce Infrastructure SCALFIN deployment



REANA client / User's perspective

Starting a workflow



REANA client / User's perspective

Starting a workflow

SCALFIN deployment issues...

- The job controller can't use Kubernetes or Docker
- The server can upload files, but the workers won't see those files

`/reana/users/<uuid>/workflows/<id>`

Container technologies in HPC facilities

- HPC centers are no strangers to the user's need for containers nowadays.
 - Docker is not an option though (security reasons)
 - User-space container technologies preferred instead. E.g.:
 - Singularity: PSC, Stampede, Comet, etc.
 - Shifter: NERSC, Blue Waters
 - Charlie Cloud: Unprivileged containers using linux user namespaces (linux 4.4+)
 - All options above have mechanisms to run/convert docker images

Notre Dame CCTools

- **Chirp:** Lightweight user-level FS for collaboration across distributed systems such as clusters, clouds, and grids. An ordinary user can share storage space and data without requiring any sort of administrator privileges anywhere. Supports multiple authentication mechanisms.
- **Parrot:** A tool for attaching existing programs to remote I/O systems through the filesystem interface. E.g.:
 - `$ parrot_run vi /chirp/server.nd.edu/mydata`
 - `$ parrot_run cp /path/file /chirp/server.nd.edu/file`

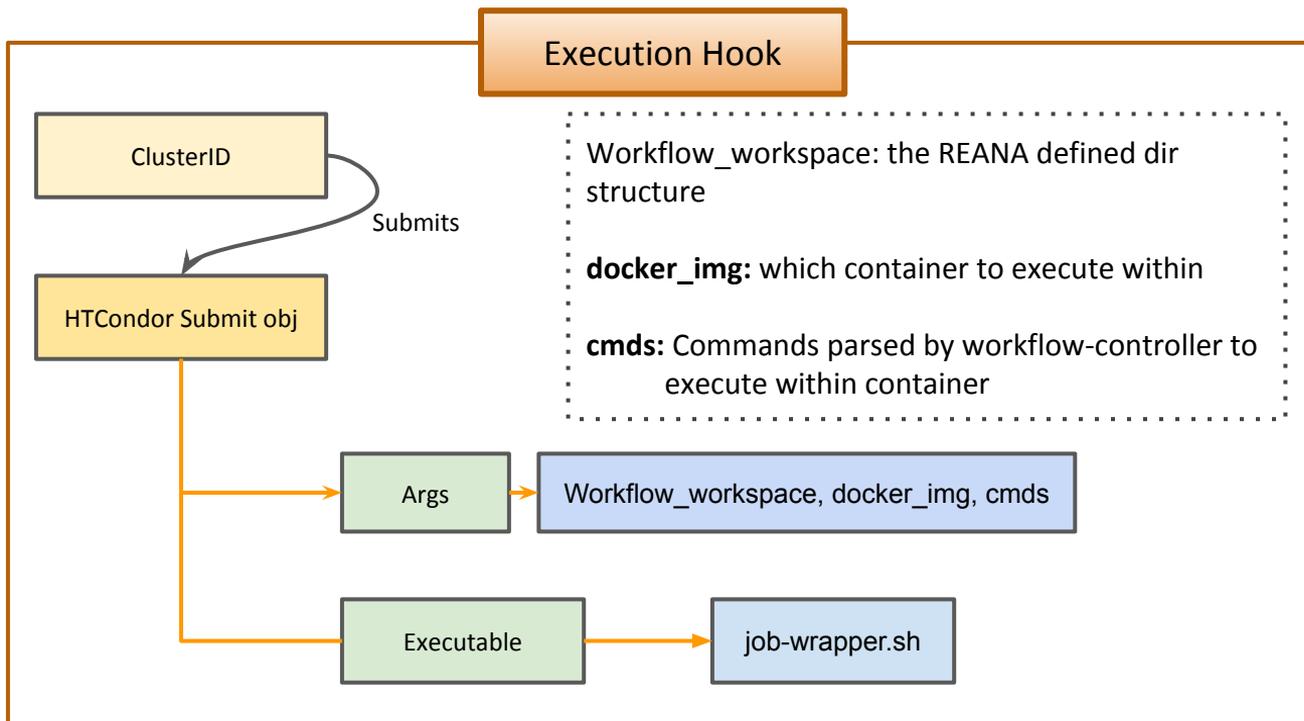
Notre Dame CCTools

- **Chirp:** Integrated with HTCondor. Adding "+WantIOProxy=True" creates a chirp server that can be used between the VC3-headnode and the workers. It also takes care of the authentication.
- **Parrot:** Static version available, runs on any x86 architecture. Can be used to interact with the chirp server created by HTCondor.
 - Note HTCondor has its own chirp client, but it doesn't e.g.: recursive copy directories.

SCAILFIN's REANA modifications

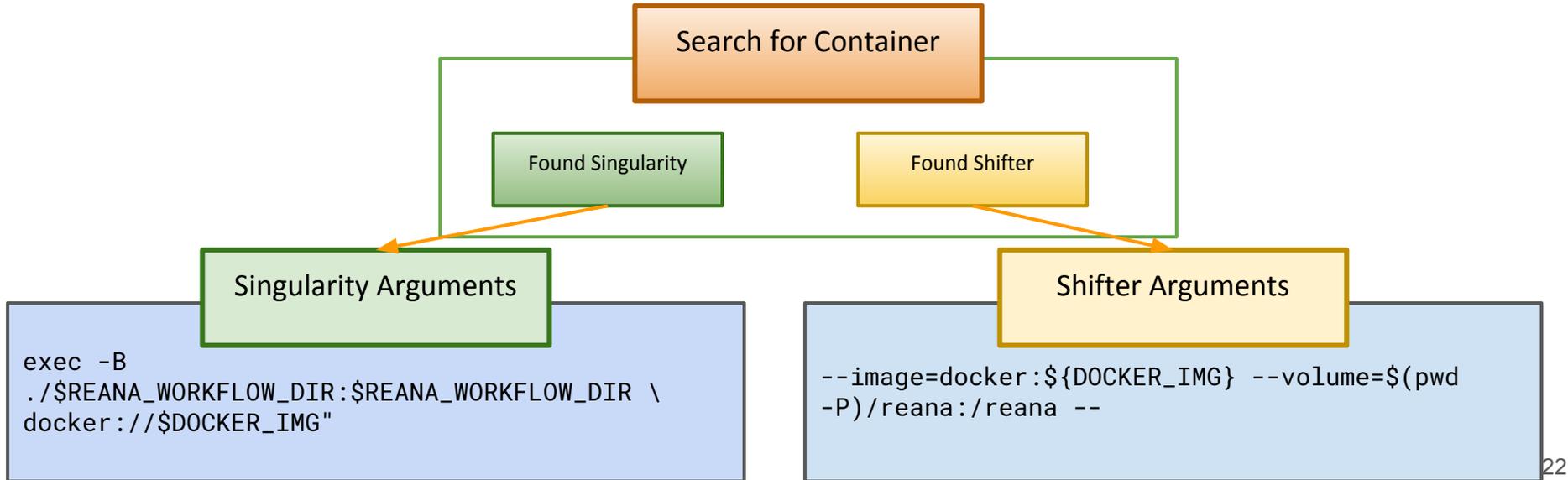
- REANA requires some form of docker supporting container technology
 - Singularity and Shifter support added. Possibly Charlie Cloud
- REANA expects to submit to a kubernetes cluster
 - Added support for VC3 specialized HTCondor submissions through a modified reana-job-controller and a job_wrapper for every workflow step.
 - The modified reana-job-controller submits each workflow step to a local condor scheduler
- Job wrapper
 - Each workflow step is wrapped by a script which searches for container technology and launches each workflow step into the available container (shifter, singularity). Handles initial/final file transfer stages via parrot+chirp.

Container technologies in HPC facilities



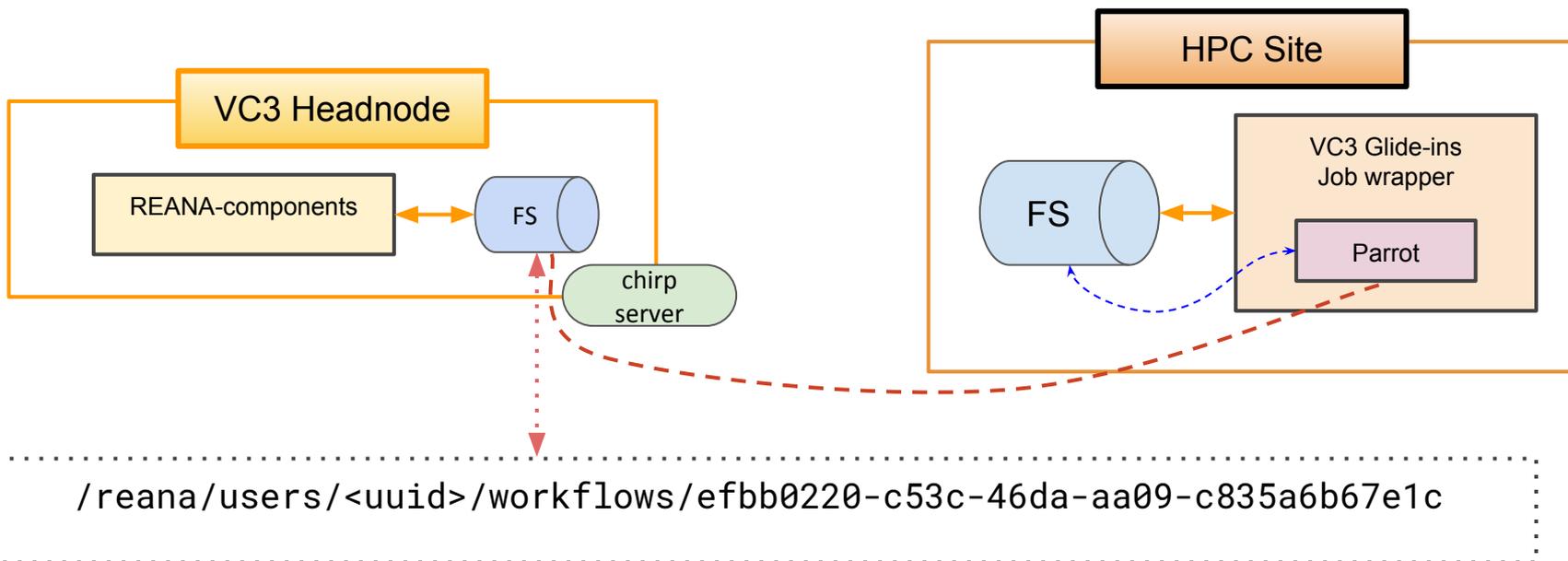
Job-wrapper Container management

- Performs check for available container technologies
 - Checks Binaries in \$PATH (VC3 may auto-load these)
 - Attempts to load modules for Singularity and Shifter (e.g.: module load singularity)
 - Executes workflow-step within discovered container
 - Will choose depending on the currently set \$default



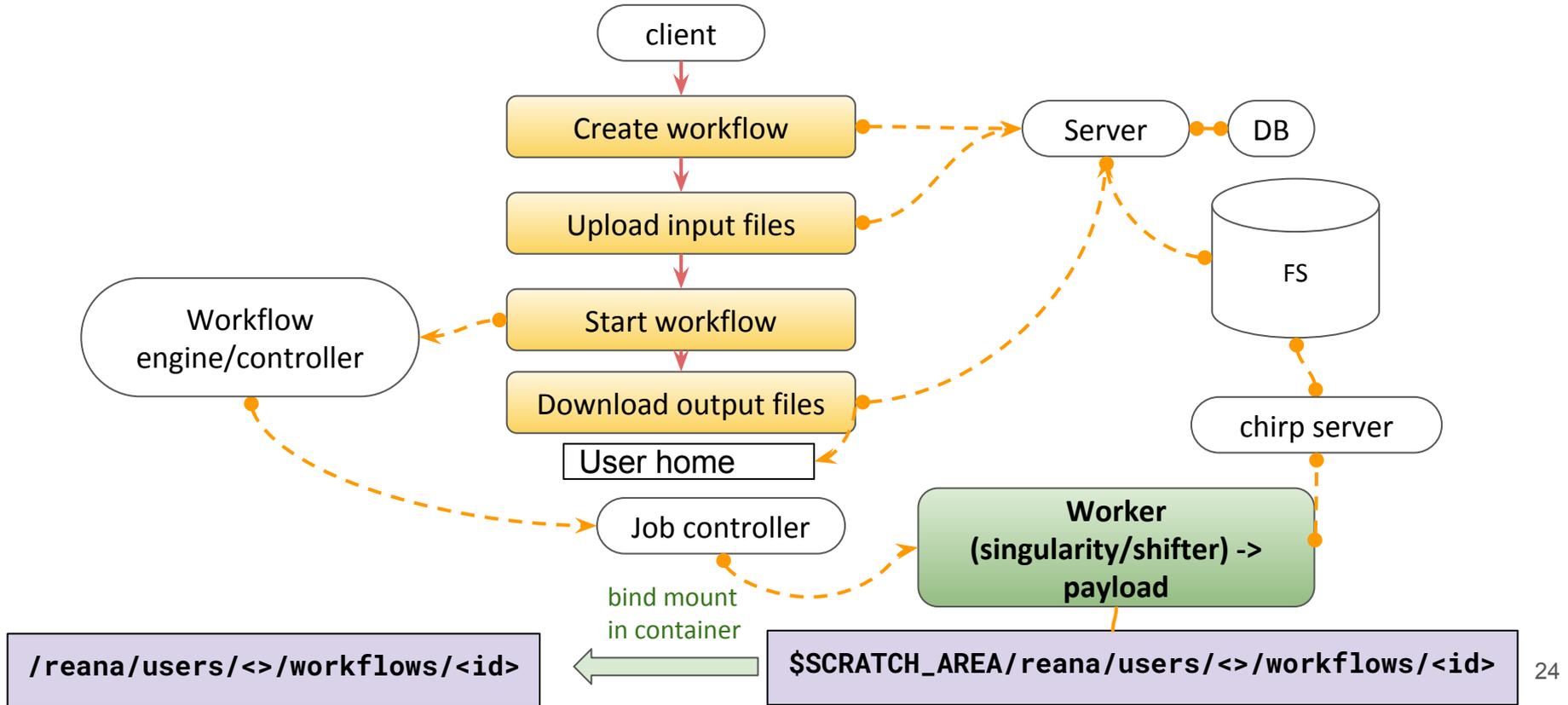
Job-wrapper File management

- REANA expects files to be where the workflow controller placed them
 - We want to simulate this directory structure on the HPC infrastructure
- We can easily transfer directory structures with HTCondor's [chirp protocol](#) and [Parrot](#)

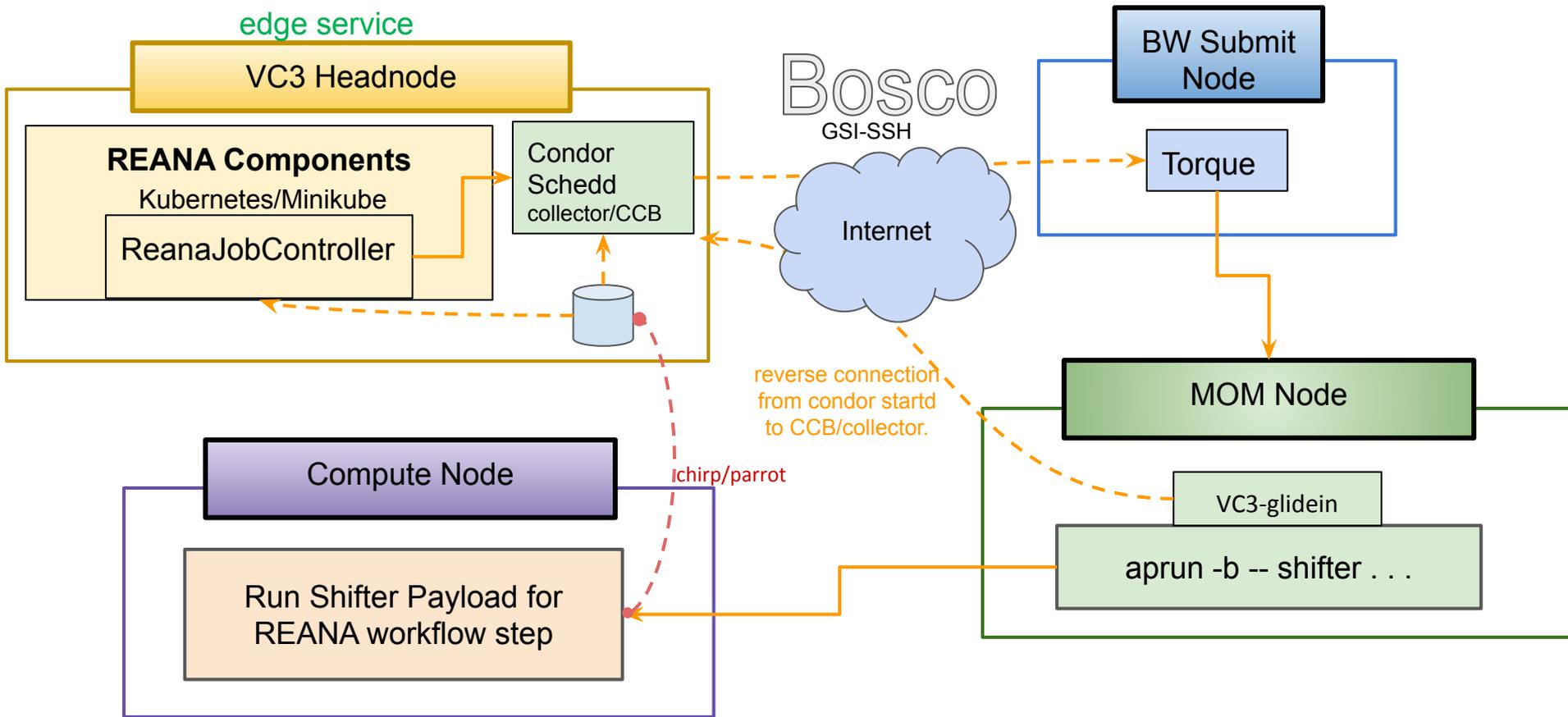


REANA client / User's perspective

Starting a workflow



SCAIFIN on Blue Waters



Blue Waters Tests...

Successfully ran a complex physics test: [BSM search](#)

```
(reana) [khurtado@khurtado-reanabwvc3 condor]$ reana-client status
NAME          RUN_NUMBER  CREATED          STATUS    PROGRESS
demobsmv2    1           2019-06-20T14:24:40 finished  18767/65

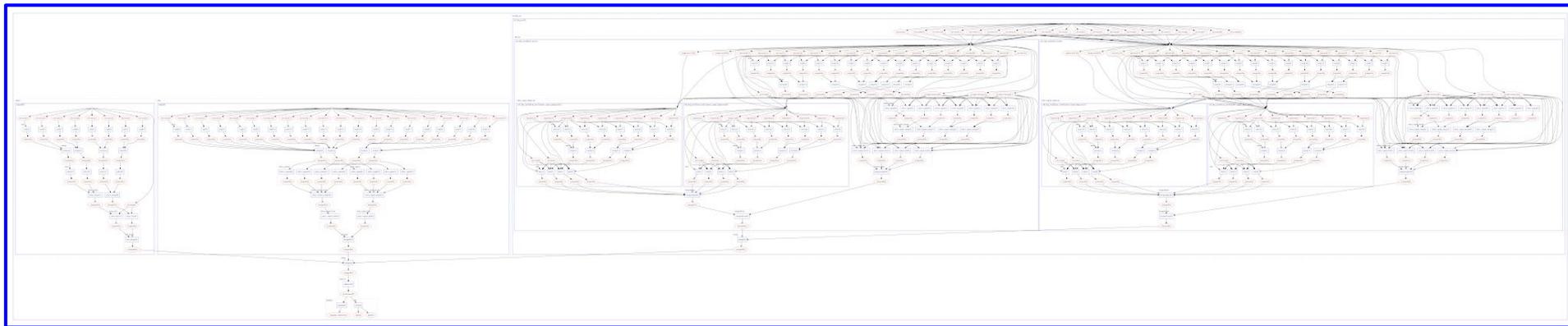
(reana) [khurtado@khurtado-reanabwvc3 condor]$ reana-client ls | grep pdf
plot/prefit.pdf                               19405
2019-06-20T15:31:49
plot/postfit.pdf                              19431
2019-06-20T15:31:49

(reana) [khurtado@khurtado-reanabwvc3 condor]$ condor_history -constraint
'regexp("12610aee-f019-4888-85e5-e9c098d28bf8", Args)' -af:h ClusterId ExitCode LastRemoteHost
ClusterId ExitCode LastRemoteHost
97         0           slot1_1@31738@nid25424
96         0           slot1_1@13090@nid27638
95         0           slot1_1@13090@nid27638
94         0           slot1_1@13090@nid27638
93         0           slot1_1@13090@nid27638
```

Blue Waters Tests...

Successfully ran a complex physics test: [BSM search](#)

Here is the [workflow diagram](#), to give you an idea of the complexity.



Blue Waters Tests...

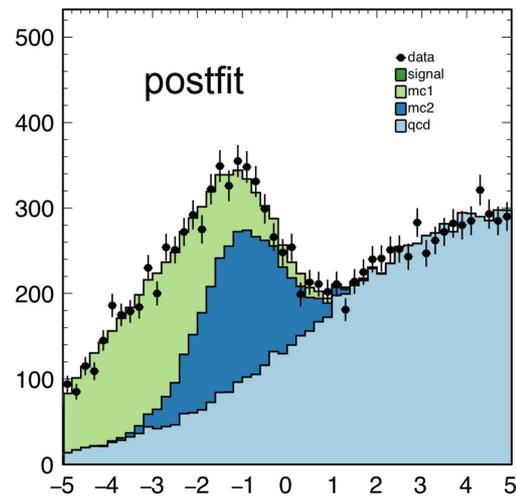
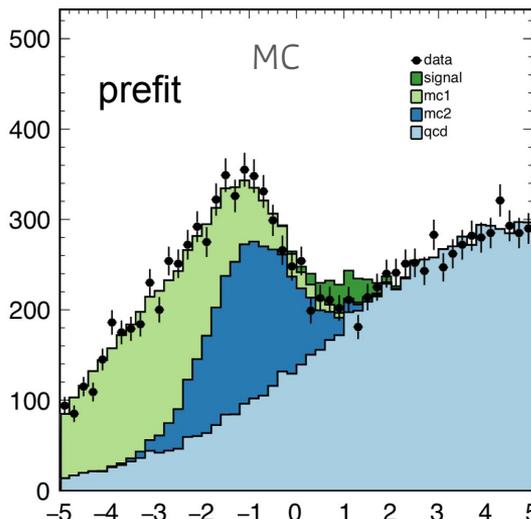
Once results are downloaded through the client, final plots look like this:

```
(reana) [khurtado@khurtado-reanabwvc3 condor]$ reana-client download plot/prefit.pdf
File plot/prefit.pdf downloaded to /home/khurtado/condor.
(reana) [khurtado@khurtado-reanabwvc3 condor]$ reana-client download plot/postfit.pdf
File plot/postfit.pdf downloaded to /home/khurtado/condor.
```

Data is generated/emulated according to Standard Model expectations.

After processing, a statistical model involving both signal and control regions is built and the model is fitted against the observed data.

The signal sample is scaled down significantly to fit the data, which is expected since the data was emulated in accordance with a SM-only scenario



Conclusions, next steps

- Run actual Machine Learning training workflows
- Benchmark performance on variety of HPC systems/sites
- Clean up implementation and documentation
- Integration into central REANA repository at CERN



Supported by NSF Award OAC-1841448

Notre Dame Contacts = Main developers

- Kenyi Hurtado
 - khurtado@nd.edu
- Cody Kankel
 - ckankel@nd.edu

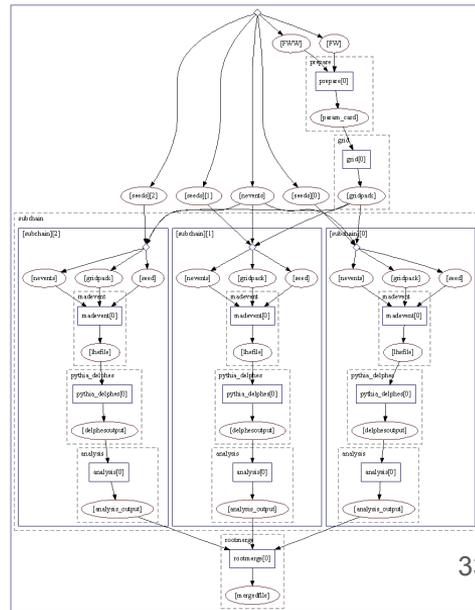
Thanks!

Backup slides

reana: Reproducible Research Data Analysis Platform

Features

- Allows creation of tightly defined, container encapsulated workflows
- Built with commodity pieces
- Purpose is to allow complete reproducibility
- Sharing workflows is as easy as sharing a specification
 - (and inputs!)
- Different workflow engines supported. e.g.:
 - CWL (Common Workflow Language) : <https://www.commonwl.org/>
 - Yadage (YAML based adage): <https://yadage.readthedocs.io>

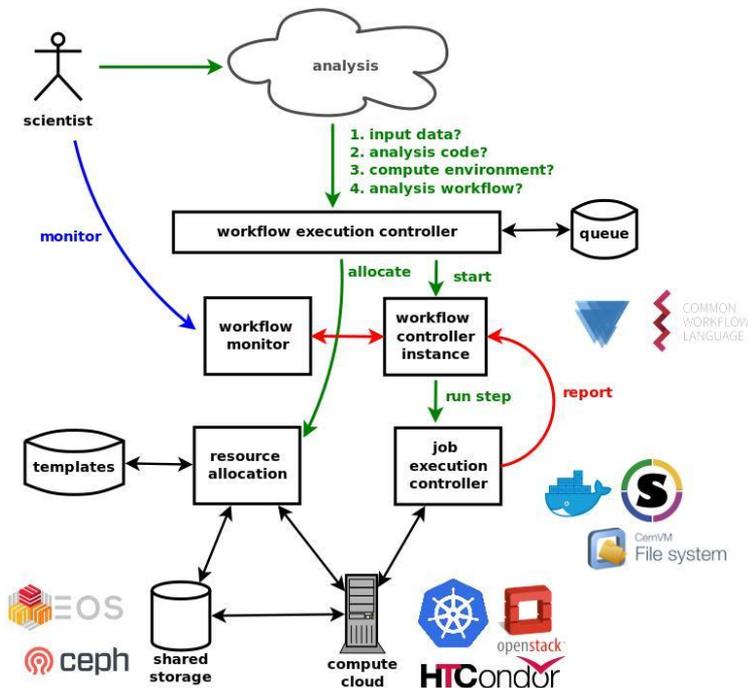


reana: Reproducible Research Data Analysis Platform

Process

1. Create workflow specification (Yadage, CWL, Serial)
2. Upload workflow and inputs to REANA cloud
3. Start workflow
4. Download / pull down results
5. Share workflow specs with others

Architecture

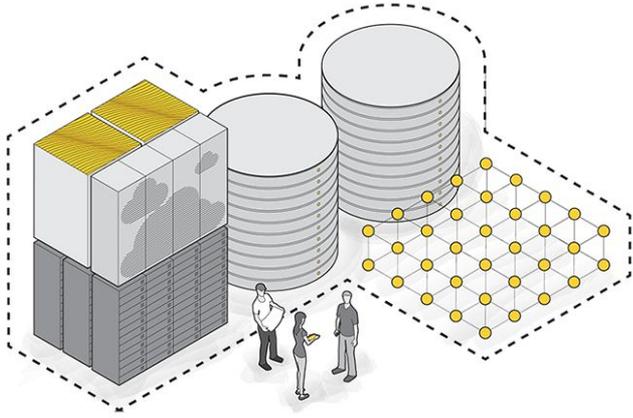


reana: Reproducible Research Data Analysis Platform

Components

- Two major components each consisting of many sub-components
 - **reana-client**: User facing component.
 - Accepts workflows and is used as interface to entire REANA system (for user).
 - **reana-cluster**: Workhorse.
 - Consists of many small pieces which handle workflows, dish out jobs, coordinates results, can be thought of as the job scheduler. Jobs are scheduled via Kubernetes.

VC3: Virtual Clusters for Community Computation



VC3: A platform for provisioning cluster frameworks over heterogeneous resources for collaborative science

- Overlays “cluster” environment on top of diverse resource allocations
- Similar to cloud services that allow you to stand up clusters, but on “your” resources

<https://www.virtualclusters.org>



VC3: Virtual Clusters for Community Computation

VC3 Architecture

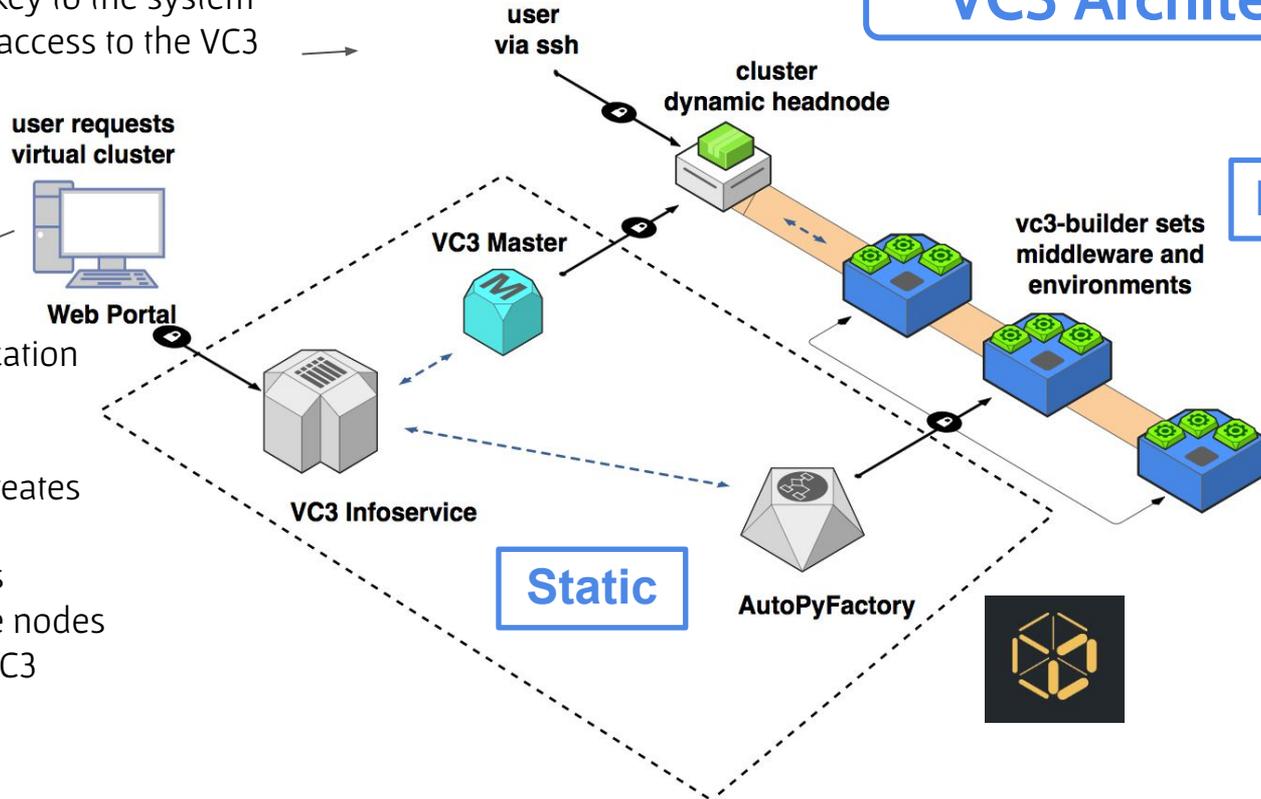
Dynamic

Static

User adds an SSH public key to the system that will be used to grant access to the VC3 dynamic headnode

start here

- User defines an allocation
- Selects middleware configuration
- VC3 infrastructure creates VC3 headnode and configures resources
- Workers on compute nodes communicate with VC3 headnode to receive compute workloads



VC3: Virtual Clusters for Community Computation

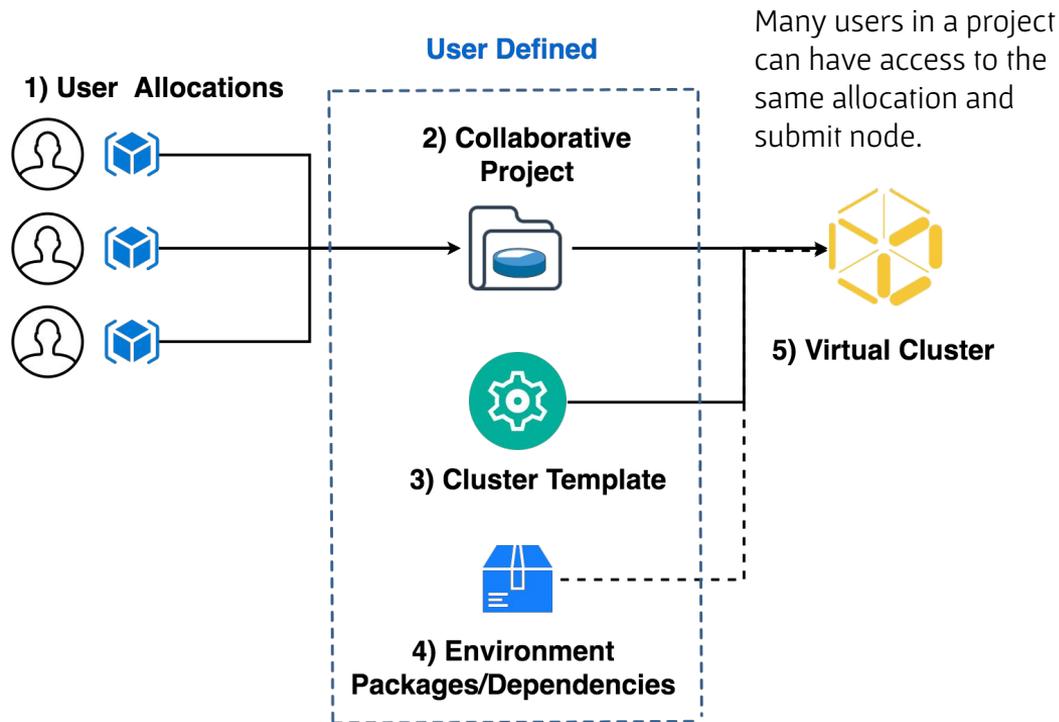
Features

- The user can select its own middleware for submission (E.g.: HTCondor, WorkQueue, Spark, REANA+HTCondor).
- It doesn't matter what the resource target batch system is (as long as it is supported by glite/blah, the translation layer for submission).

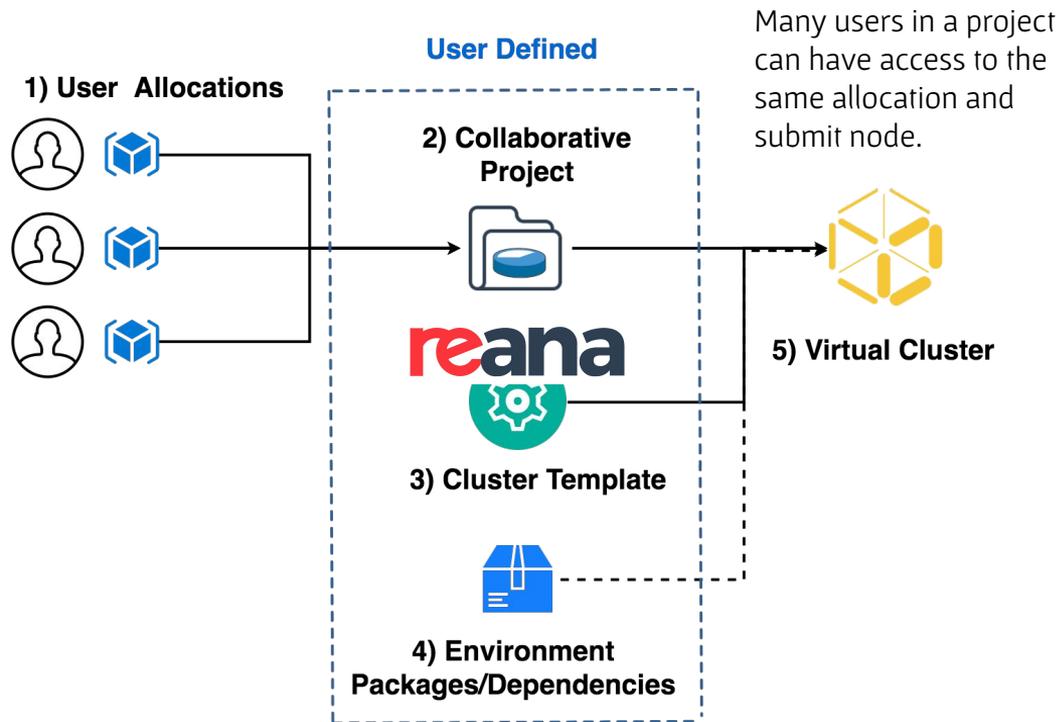
E.g.: Torque (Blue Waters), SLURM (NERSC, PSC-Bridges, Stampede2), HTCondor, LSF, SGE, PBS.



Implementation: A REANA Cluster Template for VC3



Implementation: A REANA Cluster Template for VC3



Using REANA + VC3 on Blue Waters*

Blue Waters cluster:

- Batch system: Torque
- Container technology: Shifter
- Authentication mechanisms:
 - Multi-factor authentication (Password + RSA token)
 - GSI-SSH tokens

Virtual cluster created on top of Blue Waters:

- VC3 Submit node with kubernetes (via minikube) and a REANA cluster deployed on the fly.
- HTCondor as the middleware
- VC3 authenticates with Blue Waters via GSI-SSH

*Note: Infrastructure worked out of the box on other resources such as the ND HPC Cluster and XSEDE/Pittsburgh

SCAILFIN Developments to make this work:

REANA Modifications:

- REANA requires some form of docker supporting container technology
 - Singularity and Shifter support finished.
- REANA expects to submit to a kubernetes cluster
 - Added support for VC3 specialized HTCondor submissions through a modified reana-job-controller and a job_wrapper for every workflow step.
 - The modified reana-job-controller submits each workflow step to a local condor scheduler
- Job Wrapper Auto-detection of container technology for workflow steps. (shifter, singularity)

VC3 Modifications:

- Cluster template for REANA+HTCondor
 - Uses the standard HTCondor template as the base to create a condor pool that sends jobs to HPC resources, translating the job to the corresponding batch system submission syntax via bosco.
 - Deploys Kubernetes via minikube
 - Deploys the REANA cluster and client and set up the environment, so the user can interact with them as soon as the VC3 headnode is created.
- GSI-SSH support
 - The GSI-SSH authentication mechanism was added in the infrastructure, in order to support e.g.P XSEDE HPC centers like Blue Waters, Stampede, NERSC.
 - Proxies can be renewed through the VC3 website for the virtual cluster allocations to remain active.

Allocation authentication mechanisms

The auth mechanisms available in VC3 are:

- SSH keys
- GSI tokens
 - Available on XSEDE HPC Systems
 - Requires renewal. The website shows when the proxy expires and the "Edit Allocation Name" button allows users to change
 - Some HPC centers have procedures to increase the default 12 hours expiration time to e.g.: 10 days (NERSC case, and previously done at BW)

Portal Home / Allocations / khurtado-bluewaters-ncsa

Allocation: khurtado-bluewaters-ncsa

[Edit Allocation Name](#) [Delete Allocation](#)

Ready

Allocation is ready to be used. This allocation may be added to any project in order to launch a Virtual Cluster.

Step 1: Log Into Resource

In a terminal, type:

```
ssh hurtadoa@h2ologin.ncsa.illinois.edu
```

Step 2: Access Resource

Enter your password for `h2ologin.ncsa.illinois.edu` for access

Step 3: Add Allocation SSH Public Key to Resource

Once the SSH key is generated below, click `Copy to`

Owner

Kenyi Hurtado

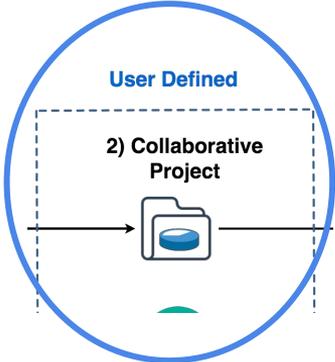
Resource

Blue Waters
Account Name: hurtadoa

Expiration

5 hours, 55 minutes and 49 seconds.

Creating A REANA cluster on Blue Waters



Portal Home / Projects / **bwtest1**

Project: **bwtest1**

Description: None Delete Project

Members **Allocations**

Project Members Filter

| Name | Email | Organization | |
|-----------------------|-----------------|--------------------------|------------------|
| Kenyi Hurtado (Owner) | khurtado@nd.edu | University of Notre Dame | |
| Cody Kankel | ckankel@nd.edu | University of Notre Dame | Rem |

SELECT NEW USERS

Nothing selected

Portal Home / Projects / **bwtest1**

Project: **bwtest1**

Description: None Delete Project

Members **Allocations**

Allocations Filter

| Name | Owner | Organization | |
|--------------------------|---------------|--------------------------|------------------|
| khurtado-bluewaters-ncsa | Kenyi Hurtado | University of Notre Dame | Rem |

SELECT ALLOCATION:

Nothing selected

Add Allocation to Project

45

Creating A REANA cluster on Blue Waters



Cluster Template

Launch New Virtual Cluster

Project: bwtest1

* = INDICATES REQUIRED FIELD

VIRTUAL CLUSTER NAME (A-Z, 0-9, _ AND -)*

reanabwv1

CLUSTER TEMPLATE FRAMEWORK*

REANA+HTCondor

NUMBER OF COMPUTE WORKERS:*

2

ENVIRONMENT

Select Environment

ALLOCATIONS*

khurtado-bluewaters-ncsa

EXPIRATION

If not specified, expiration defaults to 6 hours from launch, when your virtual cluster will automatically be terminated.

HOURS:

98

Cancel

Launch Virtual Cluster

Creating A REANA cluster on Blue Waters

| My Virtual Clusters Filter | | | | | | |
|---|--------------|-----------------|--|--|------------------|--|
| Name | Project | Head Node | Workers | State | Cluster Template | |
| khurtado-ndvc1 | scailfin-dev | 128.135.158.232 | <ul style="list-style-type: none">3 Requested3 Running0 Queued0 Error | <div>Running</div> <p>All requested compute workers are running.</p> | reana+htcondor | |
| khurtado-reanabwv1 | bwtest1 | 128.135.158.188 | <ul style="list-style-type: none">2 Requested2 Running2 Queued0 Error | <div>Running</div> <p>Requesting 0 less compute worker(s).</p> | reana+htcondor | |

End result is a REANA cluster deployed on the VC3 headnode

Components are deployed via Kubernetes (minikube)

```
(reana) [khurtado@khurtado-reanabwv1 ~]$ reana-cluster status
COMPONENT      STATUS
job-controller  Running
server         Running
db             Running
workflow-controller Running
message-broker  Running
REANA cluster is ready.
(reana) [khurtado@khurtado-reanabwv1 ~]$ kubectl get pods
NAME                                                    READY   STATUS    RESTARTS   AGE
batch-serial-7e79ee48-036f-4049-87ee-a3dc66d8a1da-tl7zd 0/1     Completed 0           5h54m
db-69744557df-wg4mt                                     1/1     Running   0           5h55m
job-controller-5c7f4c8b4f-sgnj6                        1/1     Running   0           5h55m
message-broker-b7d66cf55-m9p4n                        1/1     Running   0           5h55m
server-58dc985c77-n2qpn                                2/2     Running   0           5h55m
workflow-controller-668f69d4bc-x62w7                  2/2     Running   0           5h55m
```

Links

- SCAILFIN Source code:

- SCAILFIN's modified RJC

- https://github.com/scailfin/reana-job-controller/tree/job_manager

- REANA

- <https://github.com/reanahub>

- VC3

- <https://github.com/vc3-project>

- Websites

- <https://www.virtualclusters.org>

- <http://www.reanahub.io/>

VC3: Virtual Clusters for Community Computation

Constraints

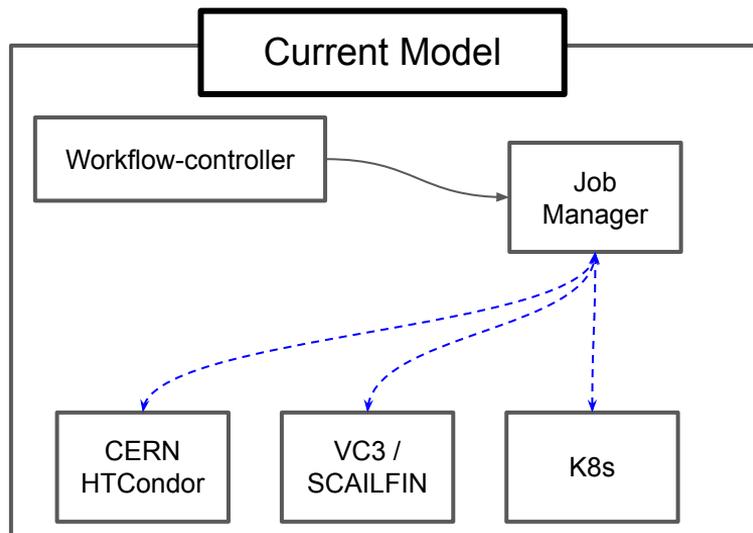
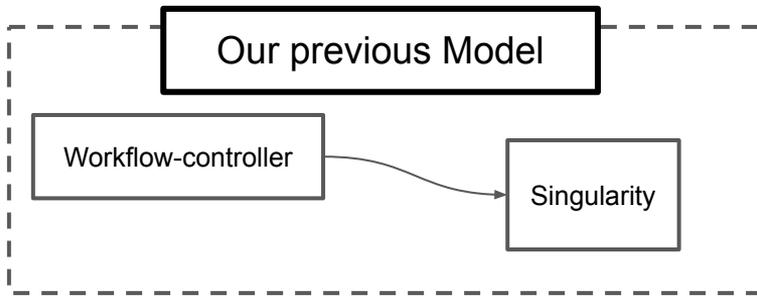
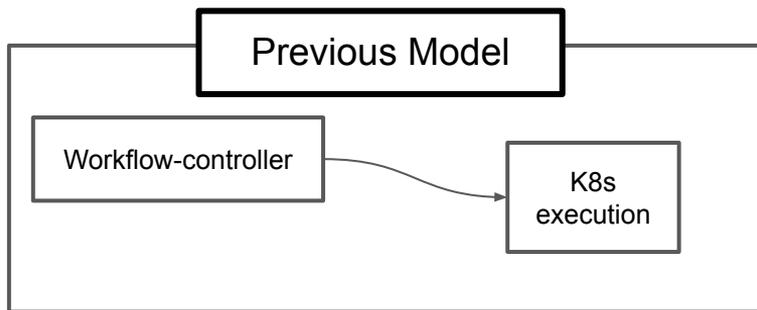
- At present, workers need outgoing network connection for a virtual cluster to work.
 - So, resources like ALCF/Theta are out of the scope with this approach.
 - But e.g.: XSEDE resources like NERSC, Blue Waters, Stampede or PSC-Bridges do meet the outgoing network requirement for example.

SCAILFIN's REANA modifications

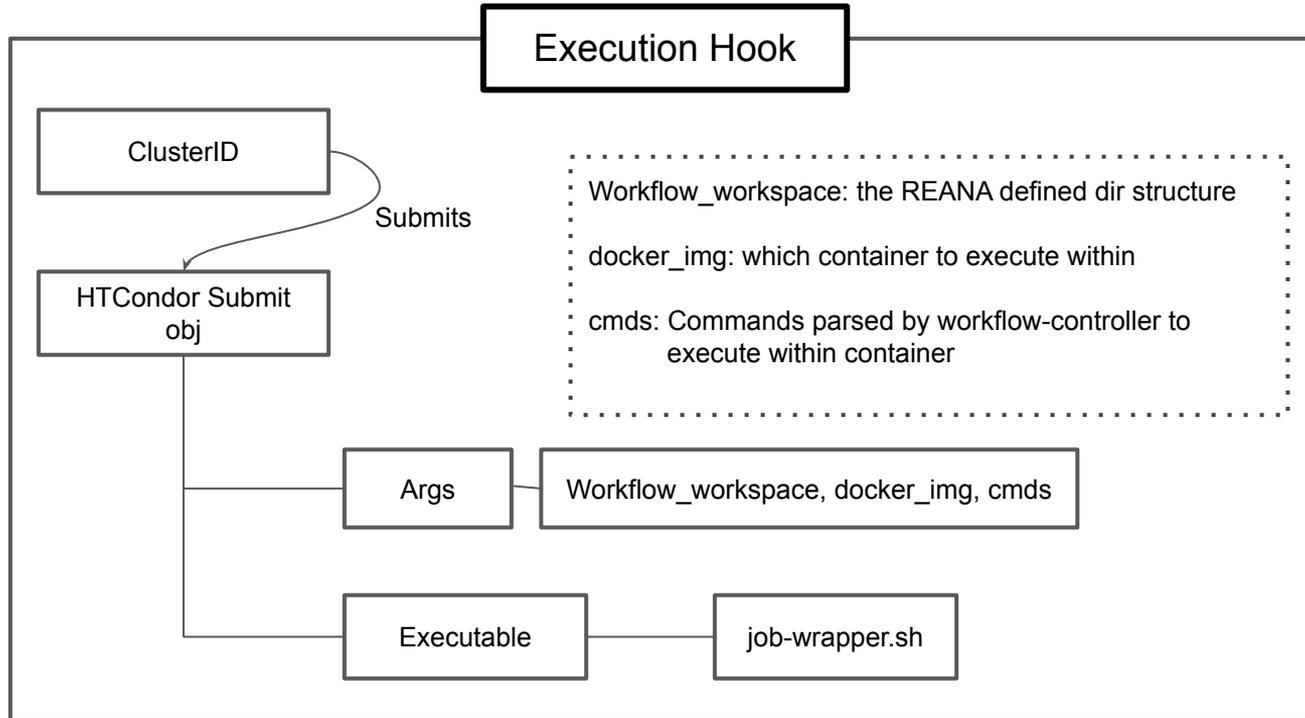
- REANA requires some form of docker supporting container technology
 - Singularity and Shifter support in the works. Possibly CharlieCloud
- REANA expects to submit to a kubernetes cluster
 - Added support for VC3 specialized HTCondor submissions through a modified reana-job-controller and a job_wrapper for every workflow step.
 - The modified reana-job-controller submits each workflow step to a local condor scheduler
- Job wrapper
 - Each workflow step is wrapped by a script which searches for container technology and launches each workflow step into the available container (shifter, singularity)

SCAILFIN's REANA modifications

- REANA developers allowed abstraction of reana-job-controller's backend
 - Allows for “plug and play” of backends / job execution component

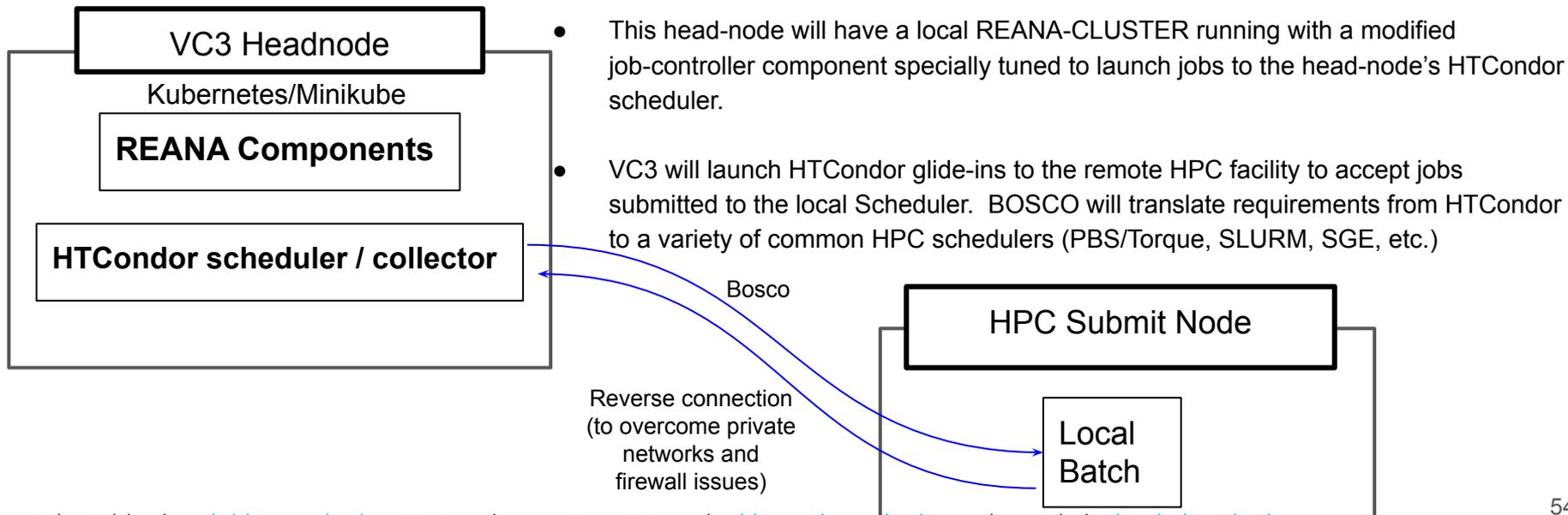


SCAILFIN's REANA modifications



SCAILFIN and VC3

VC3 is utilized for remote connections to clusters.



SCALFIN's VC3 modifications

- Cluster template for REANA+HTCondor
 - Uses the standard HTCondor template as the base to create a condor pool that sends jobs to HPC resources, translating the job to the corresponding batch system submission syntax via bosco.
 - Deploys Kubernetes via minikube
 - Deploys the REANA cluster and client and set up the environment, so the user can interact with them as soon as the VC3 headnode is created.
- GSI-SSH support
 - The GSI-SSH authentication mechanism was added in the infrastructure, in order to support e.g. XSEDE HPC centers like Blue Waters, Stampede, NERSC.
 - Proxies can be renewed through the VC3 website for the virtual cluster allocations to remain active.
- Additional patches to support Cray Linux environments (NERSC, Blue Waters)