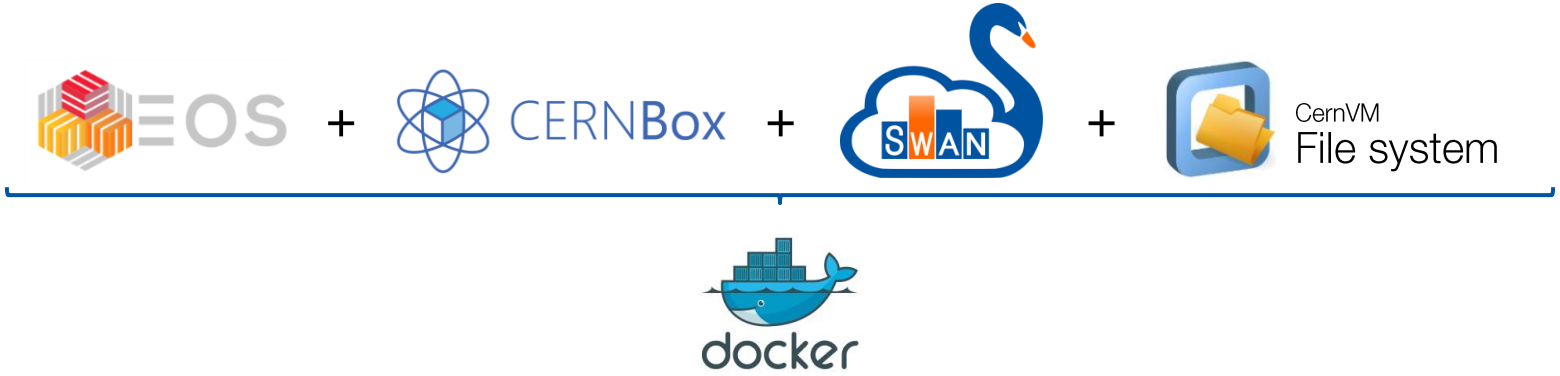
ScienceBox 2.0

Enrico Bocchi
CERN IT, Storage Group

ScienceBox



- **EOS:** Storage backbone for LHC + Physics data, and CERNBox
- **CERNBox:** Sync&Share for Personal and Project Files
- **SWAN:** Data Analysis Platform with Interactive Jupyter Notebooks
- **CVMFS:** Software stacks for LHC experiments and scientific analysis

ScienceBox – Raison d'être

- Facilitate distribution of successful technology operated at CERN
  - Scalable storage, Sync & Share, Integrated Analysis Platforms, …
  - High Energy Physics sites, NRENs, EU-project collaborators, partnering institutions

- Increasing interest in Data Management and Analysis tools for Open Science
  - 2PB of particle physics data
    and tools to explore them → http://opendata.cern.ch/

- Future opportunities for broader adoption
  - ScienceMesh interest in services beyond EFSS
  - Worldwide LHC Computing Grid Tier-2 sites

ScienceBox Timeline

2017

ScienceBox Project Epiphany

- First ever replica of CERN production services in containers
- Automated deployment in Docker Compose, single-host

2018

2020

2021

ScienceBox Timeline

2017

ScienceBox Project Epiphany

ScienceBox for Education

- Scalable deployment on Kuberentes-managed clusters
- Resources describe in k8s YAMLs

2018

2020

➢ Deployed for EU-project **Up2U** at PSNC and CERN

Up To University

Students in high-schools to adopt tools used in science:

- **CERNBox** – Access and share content from any device
- **SWAN** – Full analysis platform in a web browser

2021

ScienceBox Timeline

2017

- ScienceBox Project Epiphany
- ScienceBox for Education

2018

- ScienceBox for High Energy Physics

2020

2021

- Deployed on Helix Nebula Cloud – 2000+ CPUs, 10+ TB memory
- Addition of Spark cluster for massive computations

"Big Data Tools and Cloud Services for High Energy Physics Analysis in TOTEM Experiment" → https://ieeexplore.ieee.org/document/8605741

ScienceBox Timeline

2017

ScienceBox Project Epiphany

ScienceBox for Education

2018

2020

ScienceBox in response to Covid19 Pandemic

2021

- Free-access remote-learning platforms for EU students
- Hosted on AWS, funded by GÉANT

ScienceBox – Use Cases and Technology

Use Cases

TOTEM

open UP2U

CERN AGAINST COVID-19

UP2U Up To University

ScienceBox

CERNBox

CernVM File system

EOS

SWAN

Infrastructure

amazon web services

openstack.

HELIX NEBULA THESCIENCECLOUD

T · · Systems ·

Sites

aarnet

CERN

Joint Research Centre JRC

Technology

kubernetes

docker

HELM

2021 – ScienceBox Reboot

- **Goals of Reboot:** 1. Use modern, widely-adopted container technologies, 2. Improve maintainability, 3. Ease contributions to the package

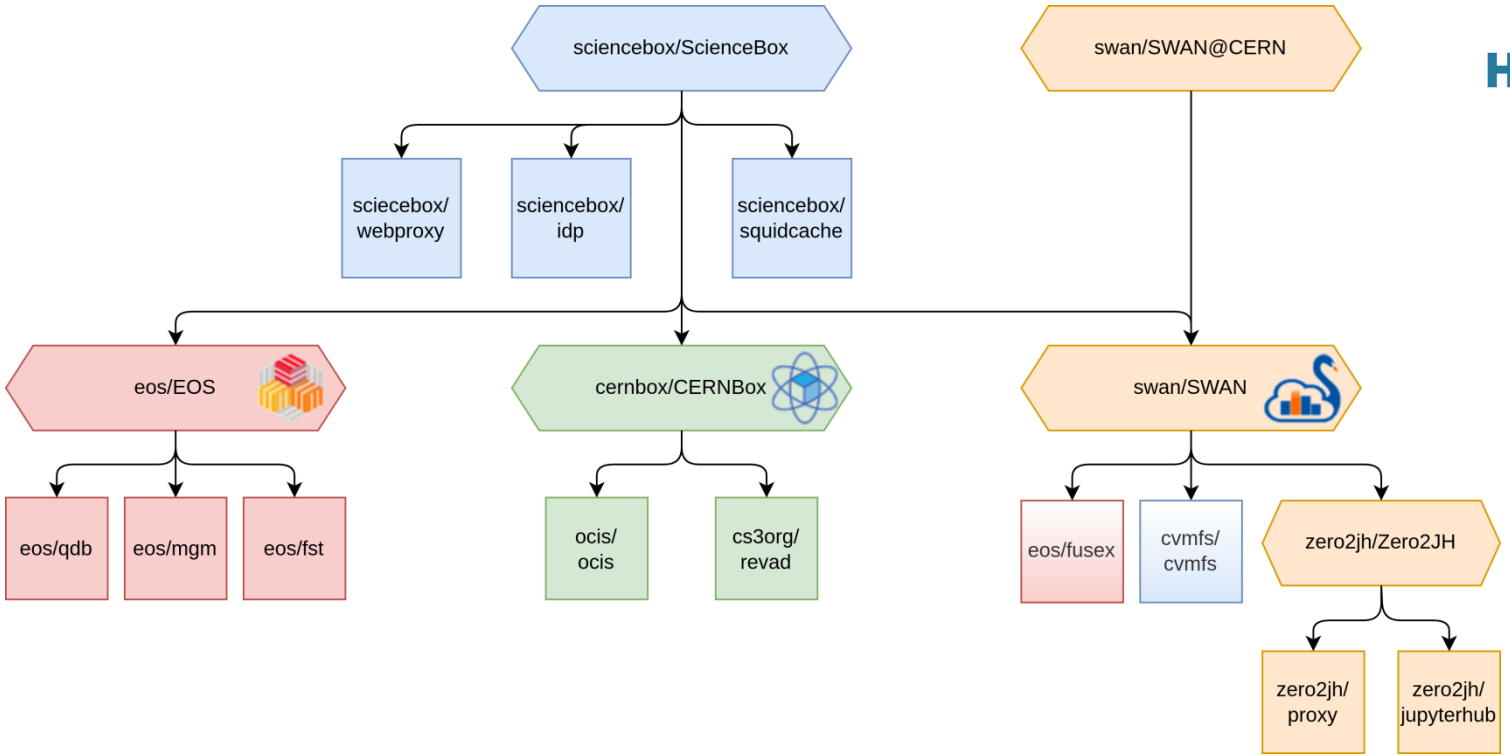
ScienceBox 2.0

- **Goals of Reboot:** 1. Use modern, widely-adopted container technologies, 2. Improve maintainability, 3. Ease contributions to the package

- **Maintainability**
  - Align and keep in sync ScienceBox with CERN production
    - ✓ Improvements and new features at CERN immediately available to ScienceBox
  - Consolidate containerization efforts at CERN into **Helm charts**

  ✅ ScienceBox described as a hierarchical collection of charts
    - ✓ Re-use charts developed and maintained by EOS, CERNBox, SWAN, CVMFS
    - ✓ Add the glue for stand-alone deployments

ScienceBox 2.0 – Helm Charts to the Rescue

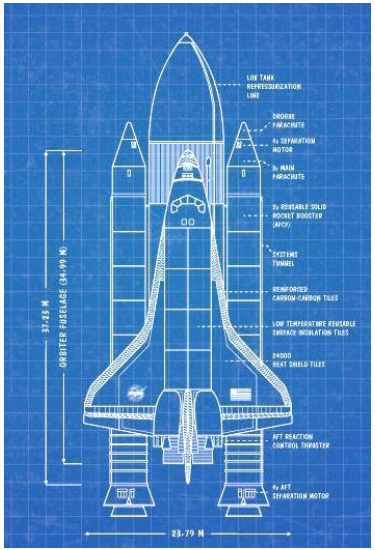


ScienceBox 2.0

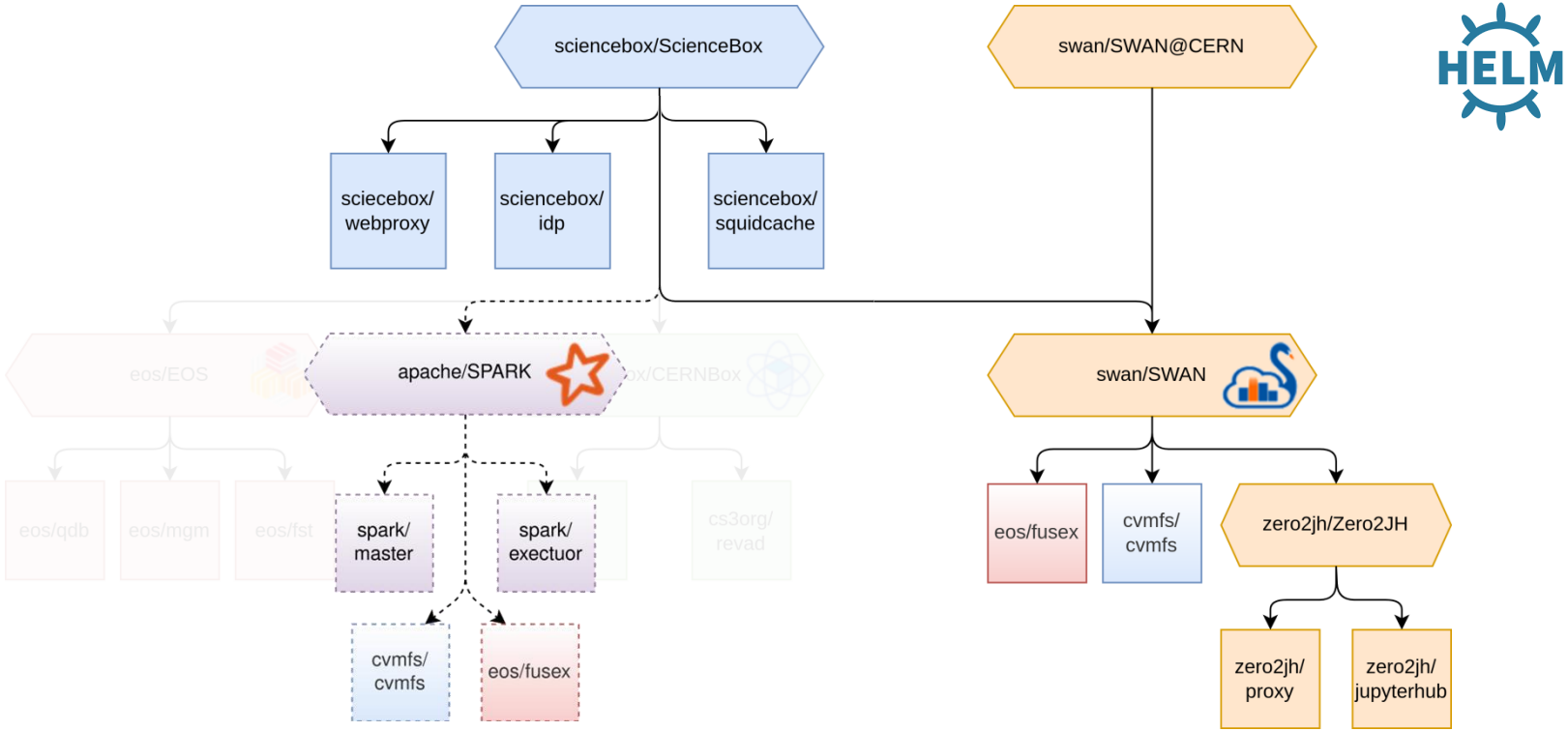
- **Goals of Reboot:** 1. Use modern, widely-adopted container technologies, 2. Improve maintainability, 3. Ease contributions to the package

- **Modularity and Ease to contribute**
  - Allow for deployment of single components, e.g., EOS
  - Facilitate addition and integration of other services

  - ✔ Each chart is a blueprint of service interfaces with own lifecycle and release process
  - ✔ New services can be packaged and added to ScienceBox by expressing a dependency on their charts



ScienceBox 2.0 – Modular Architecture



ScienceBox 2.0 – Where are we now?

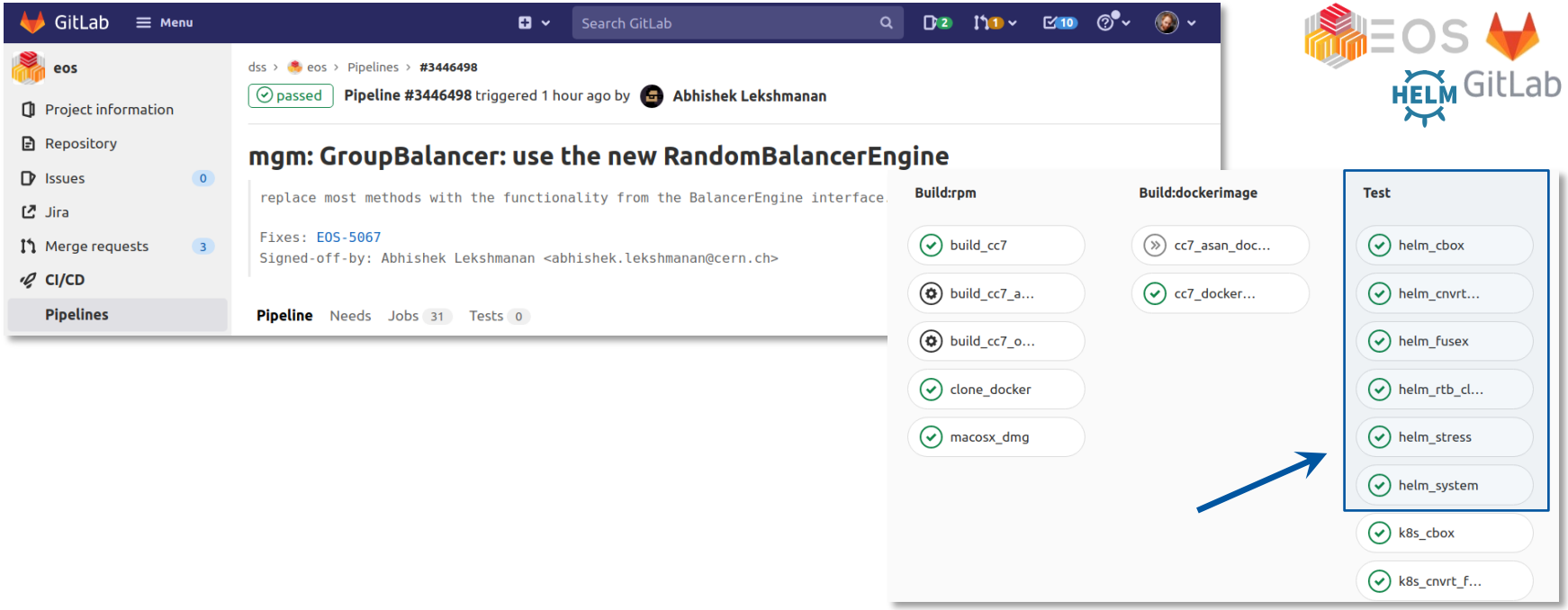
- ✅ EOS charts ready → https://github.com/cern-eos/eos-charts
- ✅ SWAN charts ready → https://github.com/swan-cern/swan-charts
- ✅ ScienceBox glue (IDP, LDAP, extra config, …) ready → https://github.com/sciencebox/charts

---

- ⚙ CERNBox integration → Ongoing
- ⚙ Get Started guide and Documentation → Ongoing
- ⚙ Validation tools, self-testing, multiple OS support → To start

ScienceBox 2.0 – Road Test

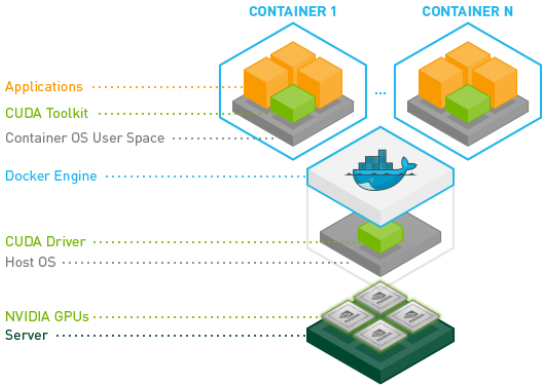
1. EOS Helm charts actively used for testing commits and new releases

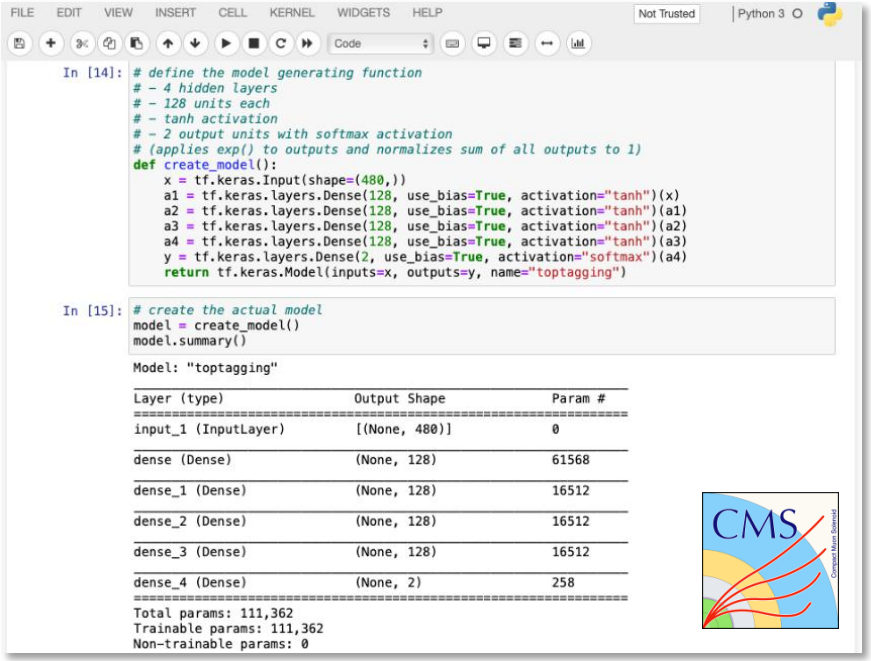


ScienceBox 2.0 – Road Test

1. EOS Helm charts actively used for testing commits and new releases

2. CMS Machine Learning on GPUs
   - SWAN + EOS deployed on AWS EKS
   - NVidia Tesla V100 GPUs
     - ✓ On-demand, dynamically-scalable





FILE EDIT VIEW INSERT CELL KERNEL WIDGETS HELP Not Trusted Python 3

Code

In [14]:
```
# define the model generating function
# - 4 hidden layers
# - 128 units each
# - tanh activation
# - 2 output units with softmax activation
# (applies exp() to outputs and normalizes sum of all outputs to 1)
def create_model():
    x = tf.keras.Input(shape=(480,))
    a1 = tf.keras.layers.Dense(128, use_bias=True, activation="tanh")(x)
    a2 = tf.keras.layers.Dense(128, use_bias=True, activation="tanh")(a1)
    a3 = tf.keras.layers.Dense(128, use_bias=True, activation="tanh")(a2)
    a4 = tf.keras.layers.Dense(128, use_bias=True, activation="tanh")(a3)
    y = tf.keras.layers.Dense(2, use_bias=True, activation="softmax")(a4)
    return tf.keras.Model(inputs=x, outputs=y, name="toptagging")
```

In [15]:
```
# create the actual model
model = create_model()
model.summary()
```

```
Model: "toptagging"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         [(None, 480)]             0
_________________________________________________________________
dense (Dense)                (None, 128)               61568
_________________________________________________________________
dense_1 (Dense)              (None, 128)               16512
_________________________________________________________________
dense_2 (Dense)              (None, 128)               16512
_________________________________________________________________
dense_3 (Dense)              (None, 128)               16512
_________________________________________________________________
dense_4 (Dense)              (None, 2)                 258
=================================================================
Total params: 111,362
Trainable params: 111,362
Non-trainable params: 0
```

Where to Find ScienceBox

- **ScienceBox**
  - https://sciencebox.web.cern.ch/
  - sciencebox-talk@cern.ch

Testing, Contributions, Comments/Discussion are very welcome!

- **Code repositories**
  - ScienceBox Organization on GitHub – https://github.com/sciencebox/
  - Minikube-based deployment – https://github.com/sciencebox/mboxed

- **More on ScienceBox services**
  - {eos,cernbox,swan,cvmfs}.web.cern.ch

Where to Find ScienceBox

- ScienceBox
  - https://scier
  - sciencebox-
- Code reposi
  - ScienceBo cebox/
  - Minikube-b nboxed
- More on Sc
  - {eos,cernbo

ontributions,
/Discussion
welcome!

Samuel, Artiz, Fabio, Abhishek, Diogo, Riccardo, Krishnan

Thank you!

ScienceBox 2.0

Enrico Bocchi
enrico.bocchi@cern.ch

CERN

Backup Slides

Why ScienceBox

- Growing interest in CERN cloud software from external collaborators
  - High Energy Physics sites
  - National Research and Education Networks
  - European projects collaborators

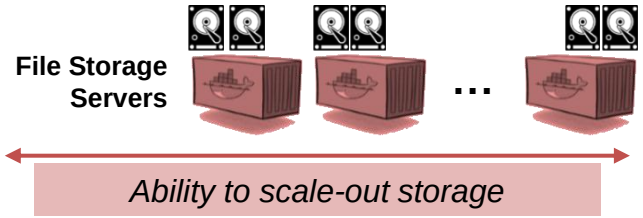
- Facilitate distribution outside CERN
  - Simplified installation leveraging on container technologies
  - Flexible and scalable deployment with container orchestration

- Disposable deployment for development at CERN
  - Software updates, new functionalities, …

ScienceBox Scalability

- Kubernetes: Deploy, orchestrate, and manage containers in a cluster
- It provides means to horizontally scale applications
  - ✓ Deployment, StatefulSet, Horizontal Pod Autoscaler, LoadBalancer on Services, …

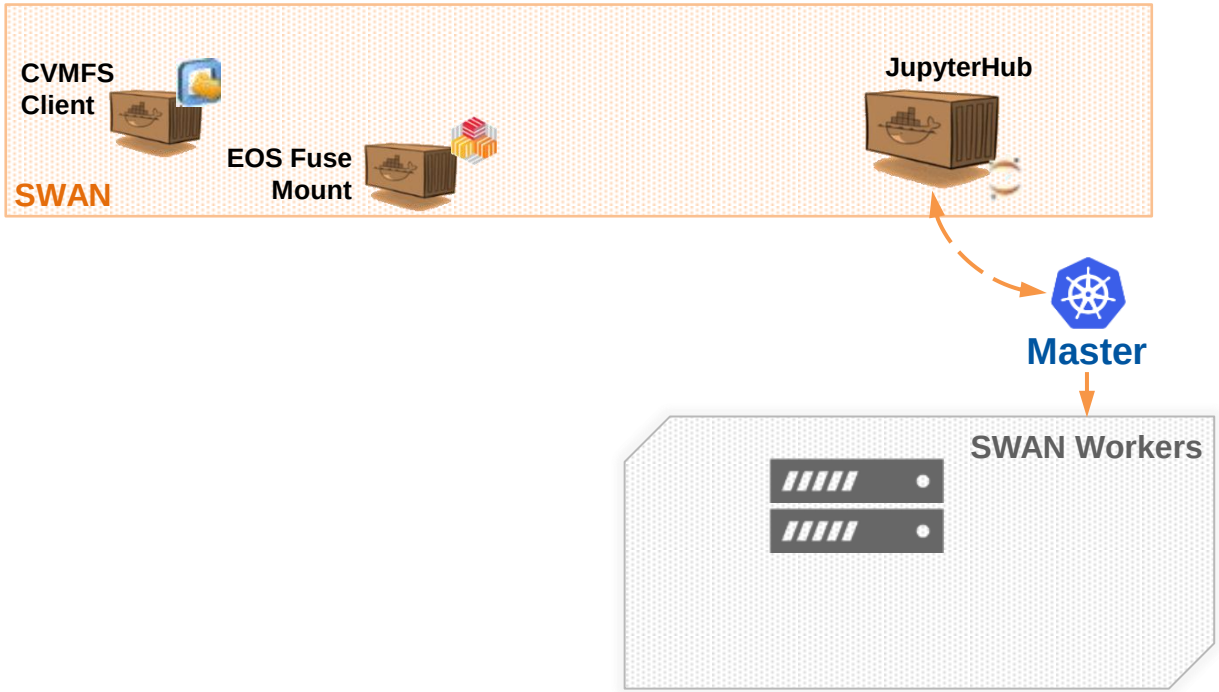
- Storage – Extend EOS capacity
  - ✓ Add machines with additional storage
  - ✓ Replicate File Storage Server containers



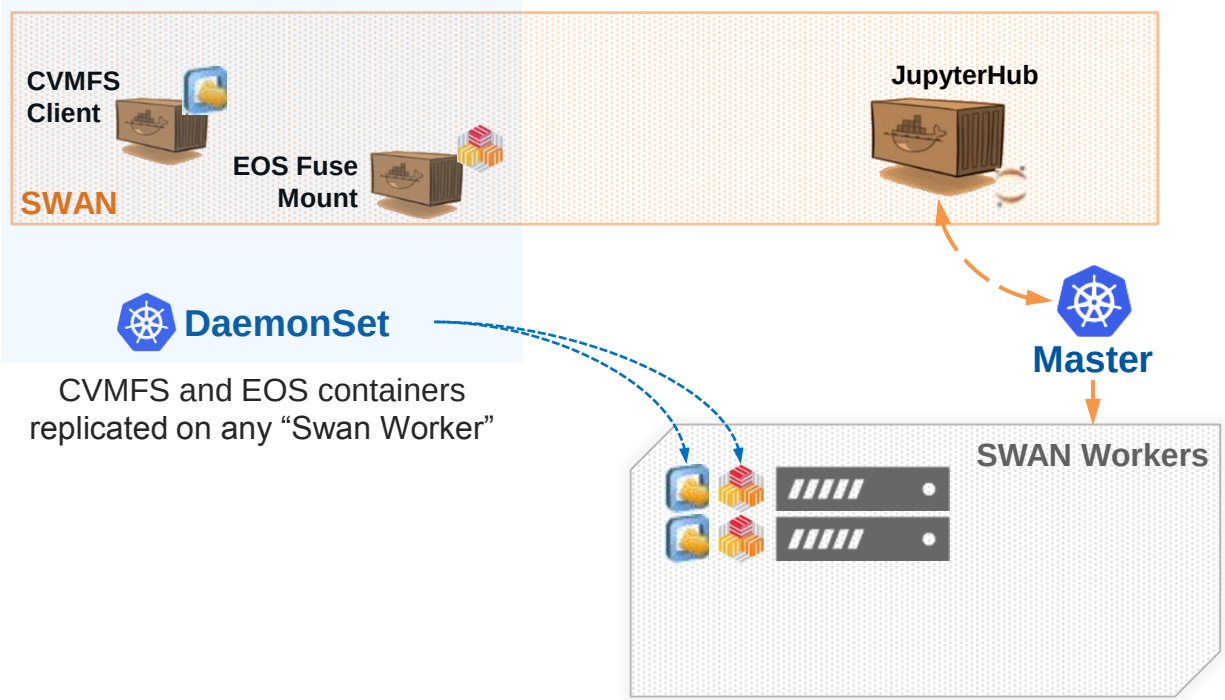
- Computing – Sustain concurrent SWAN sessions
  - ✓ Need of multiple cluster nodes where to spawn Single-user Jupyter Servers
  - ✓ Replicate EOS and CVMFS containers for SWAN sessions

Elastic resources for SWAN

Elastic resources for SWAN

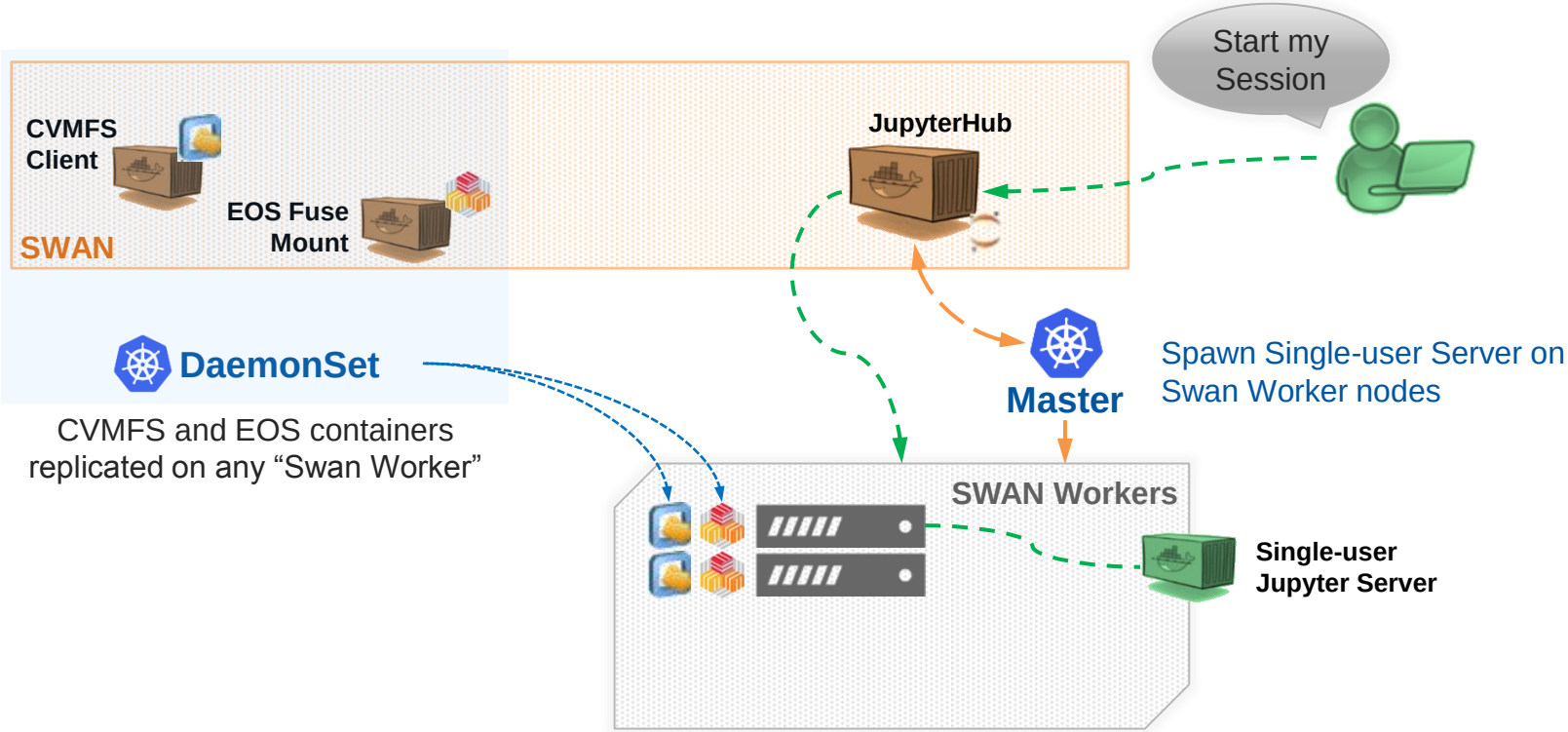


Elastic resources for SWAN

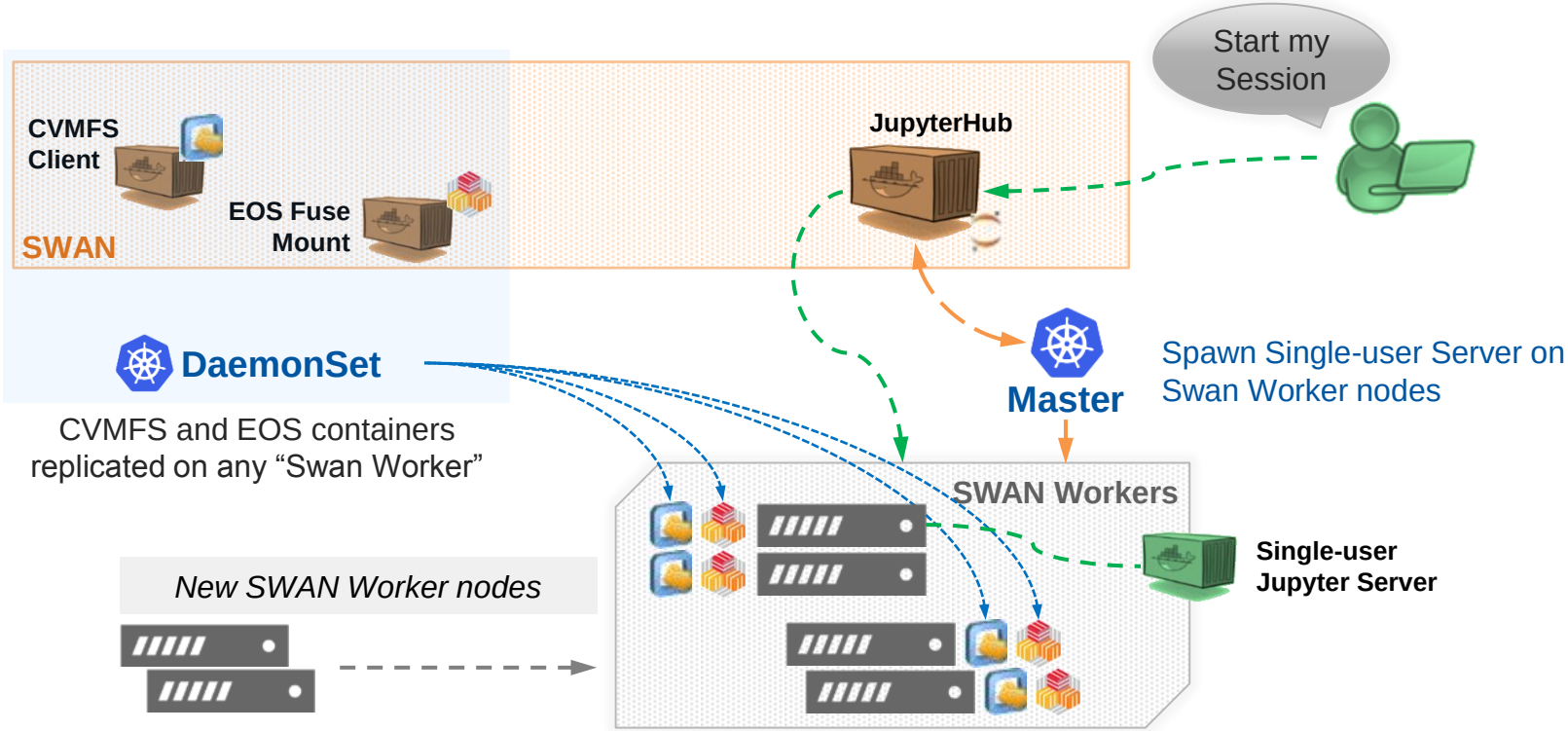


CVMFS and EOS containers replicated on any “Swan Worker”

Elastic resources for SWAN



Elastic resources for SWAN



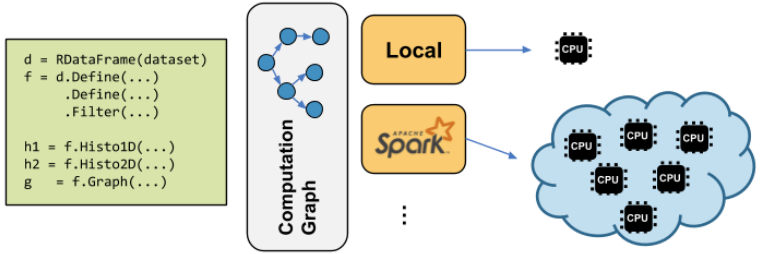
TOTEM Analysis on Commercial Cloud

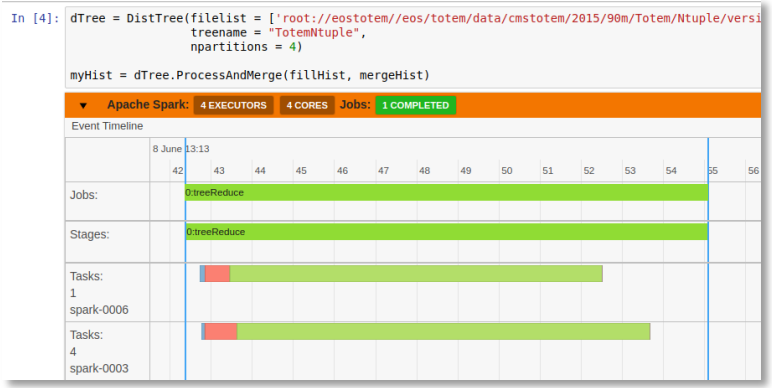
- RDataFrame
  - Interface for declarative analysis, introduced in ROOT 6.14 (2018-07-27)
  - Implicit parallelization
  - Better utilization of multicore resources

```
ROOT::EnableImplicitMT();                          Run a parallel analysis
ROOT::RDataFrame df(dataset);                      on this (ROOT, CSV, ...) dataset
auto df2 = df.Filter("x > 0")                      only accept events for which x > 0
              .Define("r2", "x*x + y*y");          define r2 = x² + y²
auto rHist = df2.Histo1D("r2");                    plot r2 for events that pass the cut
df2.Snapshot("newtree", "out.root");               write the skimmed data and r2
                                                   to a new ROOT file
```

TOTEM Analysis on Commercial Cloud

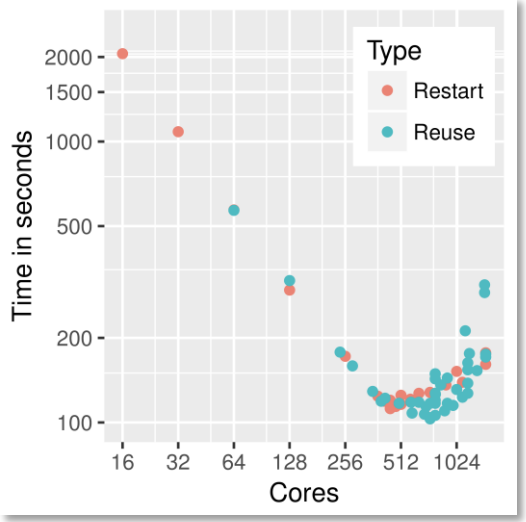
- Allow interactive analysis with ROOT RDataFrame + SWAN + Spark
  - RDataFrame: Interface for declarative analysis with implicit parallelism
  - Use Spark cluster with no changes to the code
  - Monitor Spark jobs from SWAN





TOTEM Analysis on Commercial Cloud

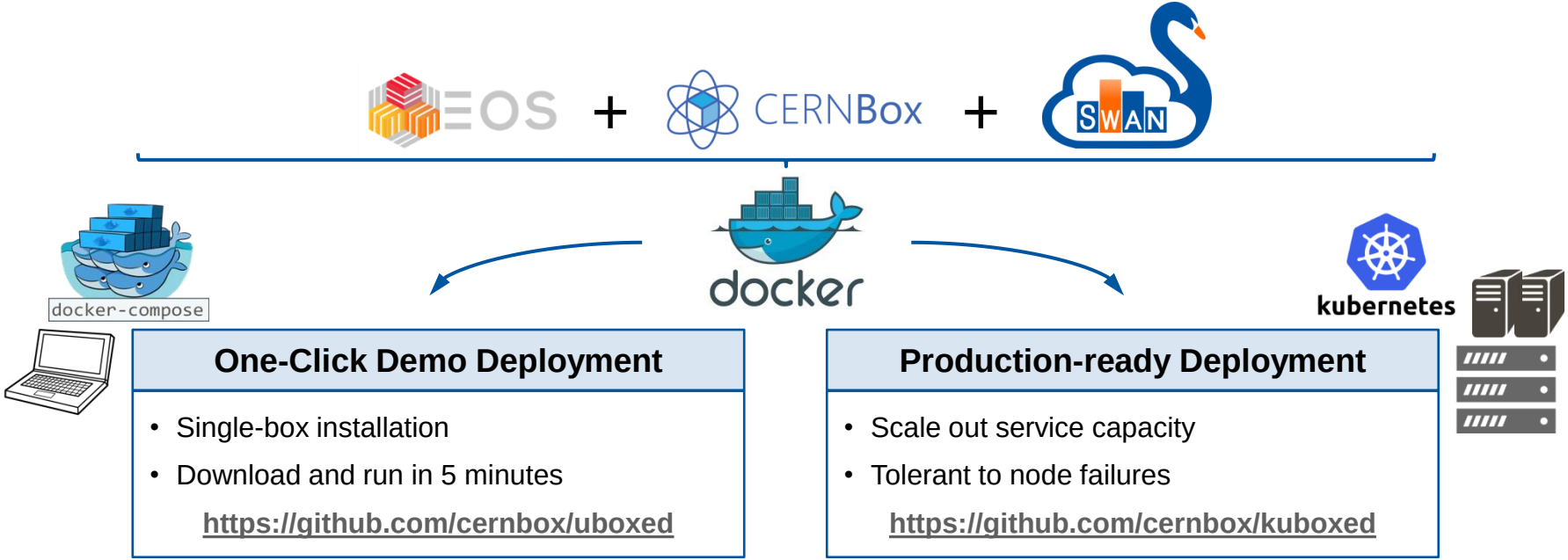
- TOTEM Analysis Dataset:
  - 4.7 TB, 1153 files, 2.8B events
  - Imported via xrootd, results synchronized with CERNBox
- Reduced processing time
  - Wall-clock down to ~2m
  - Optimal at ~750 cores
- Validated Physics Results



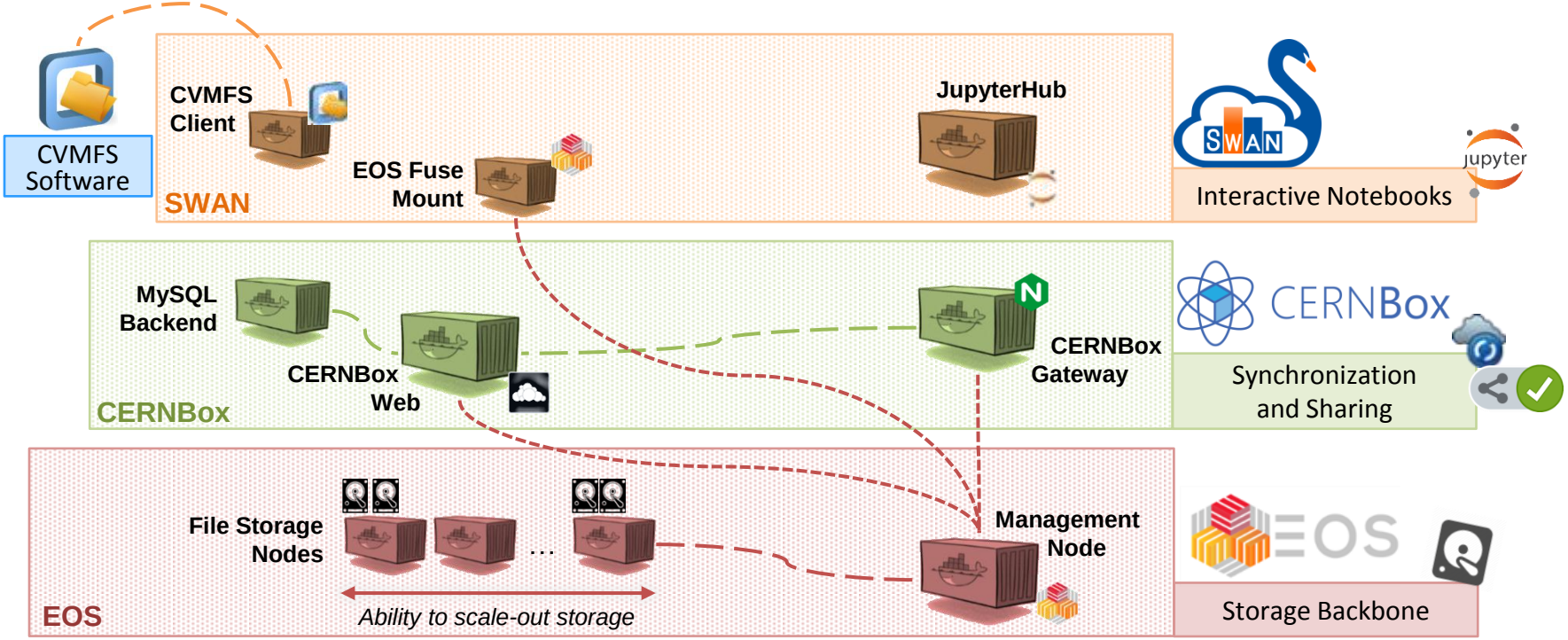
Big Data Tools and Cloud Services for High Energy Physics Analysis in TOTEM Experiment - V. Avati et al.

https://ieeexplore.ieee.org/document/8605741

ScienceBox


EOS
+
CERNBox
+
SWAN
docker-compose
docker
kubernetes
One-Click Demo Deployment
• Single-box installation
• Download and run in 5 minutes
https://github.com/cernbox/uboxed
Production-ready Deployment
• Scale out service capacity
• Tolerant to node failures
https://github.com/cernbox/kuboxed

ScienceBox Architecture



ScienceBox Timeline

2017

ScienceBox Project Epiphany

- Dockerfiles, container images, configuration scripting
- Automated deployment in Docker Compose, single-host
- First ever replica of CERN production services in containers

2018

2020

2021

ScienceBox Timeline

2017

ScienceBox Project Epiphany

ScienceBox in Kubernetes for Education

- Scalable deployment on Kuberentes-managed clusters
- Resources describe in k8s YAMLs

2018

2020

2021

ScienceBox Timeline

2017

ScienceBox Project Epiphany

ScienceBox in Kubernetes for Education

- Scalable deployment on Kuberentes-managed clusters
- Resources describe in k8s YAMLs

2018

➢ Deployed for EU-project **Up2U** at PSNC and CERN

2020

Up To University

Allow students in high-schools
to adopt tools used in science:

- **CERNBox** – Access and share content from any device
- **SWAN** – Full analysis platform in a web browser

2021

ScienceBox Timeline

2017

ScienceBox Project Epiphany

ScienceBox in Kubernetes for Education

2018

ScienceBox for HEP on Commercial Clouds

- Deployed on Helix Nebula Science Cloud
  - T-Systems: 2000+ CPUs, 10+ TB memory
- Addition of Spark cluster for massive compute

2020

2021

ScienceBox Timeline

2017

ScienceBox Project E

ScienceBox in Kuber

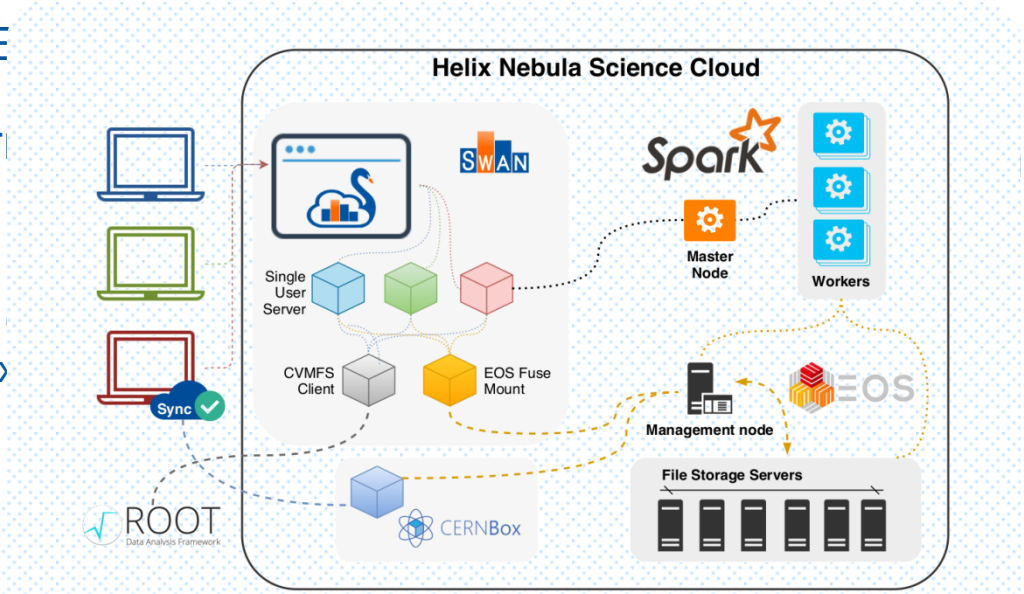
2018

ScienceBox for HEP

- Deployed on Heli
  - T-Systems:
- Addition of Spark

2020

2021



TOTEM

Big Data Tools and Cloud Services for High Energy Physics Analysis in TOTEM Experiment
https://ieeexplore.ieee.org/document/8605741

ScienceBox Timeline

2017

- ScienceBox Project Epiphany
- ScienceBox in Kubernetes for Education

2018

- ScienceBox for HEP on Commercial Clouds

2020

- ScienceBox in response to Covid19 Pandemic
  - Free-access remote-learning platforms for EU students
  - Open Up2U initiative, March 2020
  - Hosted on AWS, funded by GÉANT

2021

ScienceBox Timeline

2017

2018

2020

2021

CERN News, Apr 2020

ScienceBox in response to Covid19 Pandemic

- Free-access remote-learning platforms for EU students
- Open Up2U initiative, March 2020
- Hosted on AWS, funded by GÉANT

2021 – ScienceBox Reboot

- **Limitations of early ScienceBox**
  1. Maintainability over time
     - ✓ Chase puppet-managed production
     - ✓ Manually build container images upon new software releases
  2. Docker Compose and Kubernetes on parallel tracks
     - ✓ Changes to be implemented in both worlds
  3. Many hacks for bootstrap and configuration
     - ✓ Container's `ENTRYPOINT` is some hundred bash lines
     - ✓ (sometimes) 2+ daemons running in one container

ScienceBox 2.0

- **Goals of Reboot:** Use widely adopted CNCF technologies, improve maintainability, make use of modern containers tooling

- Major clean-up of bootstrap hacks:

```
command: ["/bin/bash", "/root/start.sh"]
```

Magic custom scripts

Plain execution of binary

```
command: ["/usr/bin/ocis", "idp"]
```

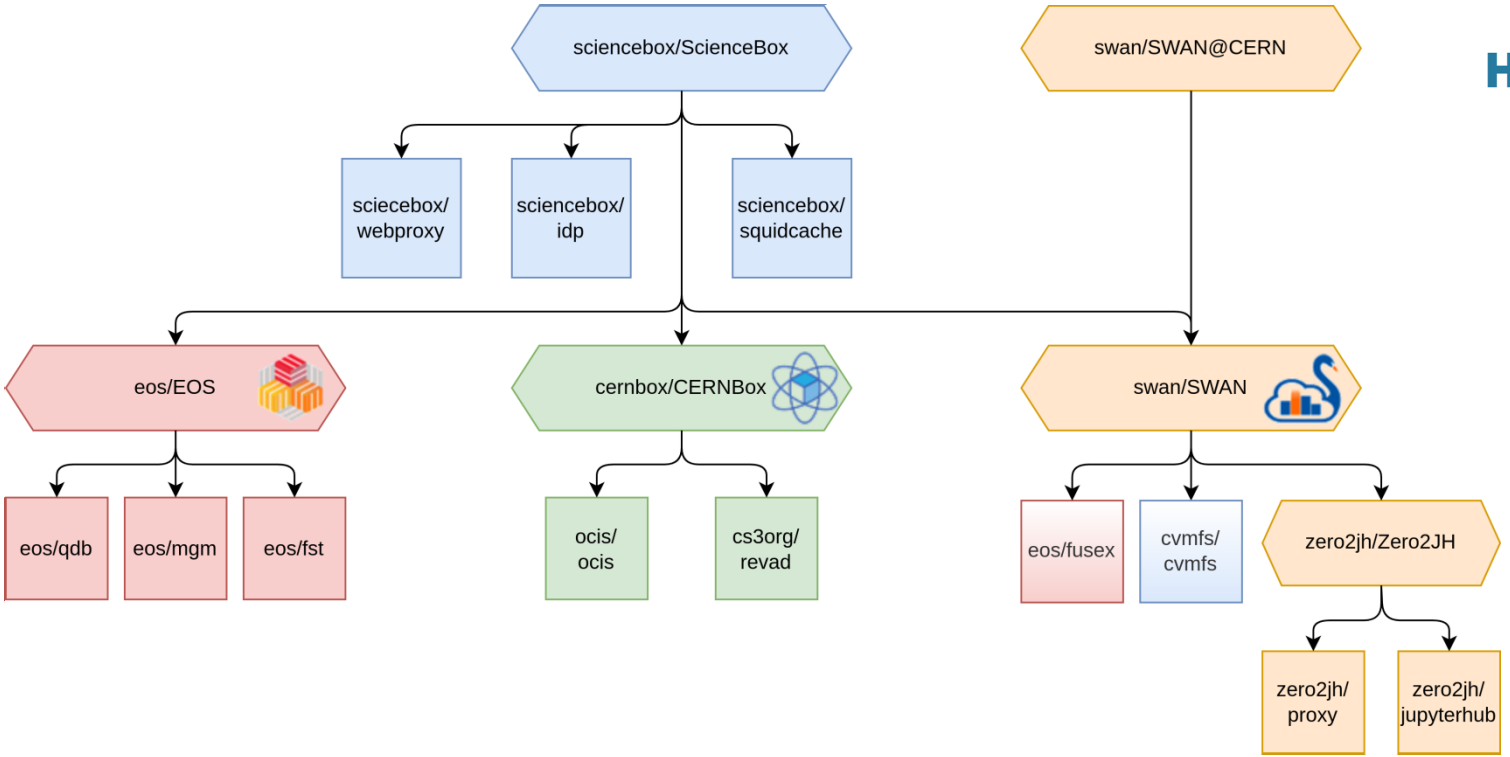
- Adopt k8s best practices (InitContainers, ConfigMaps, custom resources, …) and advanced capabilities:
  - Health-check probes, Node Selectors, Node Affinity/Anti-Affinity, Persistent Volumes Claims, Ingress and Load Balancers, etc.

ScienceBox 2.0 – Helm Charts to the Rescue

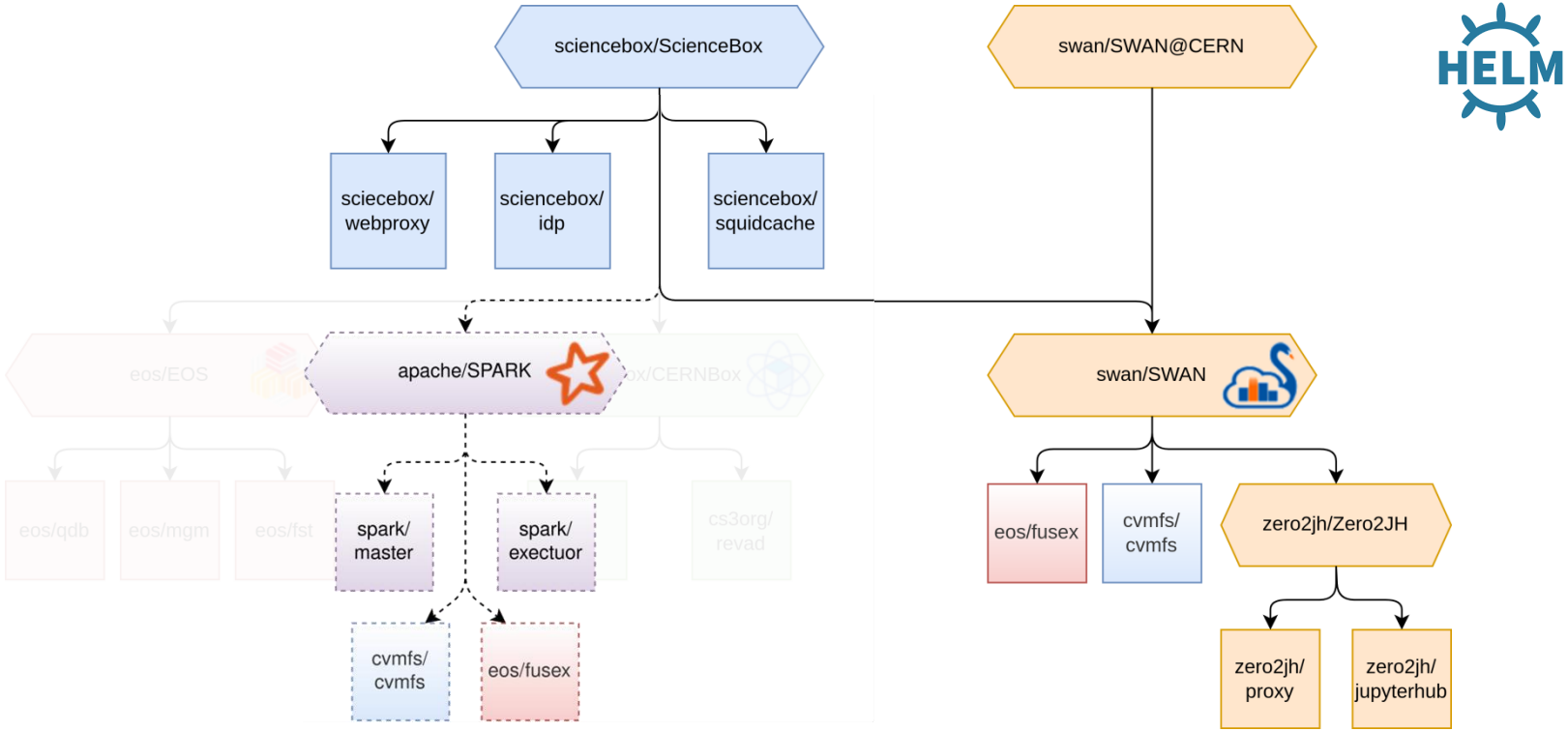
- ScienceBox is described as a collection of **Helm charts**
  - Re-use charts from main services – EOS, CERNBox, SWAN, CVMFS
  - Add the glue for stand-alone deployments
  - Allow for integrations more easily

```yaml
name: sciencebox
type: application
version: 0.0.1
description: The chart to deploy and configure ScienceBox
#
dependencies:
- name: eos
  version: 0.1.0
  repository: "https://registry.cern.ch/chartrepo/eos"
- name: swan
  version: 0.0.5
  repository: "https://registry.cern.ch/chartrepo/swan"
- name: ocis-idp
  version: 0.0.4
  repository: "https://registry.cern.ch/chartrepo/sciencebox"
```

ScienceBox 2.0 – Helm Charts to the Rescue



ScienceBox 2.0 – Modular Architecture



ScienceBox 2.0

- **Goals of Reboot:** 1. Use modern, widely-adopted container technologies, 2. Improve maintainability, 3. Ease contributions to the package

- **Modern technologies for one-click demos**
  - Get rid of Docker Compose and Kubernetes duality → Use k8s APIs everywhere

✅ Deployment on k8s-managed clusters natively via Helm

✅ Use `minikube` (or `kind`) for single-host demos and leverage on Helm again

```
1. helm repo add sciencebox https://registry.cern.ch/chartrepo/sciencebox
2. helm install sciencebox sciencebox/sciencebox
```