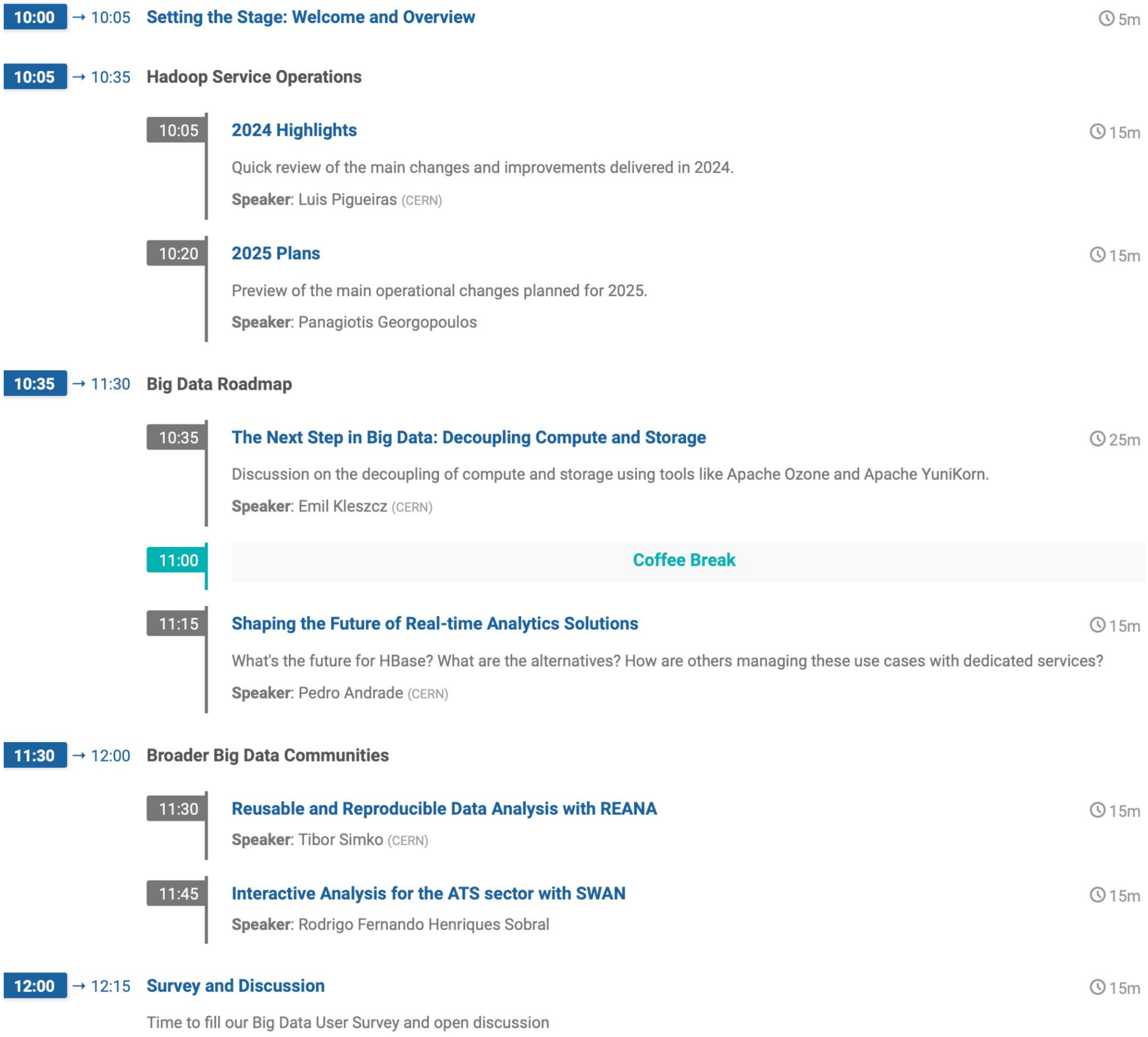
Big Data User Forum #2

The Hadoop Team
it-hadoop-support@cern.ch

Welcome

Thanks for joining :)



10:00 → 10:05 **Setting the Stage: Welcome and Overview** 5m

10:05 → 10:35 **Hadoop Service Operations**

10:05 **2024 Highlights** 15m

Quick review of the main changes and improvements delivered in 2024.

Speaker: Luis Pigueiras (CERN)

10:20 **2025 Plans** 15m

Preview of the main operational changes planned for 2025.

Speaker: Panagiotis Georgopoulos

10:35 → 11:30 **Big Data Roadmap**

10:35 **The Next Step in Big Data: Decoupling Compute and Storage** 25m

Discussion on the decoupling of compute and storage using tools like Apache Ozone and Apache YuniKorn.

Speaker: Emil Kleszcz (CERN)

11:00 **Coffee Break**

11:15 **Shaping the Future of Real-time Analytics Solutions** 15m

What's the future for HBase? What are the alternatives? How are others managing these use cases with dedicated services?

Speaker: Pedro Andrade (CERN)

11:30 → 12:00 **Broader Big Data Communities**

11:30 **Reusable and Reproducible Data Analysis with REANA** 15m

Speaker: Tibor Simko (CERN)

11:45 **Interactive Analysis for the ATS sector with SWAN** 15m

Speaker: Rodrigo Fernando Henriques Sobral

12:00 → 12:15 **Survey and Discussion** 15m

Time to fill our Big Data User Survey and open discussion

2024 Highlights

Hadoop Service

Overview

Migration to AlmaLinux 9 and upgrades

Apache Knox (SSO Integration)

BC/DR cold tests

Hardware upgrades

Alma 9 migration and upgrades

Alma 9 migration

- Reinstalled all clusters involving 143 physical machines
- Adapted our internal tools to run from Python 2 to Python 3

Software upgrades

- HDFS: from 3.2.1 to 3.3.6
  - Startup improvements + Support for Prometheus metrics
- HBase: from 2.3.4 to 2.5.10
  - More performance with new HBase metatable replication
- Phoenix: from 5.1 to 5.2
  - Security enhancements

Knox: what’s to improve?

No single gateway for Hadoop web UIs

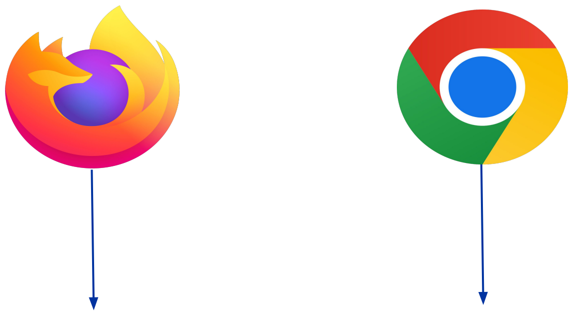
Auth requirement with Kerberos ticket/keytab

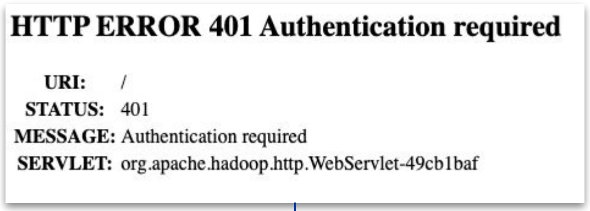
- Based on SPNEGO (GSSAPI Negotiation Mechanism)

Browser configuration adjustments

- Different for each web browser/OS

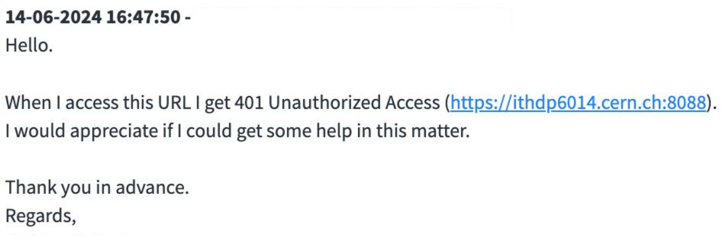
```
# Example extra settings in chrome://policy/
defaults write com.google.Chrome AuthNegotiateDelegateWhitelist "*.cern.ch"
defaults write com.google.Chrome AuthServerAllowlist "*.cern.ch"
# Restart web browser and reload policies
google-chrome --auth-server-whitelist="*cern.ch" \
--auth-negotiate-delegate-whitelist="*cern.ch"
```





HTTP ERROR 401 Authentication required

URI: /
STATUS: 401
MESSAGE: Authentication required
SERVLET: org.apache.hadoop.http.WebServlet-49cb1baf



14-06-2024 16:47:50 -
Hello.

When I access this URL I get 401 Unauthorized Access (https://ithdp6014.cern.ch:8088).
I would appreciate if I could get some help in this matter.

Thank you in advance.
Regards,

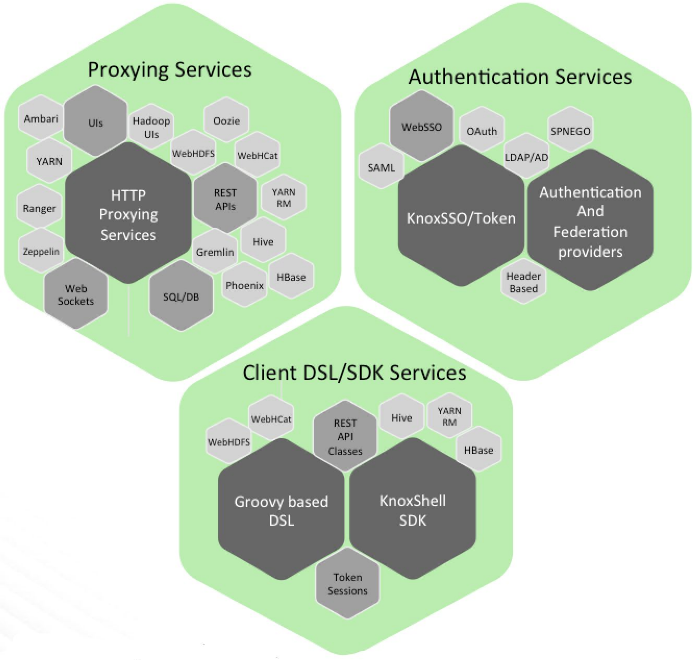
Knox: what is it?

Gateway for APIs/UIs of Apache Hadoop services

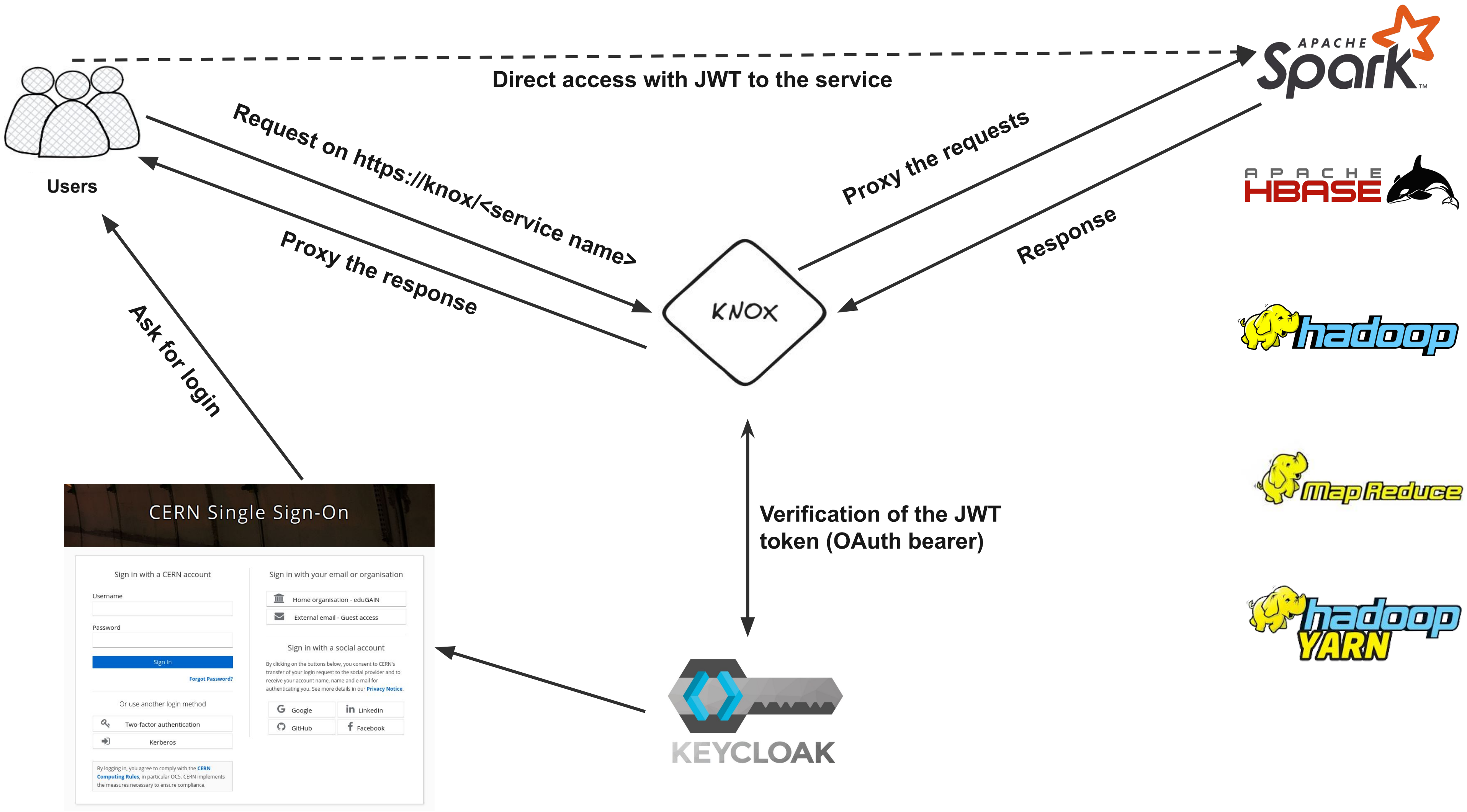
Access to Hadoop services by proxying HTTP resources

Single point of access to Hadoop clusters

Enables SSO authentication for all services



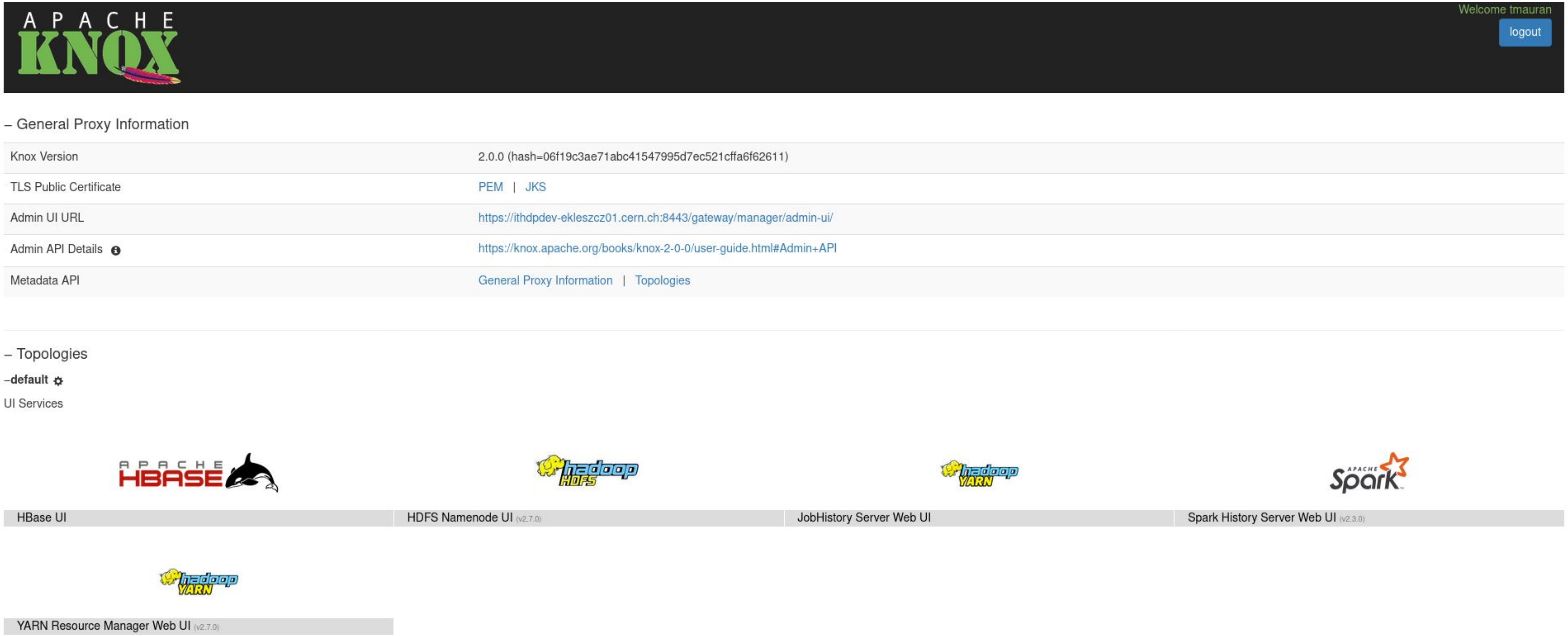
Knox: how does it work?


Direct access with JWT to the service
Users
Request on https://knox/<service name>
Proxy the response
Ask for login
KNOX
Proxy the requests
Response
Verification of the JWT token (OAuth bearer)
KEYCLOAK
APACHE Spark
APACHE HBASE
hadoop
MapReduce
hadoop YARN
CERN Single Sign-On
Sign in with a CERN account
Username
Password
Sign In
Forgot Password?
Or use another login method
Two-factor authentication
Kerberos
Sign in with your email or organisation
Home organisation - eduGAIN
External email - Guest access
Sign in with a social account
Google
LinkedIn
GitHub
Facebook

By clicking on the buttons below, you consent to CERN's transfer of your login request to the social provider and to receive your account name, name and e-mail for authenticating you. See more details in our **Privacy Notice**.

By logging in, you agree to comply with the **CERN Computing Rules**, in particular OC5. CERN implements the measures necessary to ensure compliance.

Knox: homepage



– General Proxy Information

Knox Version	2.0.0 (hash=06f19c3ae71abc41547995d7ec521cffa6f62611)
TLS Public Certificate	PEM | JKS
Admin UI URL	https://ithdpdev-ekleszcz01.cern.ch:8443/gateway/manager/admin-ui/
Admin API Details	https://knox.apache.org/books/knox-2-0-0/user-guide.html#Admin+API
Metadata API	General Proxy Information | Topologies

– Topologies

–**default**

UI Services

HBase UI

HDFS Namenode UI (v2.7.0)

JobHistory Server Web UI

Spark History Server Web UI (v2.3.0)

YARN Resource Manager Web UI (v2.7.0)

Knox: next steps

Ensure production readiness

- Conduct high availability tests
- Perform internal code refactoring and other improvements

Complete user documentation

- Update and adapt user documentation
- Provide clear instructions on accessing UIs post-deployment

Implement internal monitoring and alarms

Deployment in QA and production clusters

BC/DR cold tests

Tested different failure scenarios

- Single: recovery time ~5 min for 1 datanode | ~10 min for 1 namenode
- Partial: recovery time ~15 min for 3 datanodes
- Total: recovery time ~120 min

Tested backups recovery

- HDFS: 200 files can even take up to 5h (highly depends on the CTA queues)
- HBase: recovery time ~1 min 30 sec (for a 10GB table)
- Zookeeper: recovery time ~2 min

Hardware upgrades

Part of continuous rolling HW replacement

Analytix cluster

- Retired: 8 servers with 4.1PBs disk capacity
- Added: 8 servers with 3.5PBs disk capacity
- Delta: Same servers with -0.6PBs disk capacity

NXCALS cluster

- Retired: 18 servers with 4.7PBs disk capacity
- Added: 16 servers with 6.9PBs disk capacity
- Delta: -2 servers with +2.2PBs disk capacity

2025 Plans

Hadoop Service

Overview

Support and daily operations

Software upgrades (Spark v4, Zookeeper v3.9, Hadoop v3.4)

Hardware renovation

GitOps improvements

Retention policies for HDFS backups

Apache Ozone production deployment

NXCALS project

Support and daily operations

Documentation (link)

- Comprehensive details of the installed Hadoop Ecosystem
- Guidance on configuring and using the service

Mattermost Channel (link)

- Dedicated channel for discussions related to the Hadoop service

SNOW Ticketing Service (link)

- Addressing all questions and suggestions regarding the service

Software upgrades

Upgrade Spark to v4.0

- A major update introducing new features, performance boosts and enhanced usability for large scale data-processing
- Coordination with SWAN and other stakeholders

Upgrade Zookeeper to v3.9

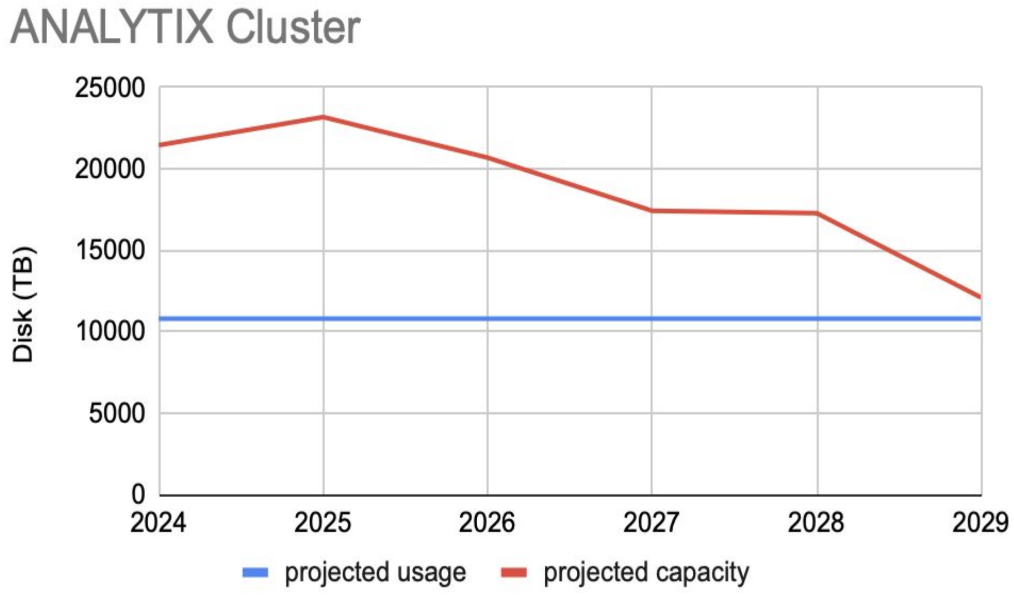
- New features like Admin server API, TLS etc

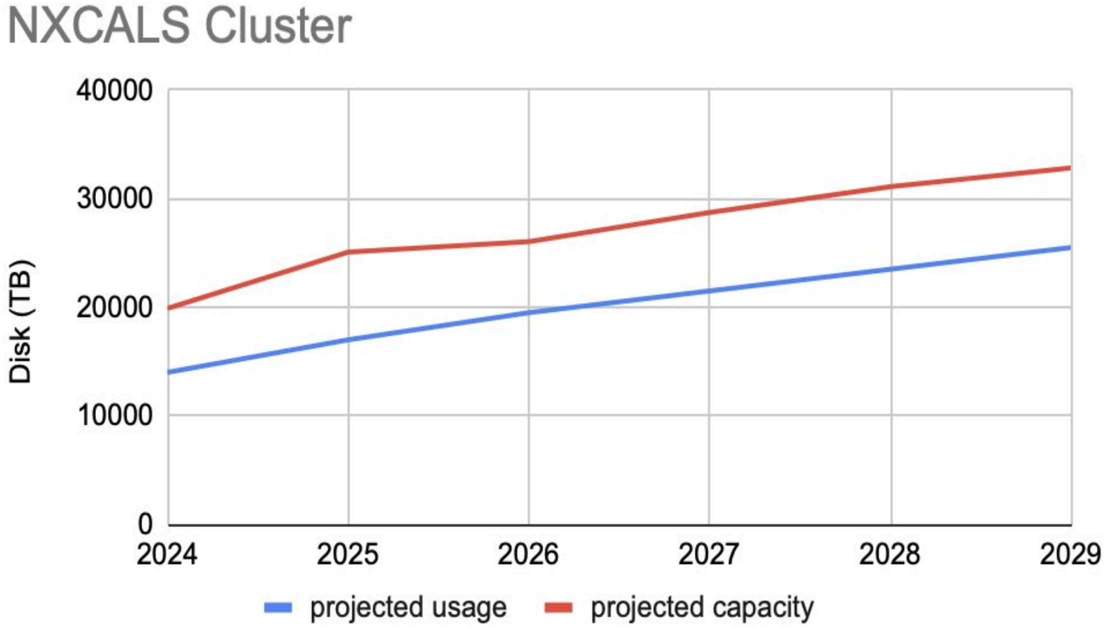
Upgrade Hadoop to v3.4

- Various improvements in HDFS/Yarn

Hardware renovation

- Install new servers and disks for **Analytix** and **NXCALS**
- 16 Servers & 16 JBODs (4 for Analytix & 12 for NXCALS)





GitOps improvements

Overview

- Adopt the latest patterns and best practises
- Enhance security, automation and consistency across workflows

Actions

- Migrate RPMs building to RPMCI
- Streamline branches, hostgroups and environments
- Add more unit tests across our repositories

Retention policies for HDFS backups

Overview

- HDFS data in all production clusters are backup up to the CTA tapes
- No current configurable retention policy for project backups
- Optimize storage usage and reduce costs

Actions

- Add new feature for data deletion in CTA
- Clean up the bulk of legacy data stored historically (PBs)

Scheduled Deletion

Apache Ozone production deployment

Overview

- Highly scalable distributed storage system optimized for Big Data
- Efficient for both object store and file system operations
- Supports HDFS and S3 compatibility

Actions

- Complete the deployment project which is already funded
- Collaborate with a technical student joining in February to specifically focus on this project

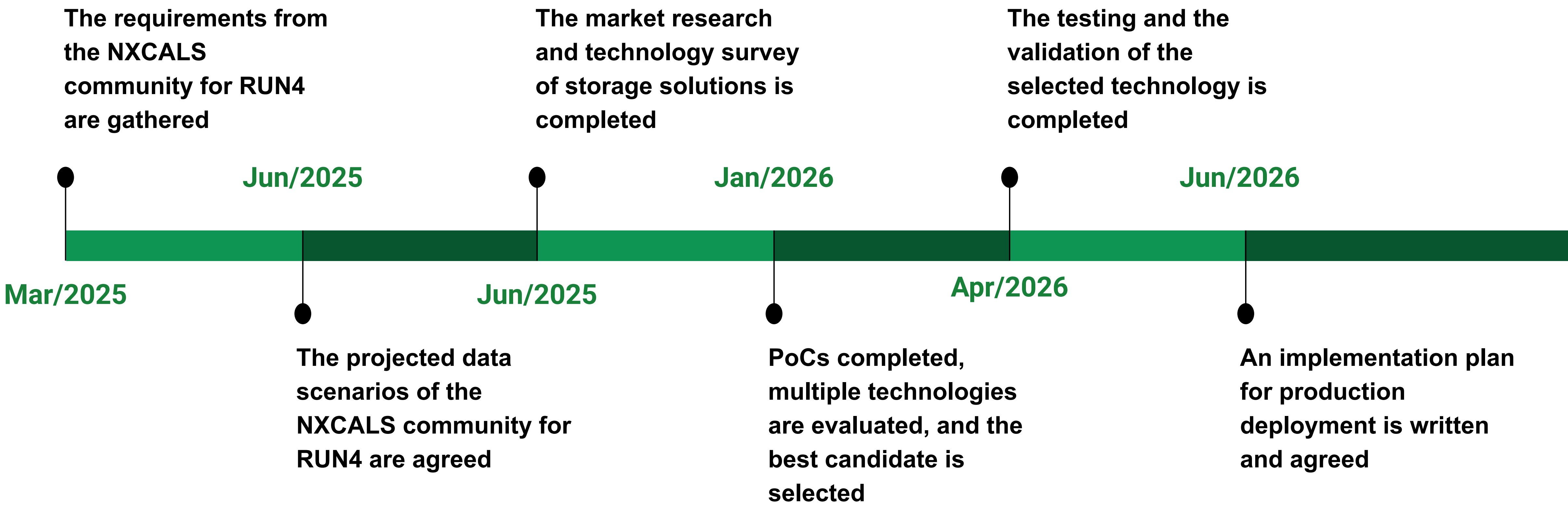
NXCALS project

Overview

Ensure that IT Services used for NXCALS can provide a smooth operation of NXCALS through RUN4 by handling all the hardware, software, and human resources requirements for that goal.

- ATS-IT engagement project with baseline effort
- Technology watch and prototyping searching for alternatives after LS3
- Requirements for RUN4 are gathered from the NXCALS community
- Explore alternatives for *HBASE* / *HDFS* / *YARN*

NXCALS project



Mar/2025

The requirements from the NXCALS community for RUN4 are gathered

Jun/2025

The projected data scenarios of the NXCALS community for RUN4 are agreed

Jun/2025

The market research and technology survey of storage solutions is completed

Jan/2026

PoCs completed, multiple technologies are evaluated, and the best candidate is selected

Apr/2026

The testing and the validation of the selected technology is completed

Jun/2026

An implementation plan for production deployment is written and agreed

The Next Step in Big Data

Decoupling Compute and Storage

Overview

Current architecture

Motivation to evolve

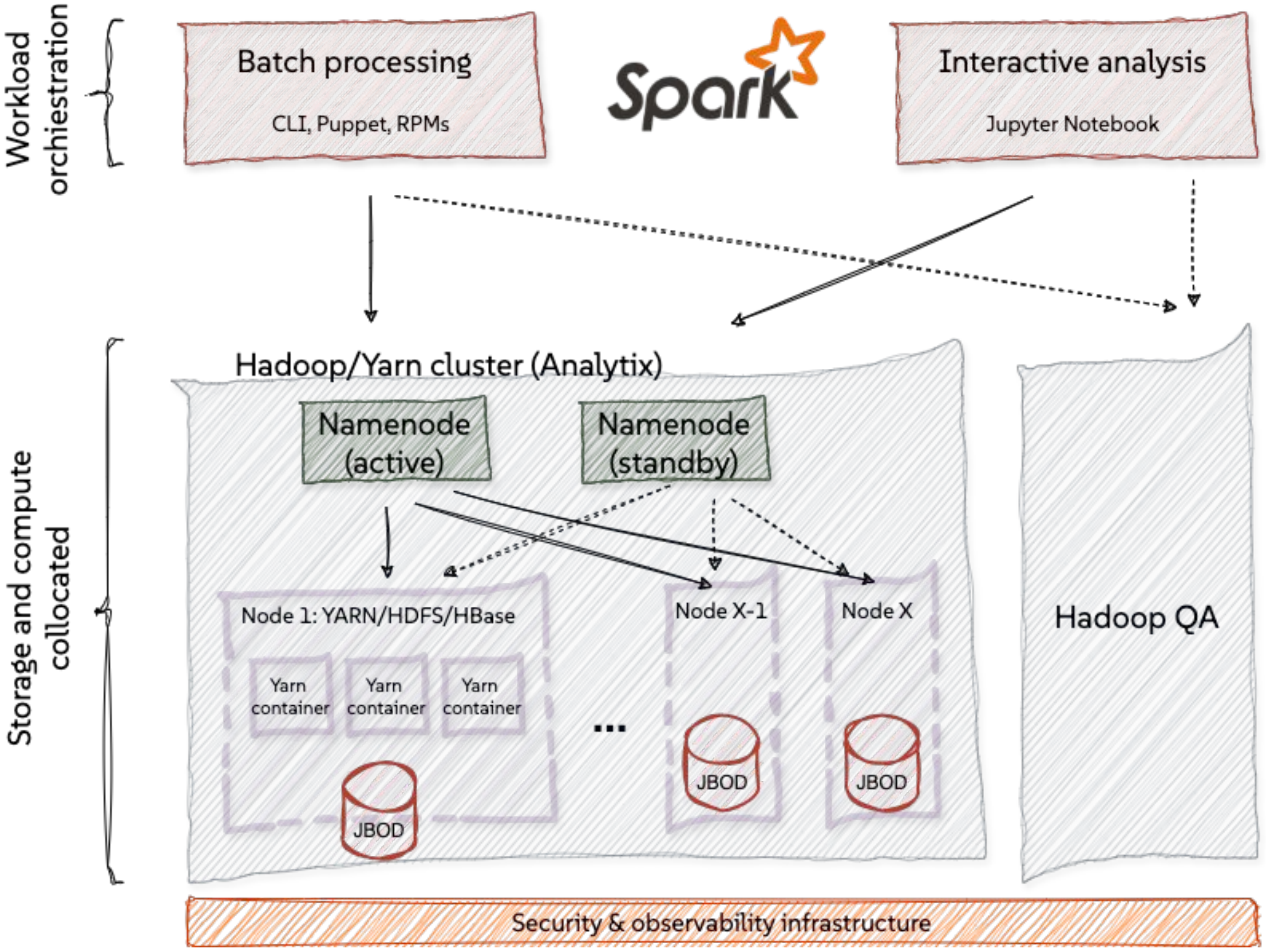
Vision for the future

Future solutions

Next steps

Current architecture

- **HDFS, YARN, HBase, Spark**
- **Puppet-managed**
- **Bare-metal machines**
- **Data locality**
- **Client access from:**
  - SWAN
  - CVMFS
  - Puppet module
  - API/CLI
  - Docker
  - RPMs
- **Monitoring of workloads:**
  - CLI, UI, API, Grafana



Motivation to evolve

Scalability Needs:

- HDFS struggles with billions of small files

Infrastructure Limitations:

- Puppet-managed bare-metal nodes are rigid
- K8s setups offer flexible, containerized envs.

Modern Storage Requirements:

- Block storage might be inefficient for massive datasets (fixed block size)
- Object storage is cost-effective, scalable solution

Keeping Pace with Industry:

- Transition aligns with modern Big Data/AI trends adopted by leading organizations

Cost efficiency:

- Bare-metal infrastructure costly to maintain
- K8s supports dynamic resource allocation and better cost management

Enhanced User Experience:

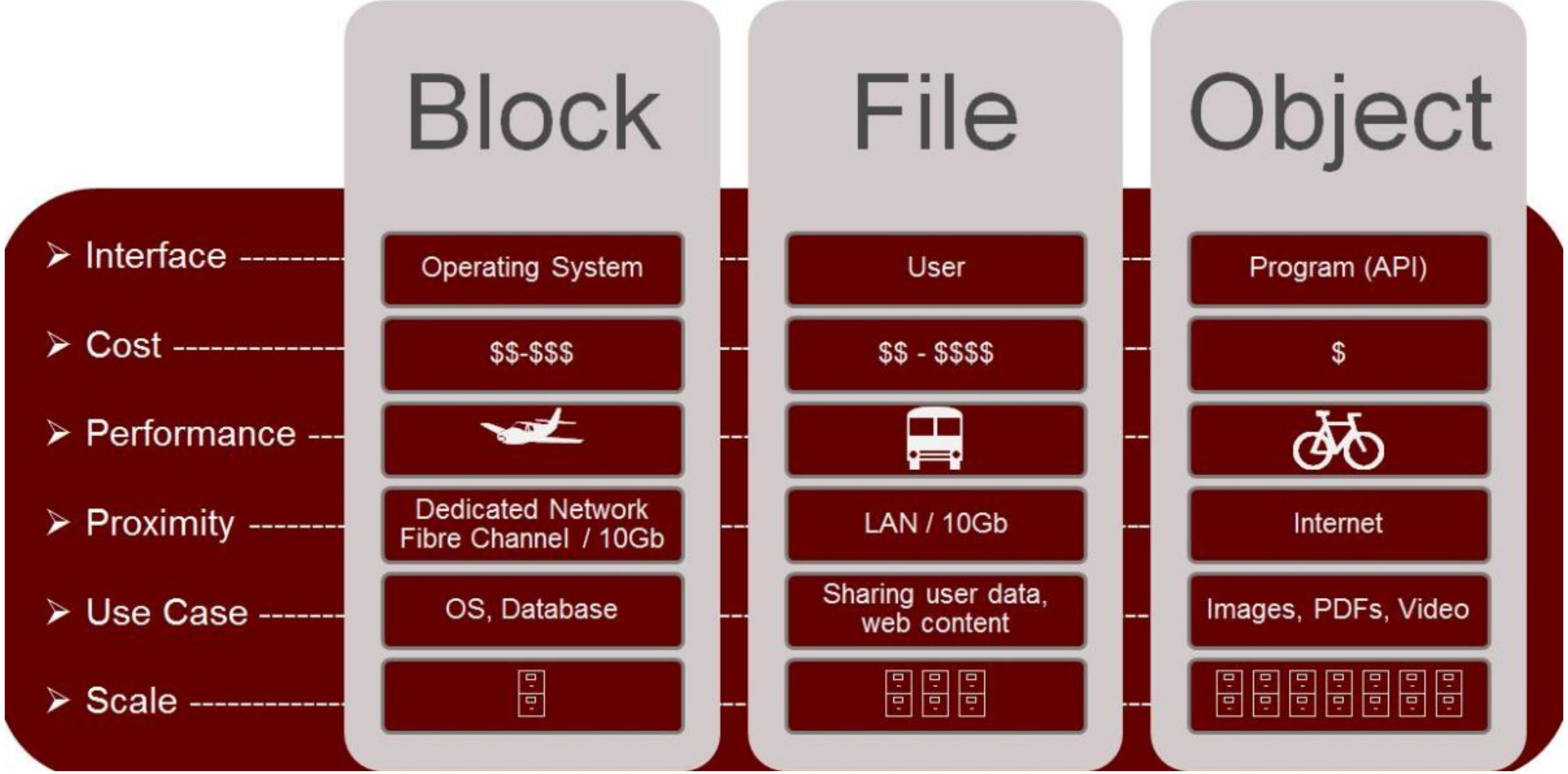
- Modern interfaces and workflows improve usability and productivity

Motivation to evolve: storage

Trade-off: cost vs performance vs scalability

HDFS: File Storage with a block-based storage mechanism

Ozone: Hybrid - supports both Object & Block Storage use cases



	Block	File	Object
➢ Interface	Operating System	User	Program (API)
➢ Cost	$$-$$$	$$ - $$$$	$
➢ Performance	✈	🚌	🚲
➢ Proximity	Dedicated Network Fibre Channel / 10Gb	LAN / 10Gb	Internet
➢ Use Case	OS, Database	Sharing user data, web content	Images, PDFs, Video
➢ Scale	🗄	🗄🗄🗄	🗄🗄🗄🗄🗄🗄🗄

Ref. https://forum.huawei.com/enterprise/en/characteristics-of-computer-storage-devices/thread/694722873471680512-667213859733254144

Motivation to evolve: compute

Cost optimisation

- Dynamic scaling
- Resource efficiency
- On-prem/multi-cloud

Why K8s?

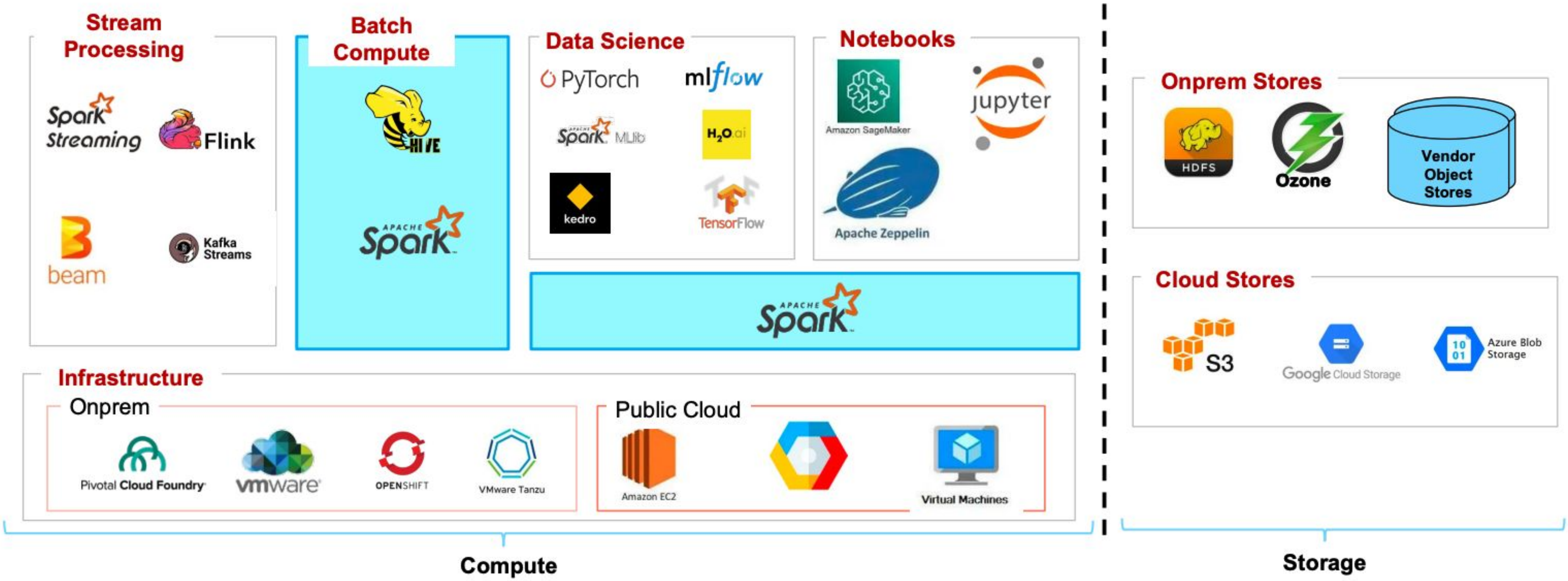
Support

- Strong community
- Vast ecosystem of tools
- Observability

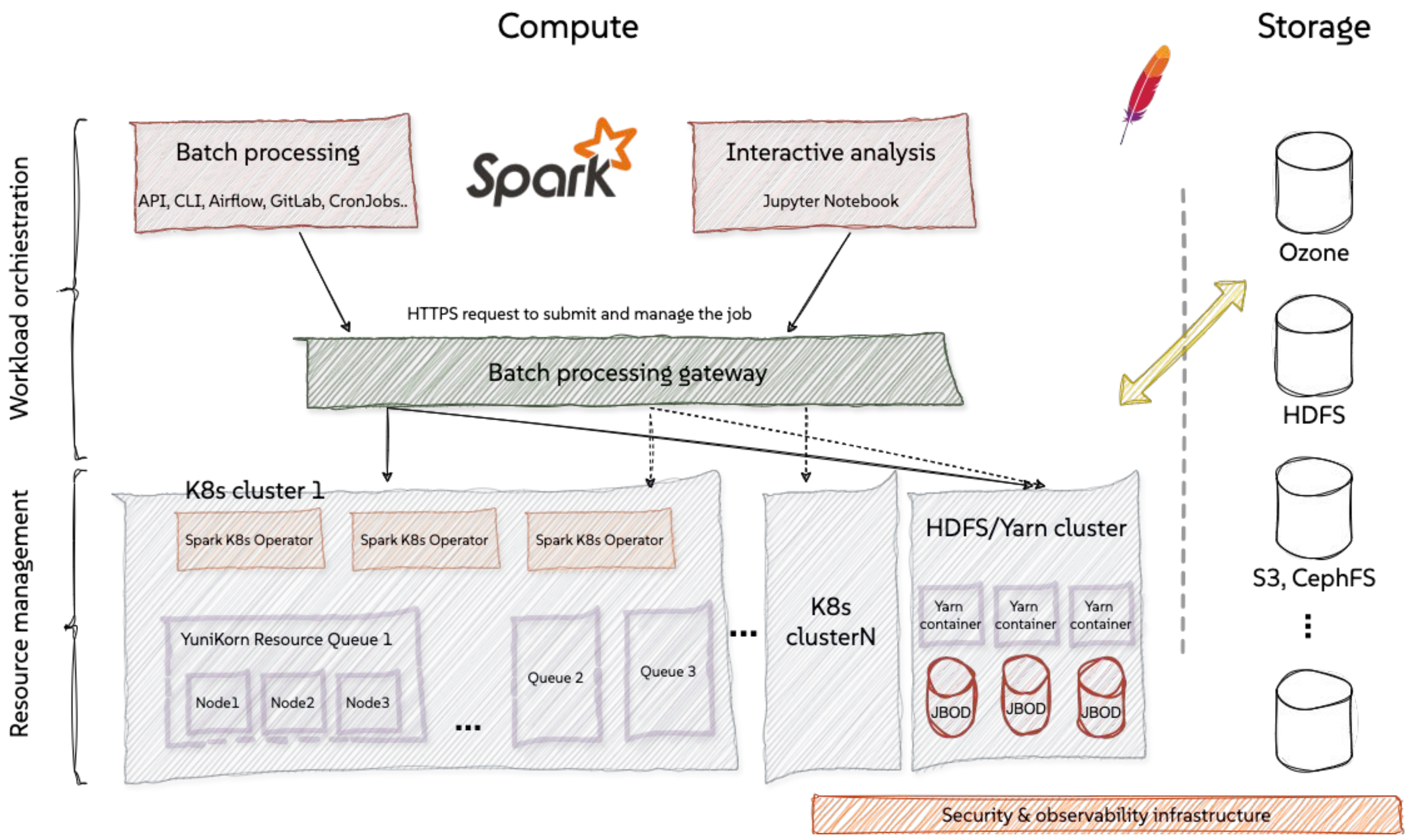
Containerisation

- Portability of the apps
- Better resource isolation
- Dependency management

What industry (Enterprise) does?



The Big Data future is bright



Future solutions: Apache Ozone

- Highly scalable distributed FS
- Scales to Exabyte
- HDFS & S3 compatible APIs

- Billions of objects
- Namespace with volumes
- Keys (objects) stored under buckets

Write a file example

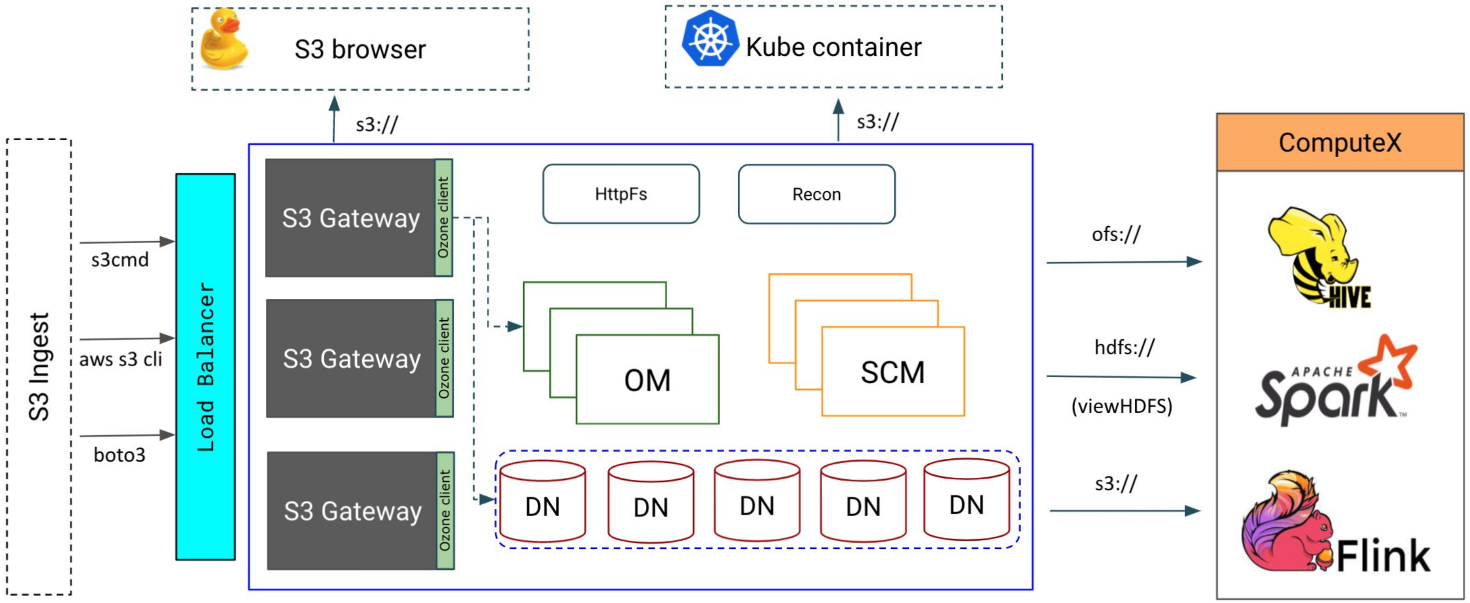
```
# Create volume and bucket
$ ozone sh volume create /vol1
$ ozone sh bucket create /vol1/buck1

# Write a file
$ ozone fs -mkdir -p /vol1/buck1/dir1
$ ozone fs -touch /vol1/buck1/dir1/key1

# Cannot create file under root or volume
$ ozone fs -touch /vol1/key1

# Migrate data
hadoop distcp
hdfs://namenode:8020/source-path
ofs://ozone1/destination-path
```

Ozone - Multiple Protocol Support



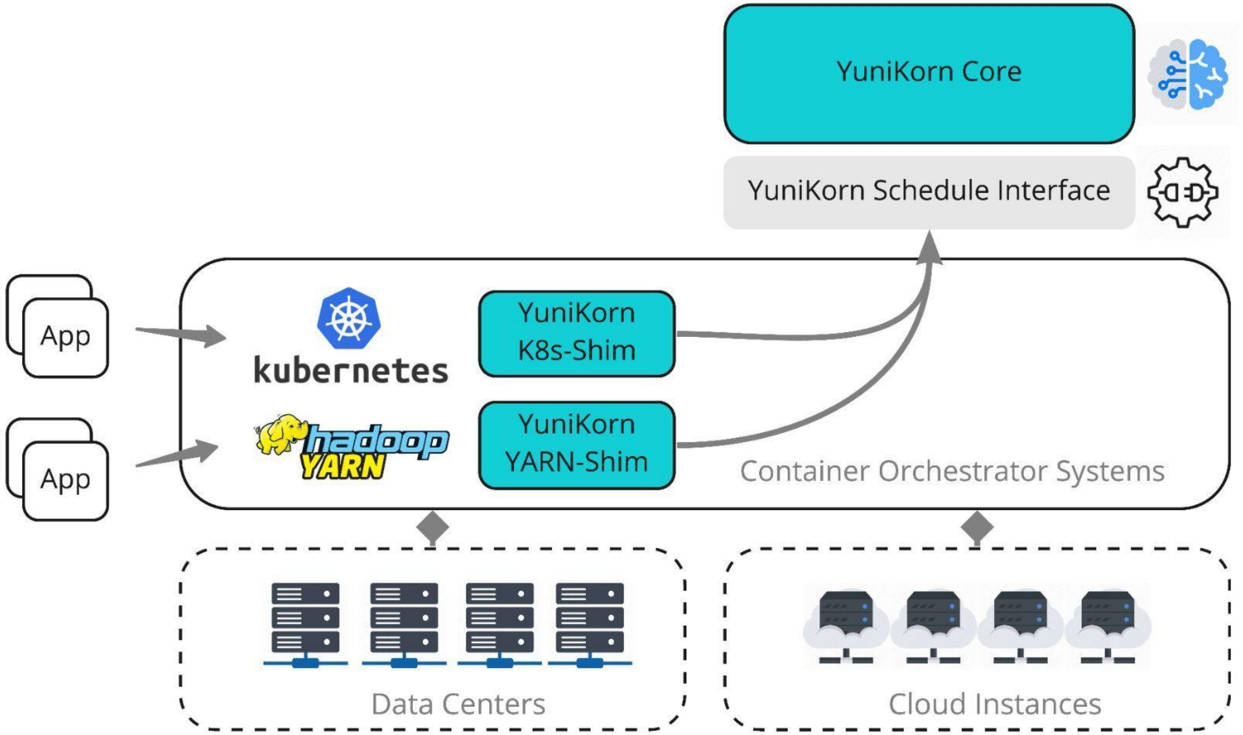
Future solutions: resource schedulers for K8s

Future solutions: Apache YuniKorn

Main characteristics

- Light-weight resource scheduler
    - for container orchestrator systems
- Suitable for batch workloads
- Introduced in 2020
- YuniKorn-web
- Active prod-ready project

Major Adopters



Future solutions: Apache YuniKorn

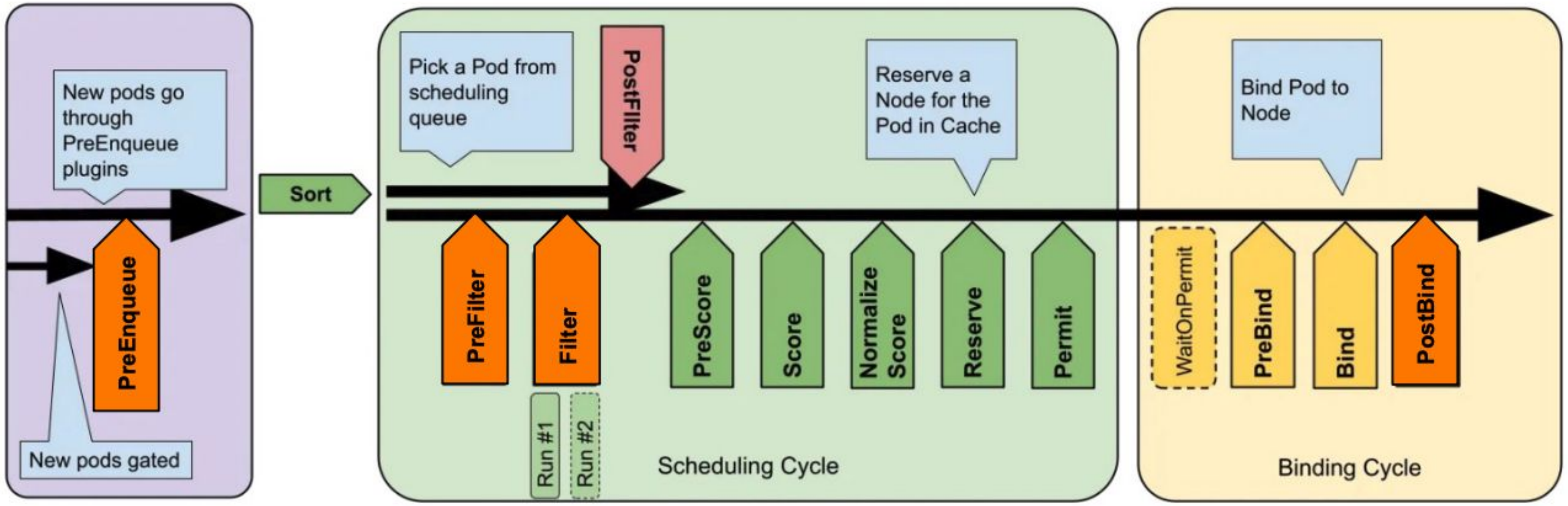
YUNIKORN

Pod scheduler

- YuniKorn extends the K8s native scheduler
- To improve resource allocation and scheduling
- Especially in multi-tenant environments
- For resource-heavy workloads

Key phases

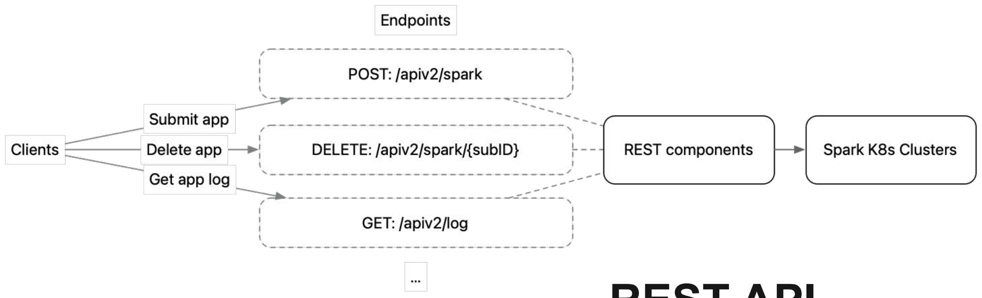
- ***PreEnqueue***: Initial checks before pod enters queue
- ***Scheduling Cycle***:
  - Resource checks, placement constraints,...
- ***Binding Cycle***: Assigns pod to a node



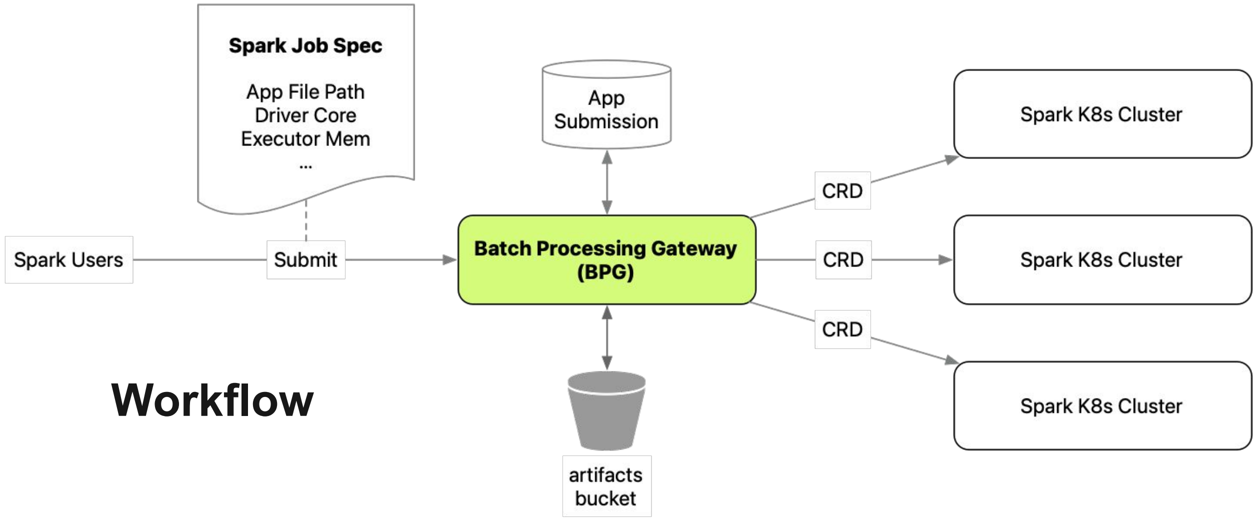
Future solutions: Batch Processing Gateway

Submission workflow

- **Publish app artefacts:** .jar, .py, .zip files to S3 bucket.
- **Compose job spec:** job path, driver core, executor memory, etc.
- **Submit job spec to REST endpoint**
- **BPG parses request:** translates to CRD format.
- **Cluster selection:** BPG chooses cluster using queue/weight conf.
- **CRD is processed and app is submitted**

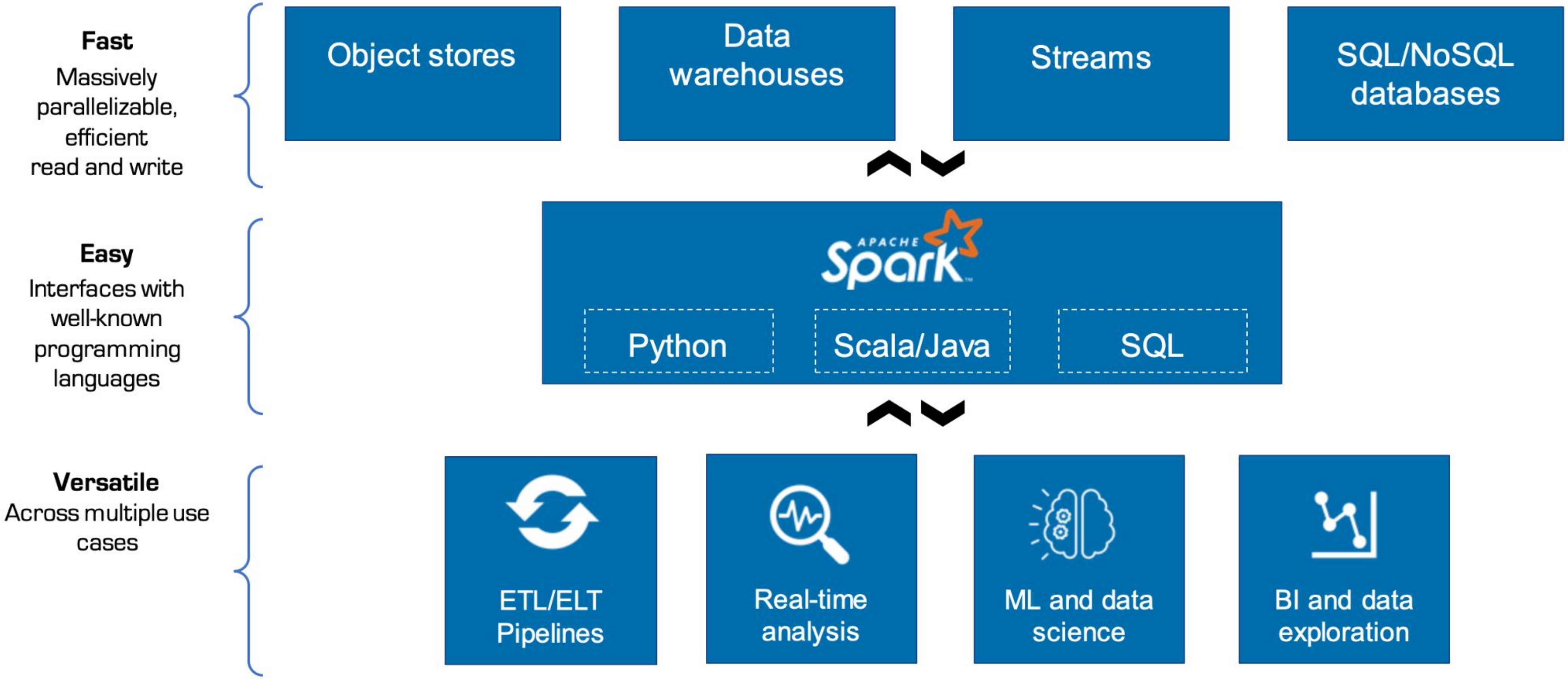


REST API



Workflow

Future solutions: Spark computing engine



Fast

Massively parallelizable, efficient read and write

Object stores

Data warehouses

Streams

SQL/NoSQL databases

Easy

Interfaces with well-known programming languages

Apache Spark

Python

Scala/Java

SQL

Versatile

Across multiple use cases

ETL/ELT Pipelines

Real-time analysis

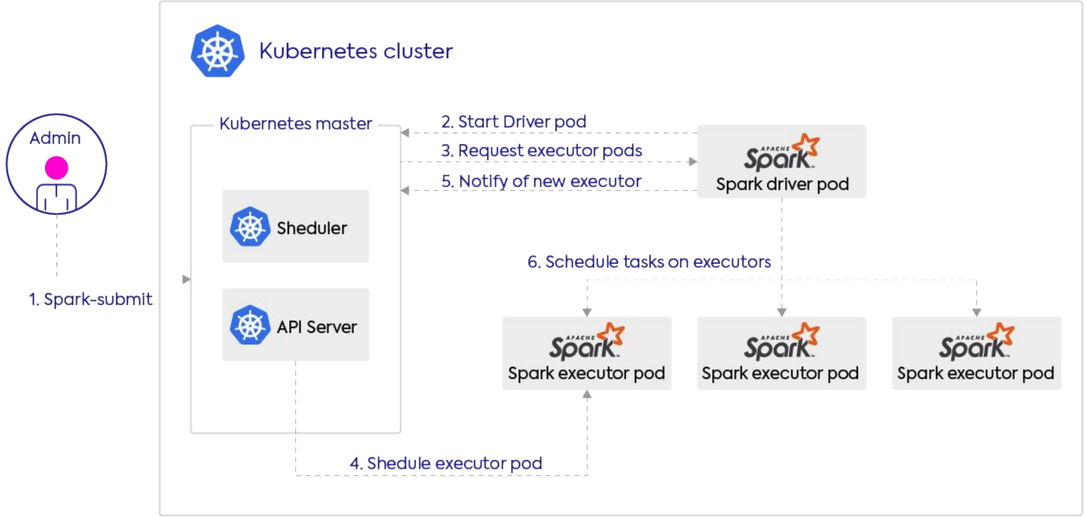
ML and data science

BI and data exploration

Future solutions: Spark on K8s

Benefits

- **Scalability**: Handle peaks with elastic provisioning
- **Orchestration**: Simplify workload management
- **Flexibility**: Enable hybrid and multi-cloud setups
- **Efficiency**: Nodes can be adjusted to compute needs
- **Performance**: Similar to YARN



Kubernetes cluster

Admin

1. Spark-submit

Kubernetes master

Sheduler

API Server

2. Start Driver pod

3. Request executor pods

5. Notify of new executor

Spark driver pod

6. Schedule tasks on executors

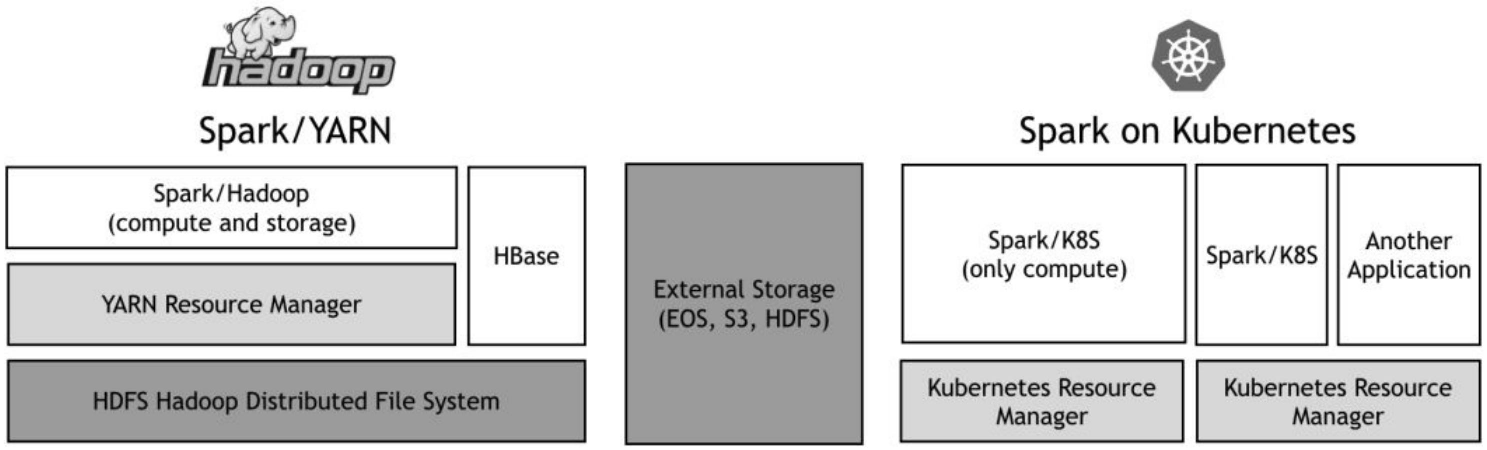
Spark executor pod

Spark executor pod

Spark executor pod

4. Shedule executor pod

https://spot.io/blog/setting-up-managing-monitoring-spark-on-kubernetes/



Spark/YARN

Spark/Hadoop (compute and storage)

HBase

YARN Resource Manager

HDFS Hadoop Distributed File System

External Storage (EOS, S3, HDFS)

Spark on Kubernetes

Spark/K8S (only compute)

Spark/K8S

Another Application

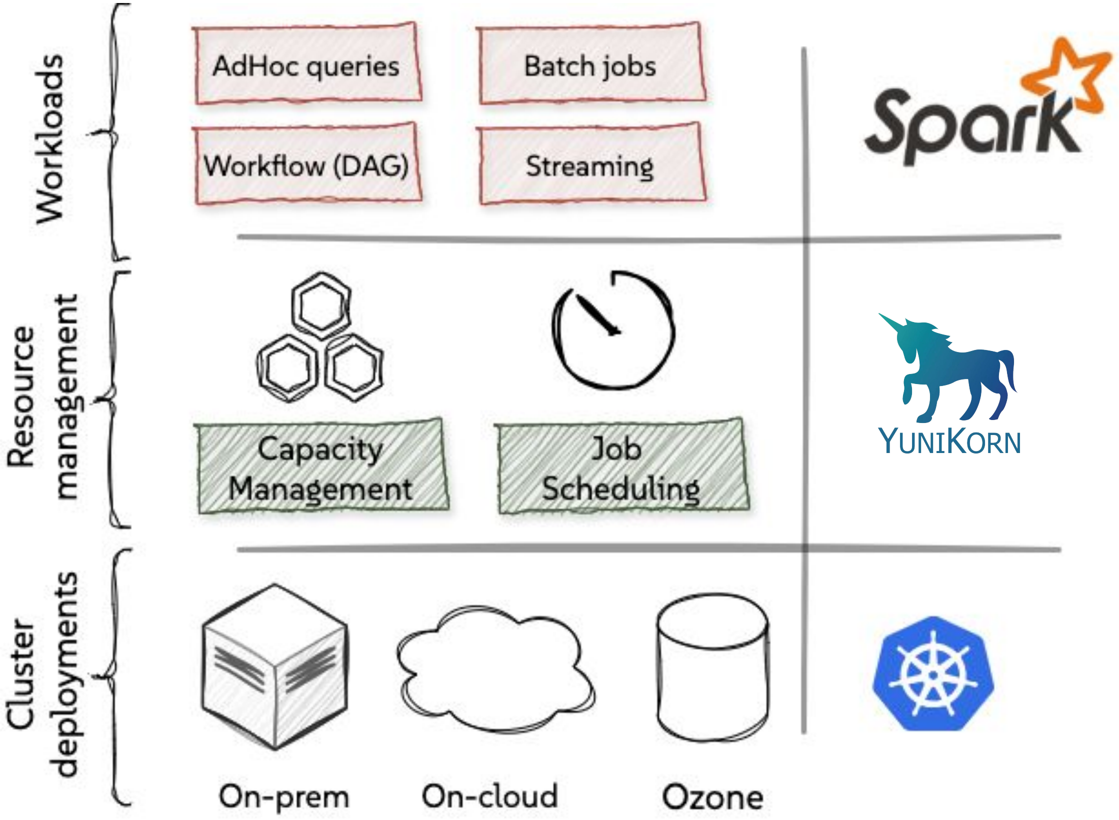
Kubernetes Resource Manager

Kubernetes Resource Manager

Future solutions: Transition

Transition goals

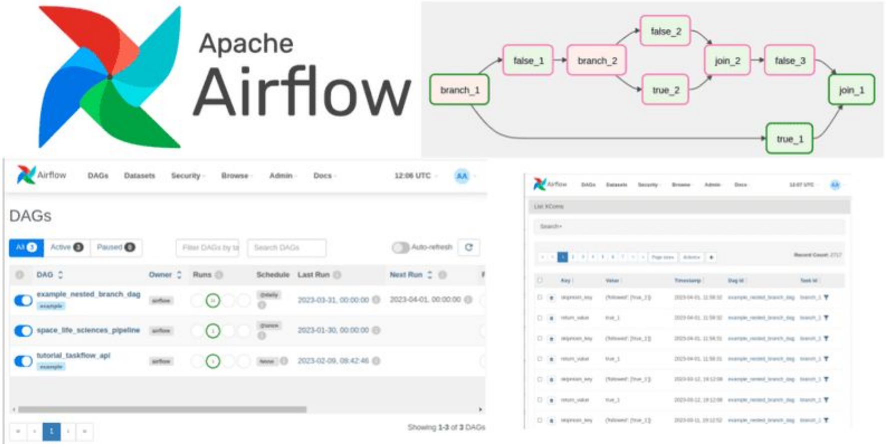
- Provide easy access to the clusters
- Integrate YuniKorn
- Move Puppet-managed nodes to K8s
- Replace HDFS /w scalable object storage
- Storage decoupled



Future solutions: data tools

Apache Airflow

- Orchestrates and automates complex data pipelines
- **Key Features**: Python-based workflows, scheduling, monitoring
- **Use Case**: Efficiently managing data workflows at scale for big data analytics and ML pipelines

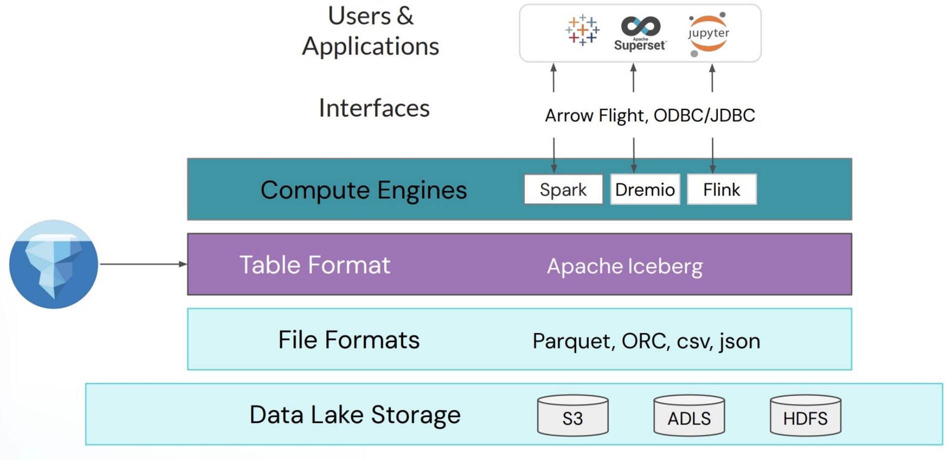


DAGs

DAG	Owner	Runs	Schedule	Last Run	Next Run
example_nested_branch_dag example	airflow	14	@daily	2023-03-31, 00:00:00	2023-04-01, 00:00:00
space_life_sciences_pipeline	airflow	1	@once	2023-01-30, 00:00:00	
tutorial_taskflow_api example	airflow	1	None	2023-02-09, 09:42:46	

Iceberg

- High-performance table format with ACID transactions
- **Key Features**: Schema evolution, distributed analytics
- **Use Case**: Efficient data handling at PB with Spark



Future solutions: data tools

Trino

- Distributed SQL query engine for low-latency analytics
- **Key Features:** Queries across multiple data sources, real-time results
- **Use Case:** Fast analytics over large datasets without data movement

SWAN @ CERN

- **Key Features:** Data visualization and high-performance analysis
- **Use Case:** Real-time insights for scientific research at CERN

Next steps for Big Data evolution

Smooth transition strategy

- Ensure seamless integration with minimal disruption **to the user community**
- Move to early adopters and gradual migration if PoCs are successful

Explore new technologies and architectures

- YuniKorn and Ozone for resource management and scalable storage
- If time allows: Kubernetes testing, ML/AI with GPUs

Collaborate with others

- Leverage synergies with SWAN, CephFS/S3, SSO, OpenShift, NXCALS, Experiments
- Starting already today with your questions/feedback and filling the survey

Real-time Analytics Solutions

What’s next?

Overview

Motivation and use cases

HBase history and status

HBase limitations and risks

Future solutions

Next steps

Motivation and use cases

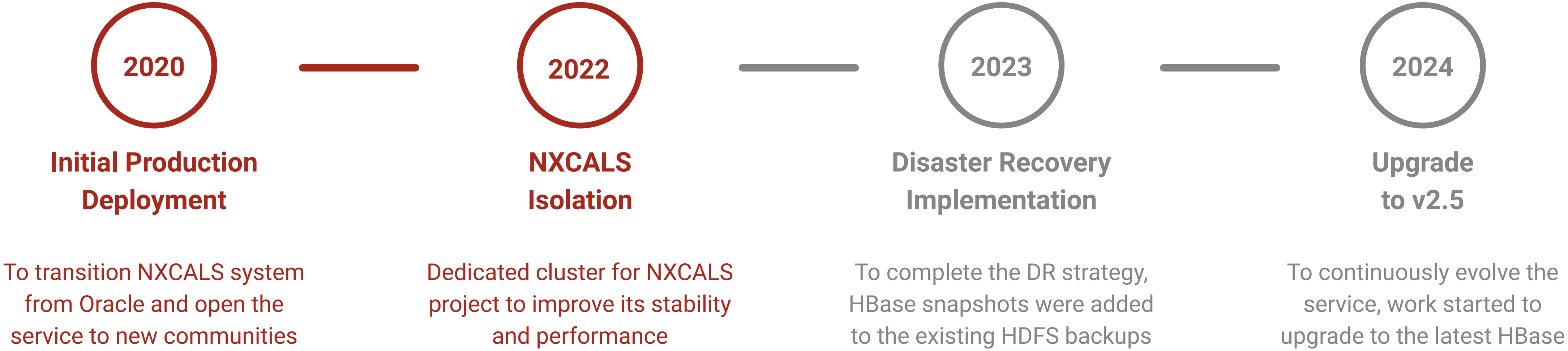
Why we started HBase

- Real-time access to data, optimized for low-latency reads and writes
- Scalable Big Data storage handling massive structured data across distributed clusters
- Hadoop integration with seamless integration with HDFS and MapReduce

What HBase enables

- *CERN NXCALS:* A scalable data archiving system that supports efficient storage and analysis of control system data for CERN's accelerators
- *ATLAS Event Index:* A metadata catalogue that indexes events from the ATLAS experiment, enabling fast searches and access to event-level data for physics analysis

HBase history and status



2020

Initial Production Deployment

To transition NXCALS system from Oracle and open the service to new communities

2022

NXCALS Isolation

Dedicated cluster for NXCALS project to improve its stability and performance

2023

Disaster Recovery Implementation

To complete the DR strategy, HBase snapshots were added to the existing HDFS backups

2024

Upgrade to v2.5

To continuously evolve the service, work started to upgrade to the latest HBase

HBase history and status

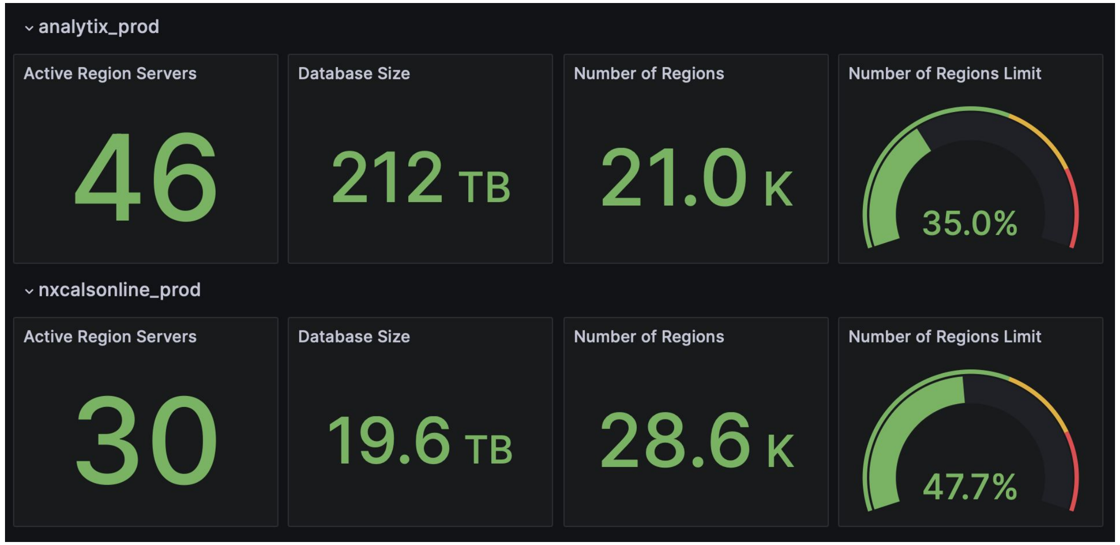
Several internal DEV clusters

3 QA/TEST clusters

- Hadoop QA *(with HDFS/YARN)*
- NXCALS Dev Online *(dedicated)*
- NXCALS PerfTest Online *(dedicated)*

2 PROD clusters

- Analytix *(with HDFS/YARN)*
- NXCALS Online *(dedicated)*



analytix_prod

Active Region Servers 46

Database Size 212 TB

Number of Regions 21.0 K

Number of Regions Limit 35.0%

nxcalsonline_prod

Active Region Servers 30

Database Size 19.6 TB

Number of Regions 28.6 K

Number of Regions Limit 47.7%

HBase limitations and risks

Technology-specific challenges

- No complaints… works very well for all our use cases!
- Community reports performance issues with compactions and flushing to HDFS

Support and maintenance

- Operational complexity: requires deep expertise for setup, tuning, and troubleshooting
- Sparse documentation: advanced features lack sufficient guidance

HBase limitations and risks

Future perspectives

- Community activity: slower development compared to other alternatives
- Decreasing usage: reduced adoption for newer big data ecosystems
- Evolving use cases: struggles to compete in cloud-native or hybrid environments
- Integration needs: dependent on other tools (e.g. Apache Phoenix for SQL support)

Future solutions: requirements

What we are looking for

- Real-time analytics allowing storing time-series data
- High throughput for real-time like data (with caching capabilities)
- Traction in the market and mature/stable project
- Compatibility with other storage backends (not only HDFS)
- Preferably providing wide-columnar store capabilities
- Scalable with partitioning of the data
- High-availability capabilities
- Compression provided with optimized file formats

Open Source

Future solutions: panorama

cassandra
SCYLLA.
pinot
databricks
ClickHouse
QuestDB
druid
snowflake
KairosDB

Future solutions: Cassandra

Apache Software Foundation (ASF)

~15 years, released in July 2008

Contributors:

- Apache Software Foundation community
- Major corporate contributors: DataStax, Netflix, Apple, Amazon

Top Users:

- Netflix: streaming data and recommendations
- Instagram: scalable social media backend
- Spotify: user activity tracking
- eBay: real-time product search and analytics
- Uber: geo-location and ride analytics

Future solutions: Cassandra

cassandra

Some initial positives

- Wide-column store:
  - Excellent for time-series data with tunable consistency
- High throughput:
  - Designed for high-speed writes tuning
- Kubernetes compatibility:
  - Operators like Cass-Operator streamline deployment on K8s
- Market traction:
  - Widely adopted and mature

Future solutions: ClickHouse

ClickHouse Inc.

~7 years, released in June 2016

Contributors:

- ClickHouse Inc.
- Yandex (initial creators)
- Open-source contributors from analytics and cloud community

Top Users:

- Cloudflare: real-time network analytics
- Uber: business metrics and monitoring
- Yandex: web analytics and search engine metrics
- Alibaba: e-commerce analytics

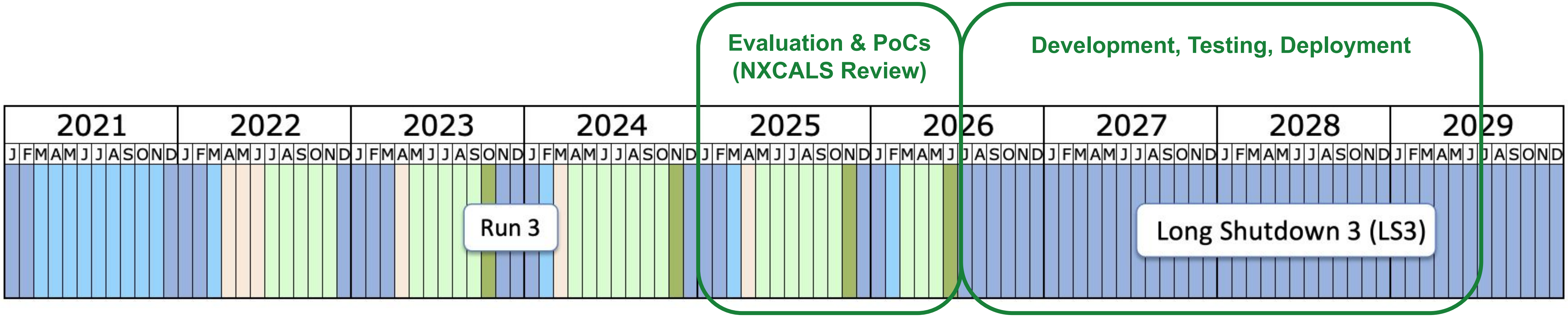
Future solutions: ClickHouse

ClickHouse

Some initial positives

- Columnar database:
  - Optimized for analytical queries on time-series and structured data
- High throughput:
  - Excellent for OLAP workloads
- Kubernetes compatibility:
  - Operators available for managing deployments
- Market traction:
  - Growing adoption due to performance and simplicity for analytics

Next steps



Investigate and select few most-promising solutions

Explore, test, and deploy proof of concepts

Get early feedback from the community and interested teams

- Starting already today with your questions/feedback and filling the survey

User Survey

https://indico.cern.ch/event/1468866/surveys/5910

Thank you! Questions?