Data Processing and Preservation System (DPPS) with focus on Bulk Data Management System (BDMS) for the Cherenkov Telescope Array (CTA) Observatory

Swiss CTA Day 2023: 14th December, 2023
International Space Science Institute (ISSI), University of Bern

Syed Hasan*, Adrian Biland (ETH Zurich), Etienne Lyard, Hancheng Li, Roland Walter (University of Geneva, Department of Astronomy)

Agenda

- DPPS overview, BDMS Bulk Archive, and Data centres

- Open Archival Information system (OAIS) ISO Standard - LOGICAL FLOW

- BDMS Architecture (Functional decomposition)

- Swiss Contribution to BDMS

- BDMS Prototyping efforts at DESY Kubernetes cluster for Release 0 - Status

- Work Plan for the next six months

Data Processing and Preservation System (DPPS)
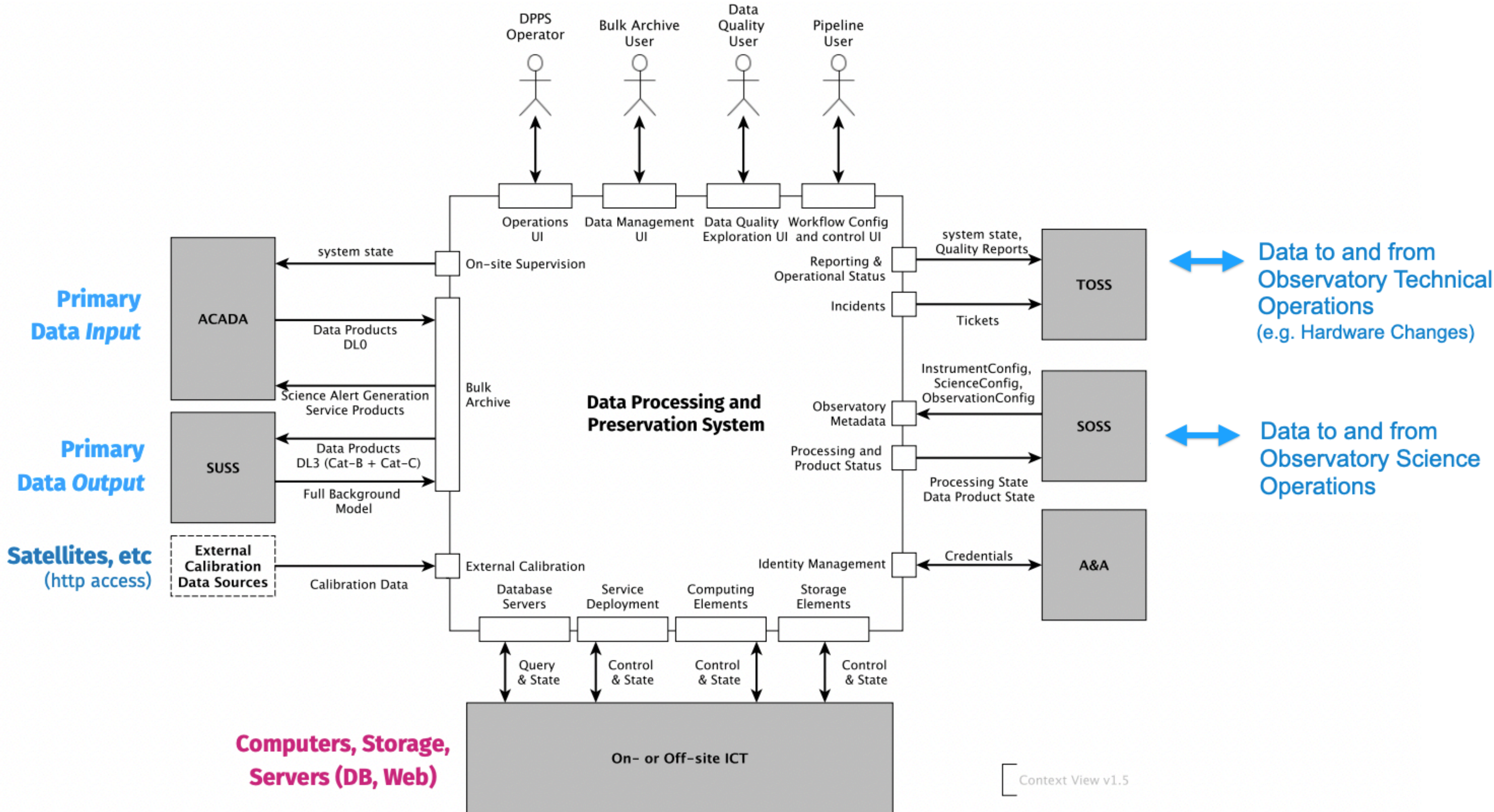
1

Bulk Archive managed by Bulk Data Management System (BDMS) and its relationship with other CTAO systems

ACADA: Array Control and Data Acquisition System

SUSS: Science User Support System

SOSS: Science Operations Support System

DPPS: BDMS

List of Contributors in Switzerland

- Syed Hasan, Adrian Biland (ETH Zürich)

- Etienne Lyard, Hancheng Li, Roland Walter (University of Geneva, Department of Astronomy)

- Close collaboration with EPFL and CSCS (Off-site ICT Work Package (WP): Volodymyr Savchenko, Pablo Fernandez, Victor Holanda Rusu, and Andrii Neronov)
  - CSCS is one of the teams participating in this WP

List of Contributors in Italy

- Stefano Gallozzi (lead), Fabrizio Lucarelli, Georgios Zacharis (INAF OAR, Roma)

Data centres for DPPS, SUSS, and SOSS

- Off-site data centres: PIC (Spain), Frascati (Italy), CSCS (Switzerland), and DESY (Germany)
- Part of BDMS runs also on on-site ICT

Open Archival Information system (OAIS) ISO Standard - LOGICAL FLOW

- **OAIS** standards design from high energy astronomy archive experience
- **INTEGRAL** archive to be the first one to be implemented in the framework of OAIS
- Similar concept could be used for missions such as **SKA**, OAIS archive can be used
- **Rucio** is a software framework (open-source, developed at CERN) that provides functionality to organize, manage, and access large volumes of scientific data. For CTA, Rucio provides a declarative engine for bulk archive with data being stored at multiple off-site datacenter
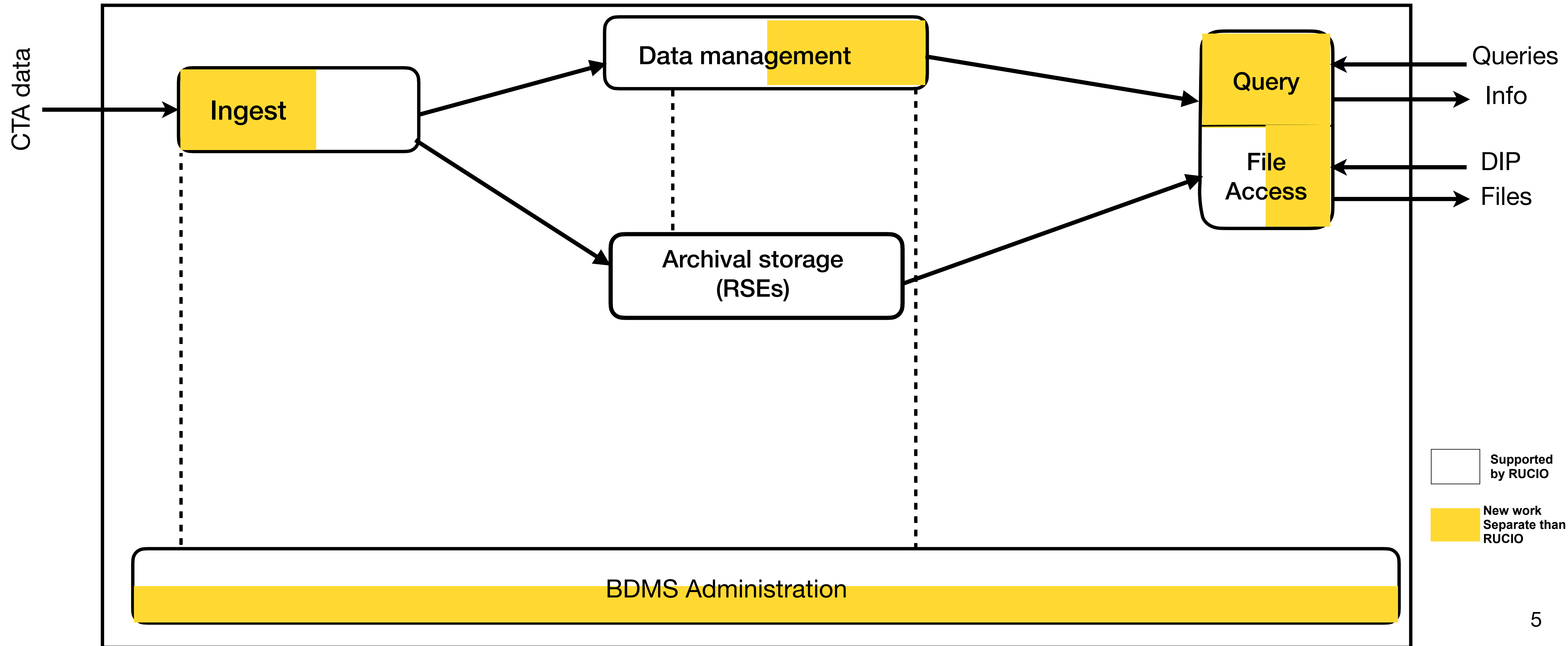
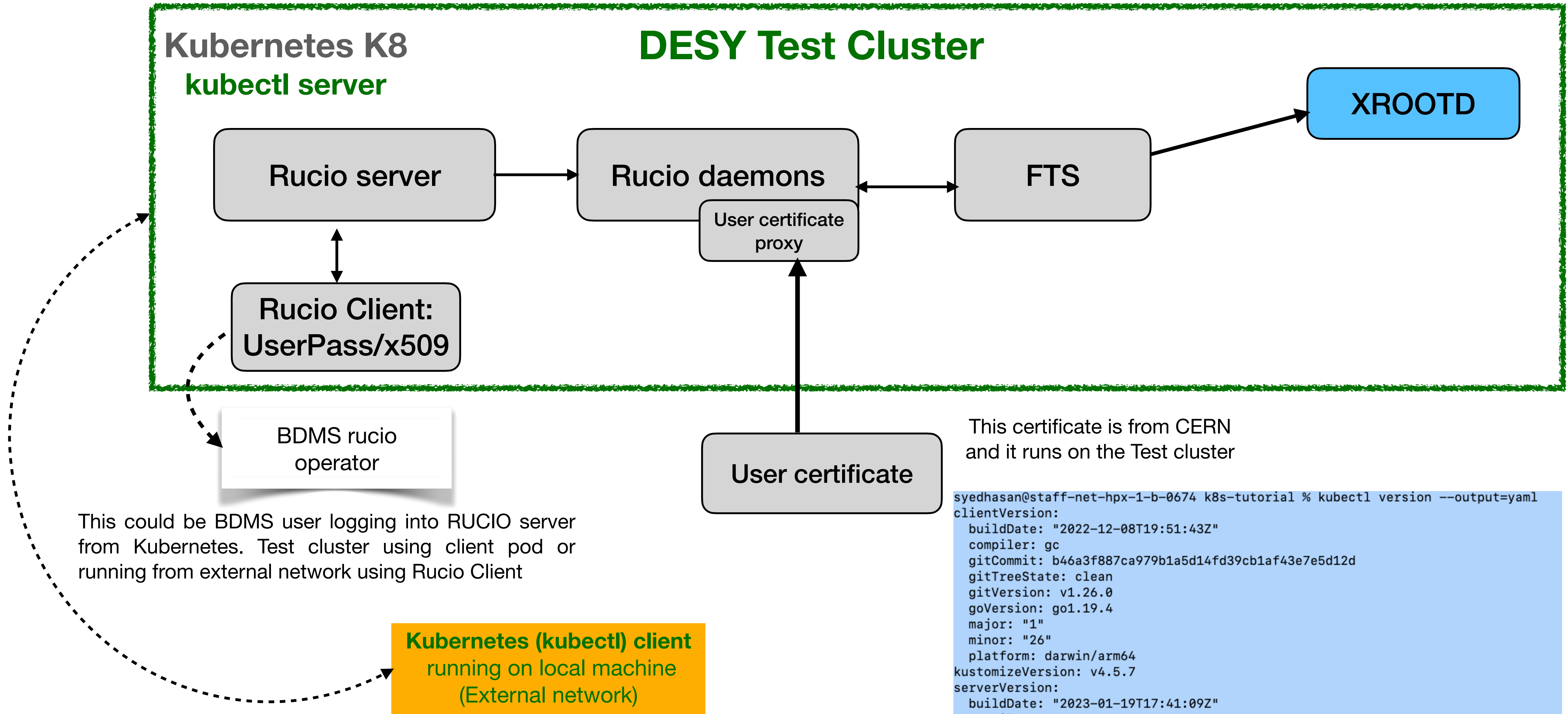# BDMS architecture (Functional decomposition)
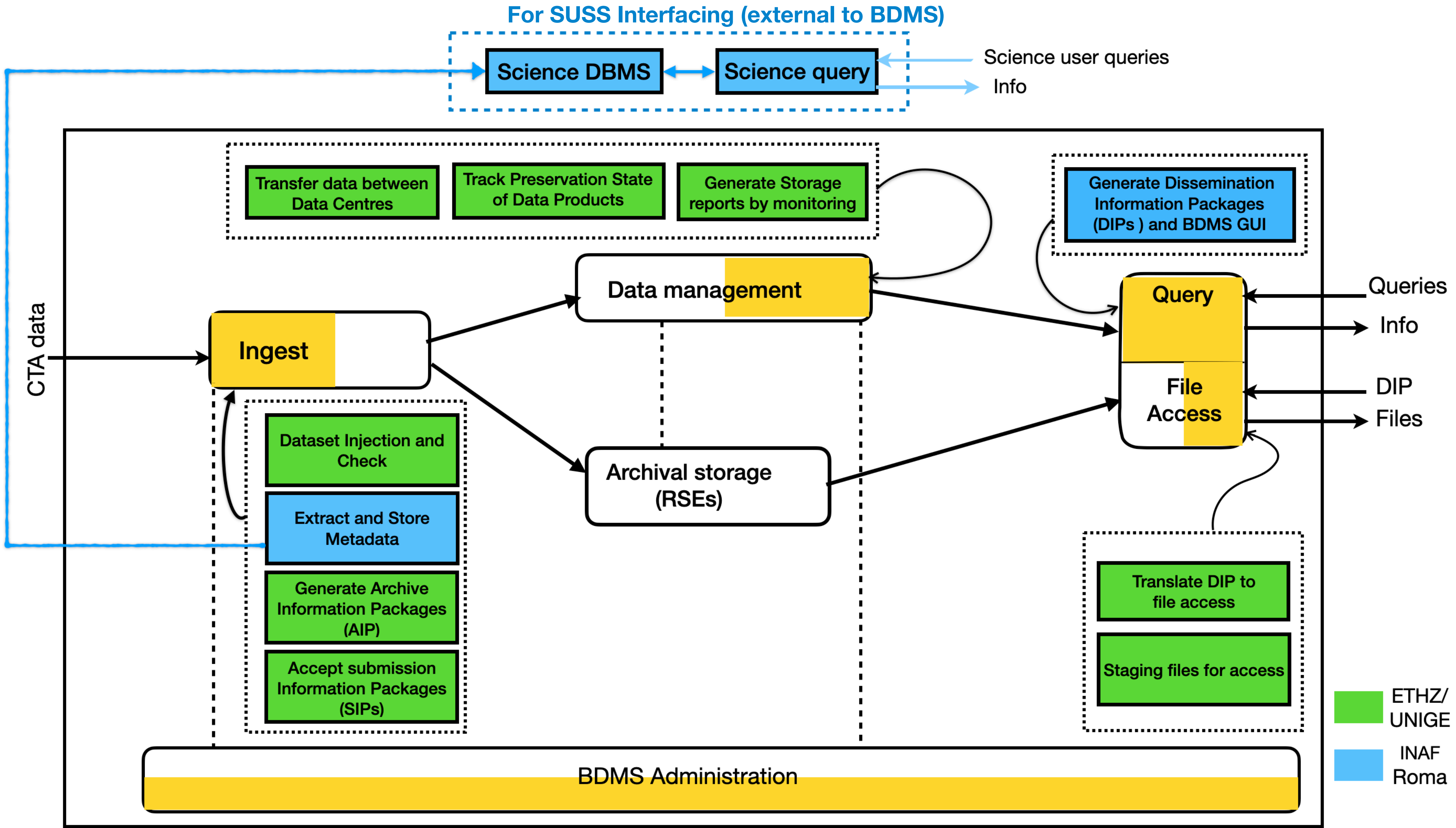
Swiss contribution to BDMS

- DPPS Use-cases for BDMS with Level B Requirements as inputs, Level C Requirements, and Architecture design

- Rucio tests with docker locally (ingest, replication, Rucio storage element (RSE) creation, rule deletion, file deletion)

- Bulk archive implementation using RUCIO deployed at CSCS Kubernetes server and storage elements and client at ETH Zurich - ingest, replication

- RUCIO deployment and testing/verifying BDMS use-cases for DPPS Release 0 at DESY Kubernetes cluster - ingest, query, and retrieval

- Supporting BDMS - WMS interfacing either via a dedicated interface from DIRAC to BDMS (ingest) or an interface between DIRAC and Query. BDMS-WMS interface is under development

- Provided inputs to Philipp from DESY AIV team to resolve networking issues for enabling RUCIO access from all external networks

- **Ongoing:**

  - BDMS prototyping with RUCIO and its services running as docker containers to speed-up development: adding interfaces and functional blocks to RUCIO core, CI/CD with gitlab runners, and tests

  - Later, we will translate this work to adapt to use RUCIO running at the Kubernetes cluster

BDMS prototyping efforts at DESY K8s cluster for Release 0 - Status

Kubernetes K8
kubectl server

DESY Test Cluster

XROOTD

Rucio server → Rucio daemons ↔ FTS

User certificate proxy

Rucio Client: UserPass/x509

BDMS rucio operator

This could be BDMS user logging into RUCIO server from Kubernetes. Test cluster using client pod or running from external network using Rucio Client

Kubernetes (kubectl) client
running on local machine
(External network)

User certificate

This certificate is from CERN and it runs on the Test cluster

```
syedhasan@staff-net-hpx-1-b-0674 k8s-tutorial % kubectl version --output=yaml
clientVersion:
  buildDate: "2022-12-08T19:51:43Z"
  compiler: gc
  gitCommit: b46a3f887ca979b1a5d14fd39cb1af43e7e5d12d
  gitTreeState: clean
  gitVersion: v1.26.0
  goVersion: go1.19.4
  major: "1"
  minor: "26"
  platform: darwin/arm64
kustomizeVersion: v4.5.7
serverVersion:
  buildDate: "2023-01-19T17:41:09Z"
  compiler: gc
  gitCommit: 5c1d2d4295f9b4eb12bfbf6429fdf989f2ca8a02
  gitTreeState: clean
  gitVersion: v1.24.10+rke2r1
  goVersion: go1.19.5 X:boringcrypto
  major: "1"
  minor: "24"
  platform: linux/amd64
```

BDMS prototyping efforts at DESY K8s cluster for Release 0 - Status

- Implemented a prototype of Ingest component with validation before ingesting files into RUCIO

  - Verification of **FITS** file using CFITSIO and FITS verify. Also, verification of **eventIO** files using ctapipe and eventio python packages

    - In the first step, the validation check is performed to extract good FITS files and eventio io files that will be ingested into RUCIO in the second step

    - Hancheng is writing the verification code and leading the efforts. We are verifying three kinds of data: FITS raw DL0 data, simulated data EventIO.zstd, and FITS auxiliary DL0 data, and extracting metadata from the headers of these files

    - Code: https://gitlab.cta-observatory.org/cta-computing/dpps/bdms/prototypes/swiss-prototypes/ingest
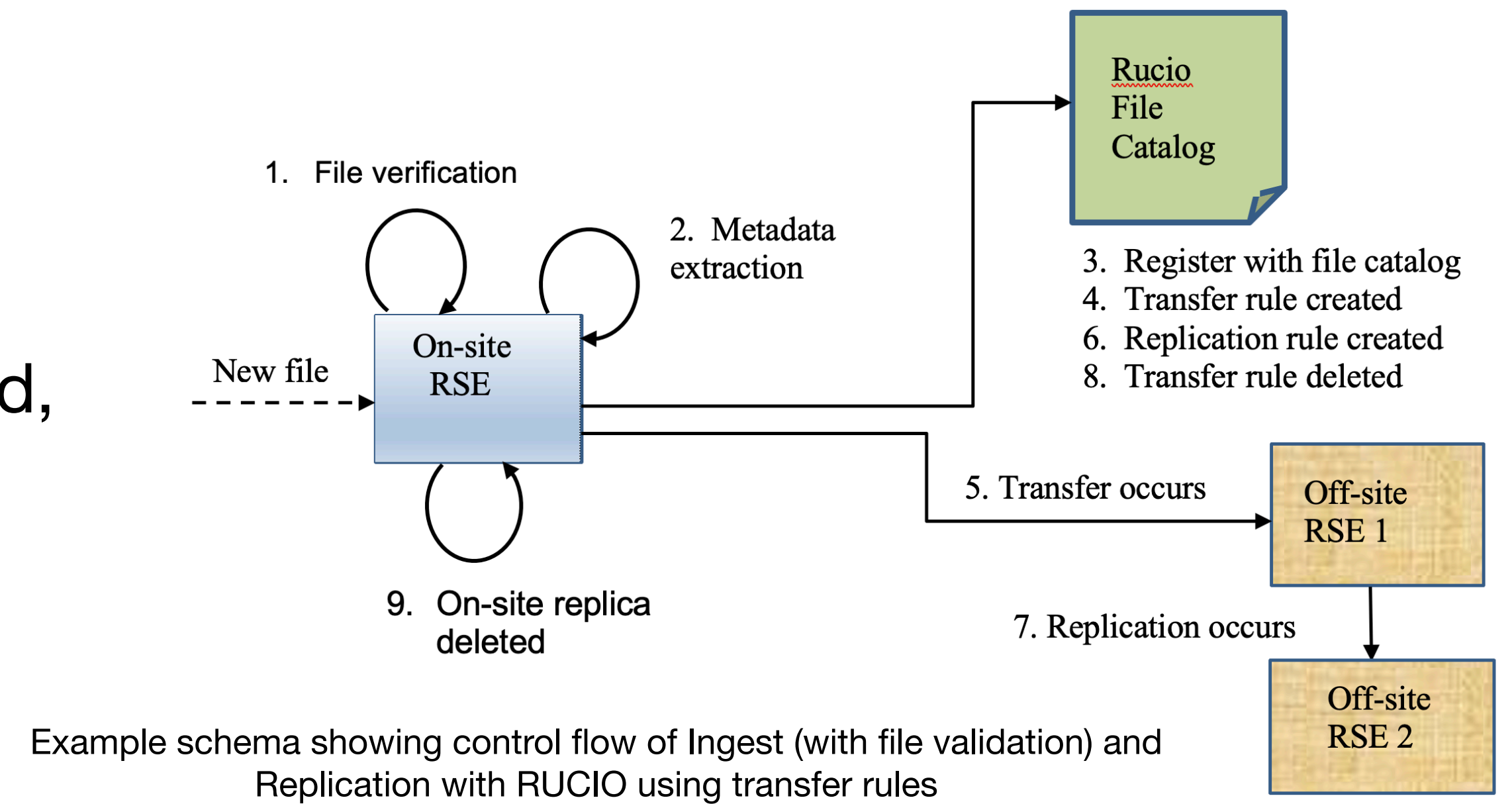
BDMS prototyping efforts at DESY K8s cluster for Release 0 - Status

- Towards making RUCIO deployment robust at the DESY Kubernetes Test cluster

  - Implemented XRootD storage elements (RSEs) persistent with some storage (Longhorn) at the test cluster so that data (and also installed libraries) saved there won't be lost achieving **data persistence**

    - Using Kubernetes persistence volume (PV) and persistence volume claims (pvc) features and attaching to all the XROOTD deployments (similar to mounting volumes in docker container to map host and container paths)

    - For now, we continue this way using **longhorn** from Kubernetes, another option for persistence will be achieved using **NFS storage** mounted from some file system

BDMS prototyping efforts at DESY K8s cluster for Release 0 - Status

- BDMS replication to two RSEs and deletion of file at origin RSE (long-haul scenario)

  - To test this scenario, the requirement is to use XrootD RSEs with AAA (authentication, authorisation, and possibly accounting) with **third-party copy (TPC) transfer** and **GSI (grid security) authentication** capabilities

  - **Solved the XrootD (with AAA) issue** by resolving the CA certificate issue both at the client (installing full IGTF certificate bundle) and also at the XrootD server (by adding the load balancer hostname as the DNS server in the host certificate generation). The final step in replication using FTS and poller daemon is to be resolved.

- We had already successfully tested the BDMS upload, download, and query scenarios using RUCIO

Example schema showing control flow of Ingest (with file validation) and Replication with RUCIO using transfer rules

BDMS prototyping at DESY Test cluster for Release 0 - Status

- Accessing and Testing/verifying BDMS RUCIO deployment from **external network**

  - This requires an external load balancer with static external IP and opening of desired ports

  - BDMS Rucio prototype access at K8s cluster is working fine from external networks. The new firewall rules and loadbalancer IP usage have been adopted now to work for all external networks

- Prototyping efforts to write unit tests to test RUCIO deployment and also deploy on gitlab

- Creation of a dedicated BDMS RUCIO clients docker image with IGTF and CA certs to make RUCIO server, storage elements (RSEs) and FTS at Kubernetes cluster accessible without any connection issues

- Supporting BDMS-WMS integration and AIV again once current prototyping efforts are completed

Work plan for the next six months

- Based on work-share agreements with INAF

  - Definition of interfaces with external subsystems (ACADA, SUSS, SOSS, A&A) and DPPS subsystem: WMS (specially to start working on an ICD for BDMS-WMS interfacing)

  - Current BDMS Bulk archive prototyping efforts (docker-based) to continue with a **demonstrator** to be set-up soon showing the archive end-to-end functionality including file retrieval based on metadata

  - Writing advanced tests for BDMS ingests including the ingests tests for example DL2/3/4 data produced by DIRAC

  - Working on data-shepherd to monitor status of datasets, detect anomalies and react to anomalies

  - BDMS Bulk archive prototyping and implementation to continue: docker-based with CI/CD and then translation to using RUCIO running at datacenter - DESY K8s cluster, and also collaboration with other data centres

  - Extending BDMS prototype to have monitoring support

  - Code reviews, documentation and improvements for long-term maintainability

Thank you for listening !

Backups

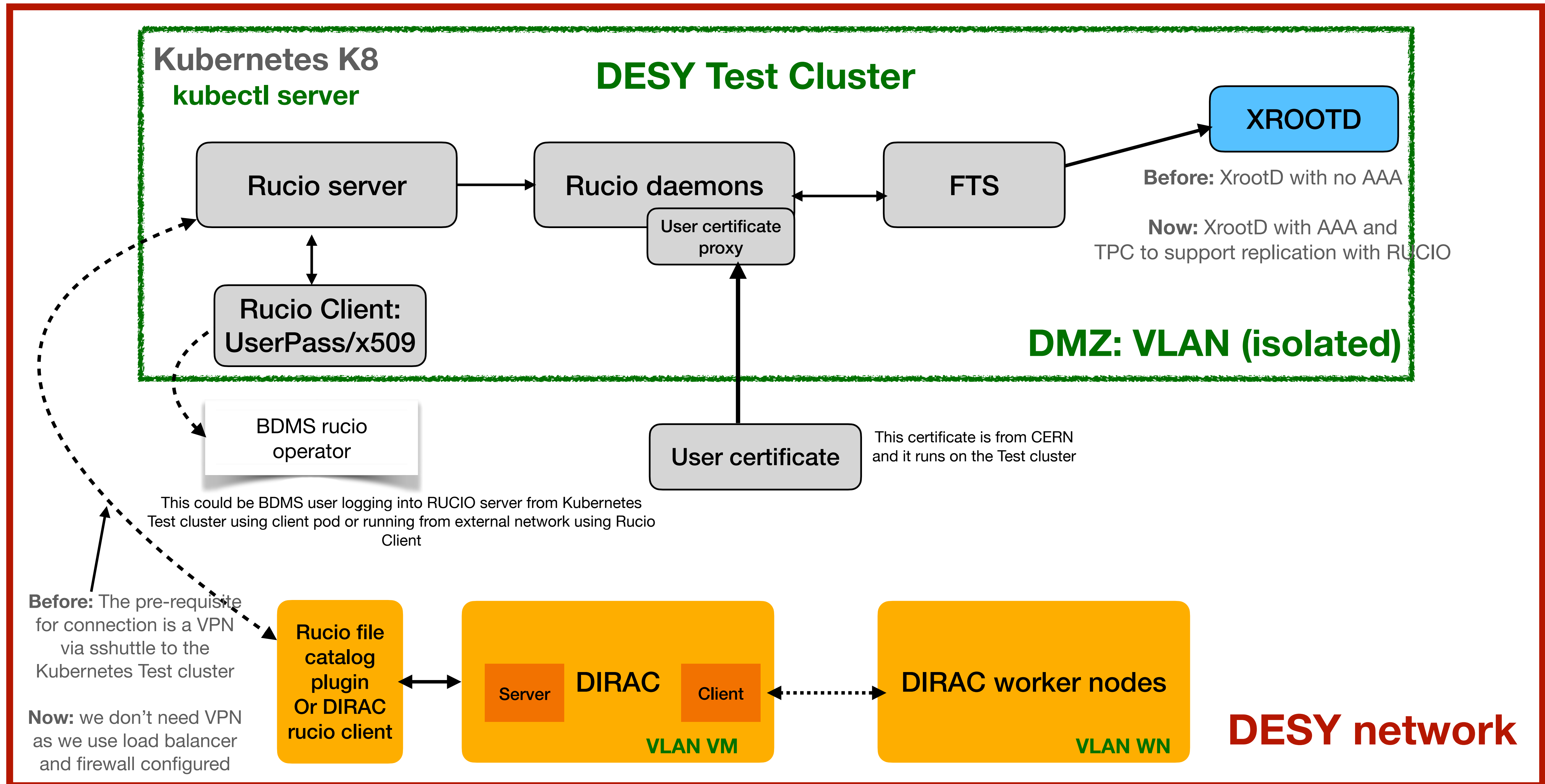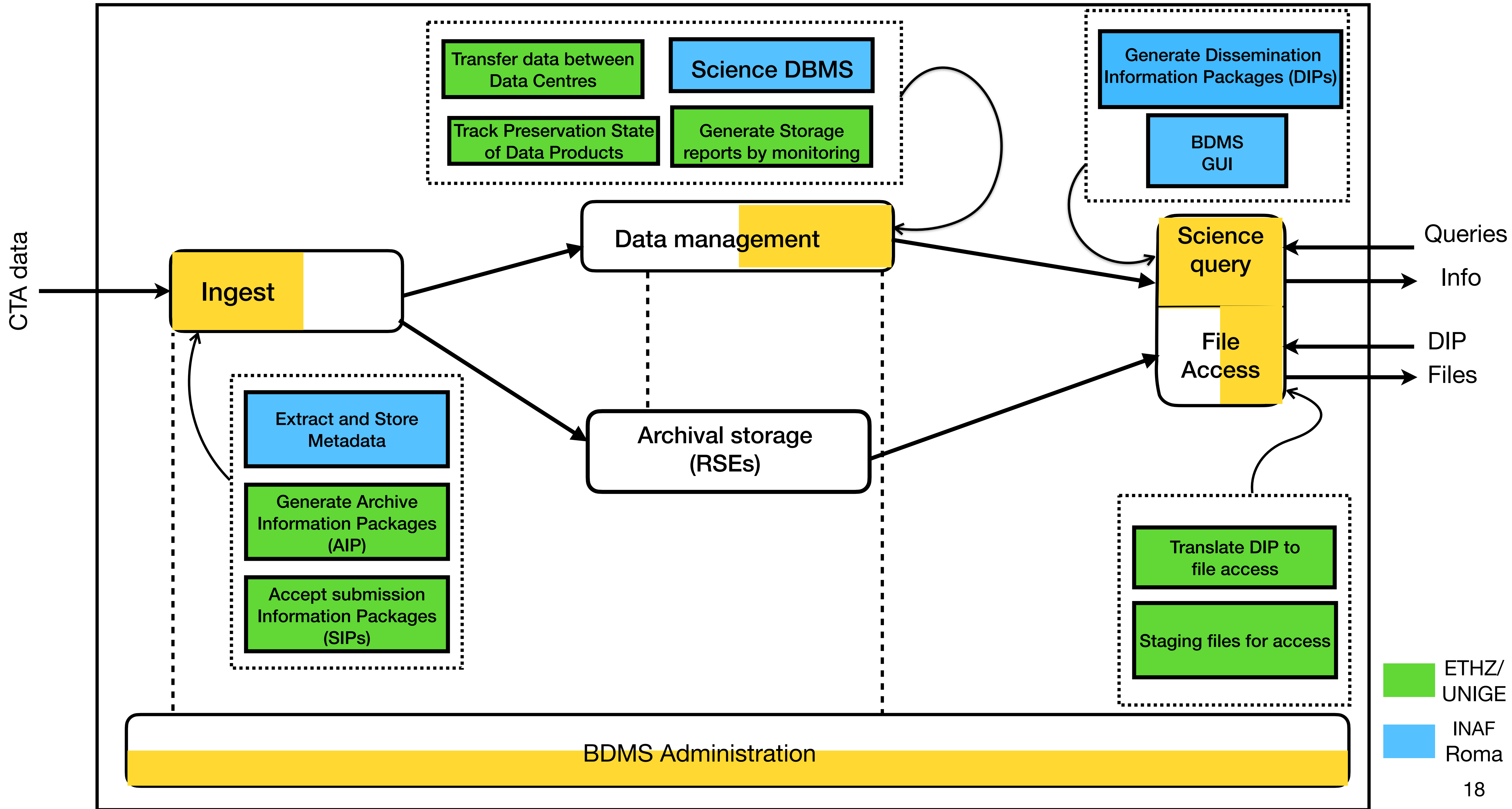# BDMS prototyping efforts in collaboration with CSCS
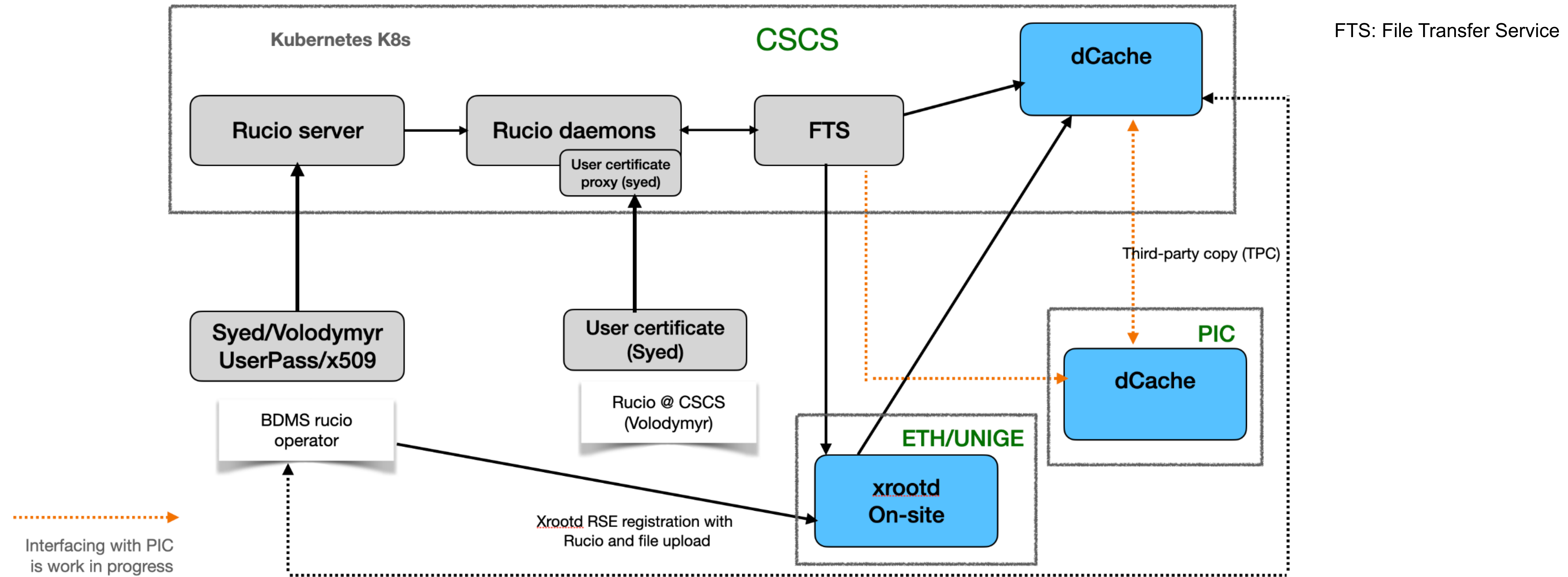
- Realistic set-up of Rucio storage element (dCache, xrootd) at CSCS, UNIGE (ETH for testing) with certificates and proxy certificates and successfully accessing them (read, write operations)

- Tested replication through RUCIO between UNIGE and CSCS using the same VO (virtual organization) proxy and different protocols (ssh, https, root) and replication

16

BDMS prototyping efforts at DESY Kubernetes cluster

Kubernetes K8
kubectl server

DESY Test Cluster

XROOTD

Rucio server → Rucio daemons ← → FTS → XROOTD

User certificate proxy

Before: XrootD with no AAA

Now: XrootD with AAA and TPC to support replication with RUCIO

Rucio Client: UserPass/x509

DMZ: VLAN (isolated)

BDMS rucio operator

User certificate

This certificate is from CERN and it runs on the Test cluster

This could be BDMS user logging into RUCIO server from Kubernetes Test cluster using client pod or running from external network using Rucio Client

Before: The pre-requisite for connection is a VPN via sshuttle to the Kubernetes Test cluster

Now: we don't need VPN as we use load balancer and firewall configured

Rucio file catalog plugin Or DIRAC rucio client

DIRAC
Server   Client
VLAN VM

DIRAC worker nodes
VLAN WN

DESY network

BDMS architecture - Work sharing

ETHZ/UNIGE

INAF Roma

18
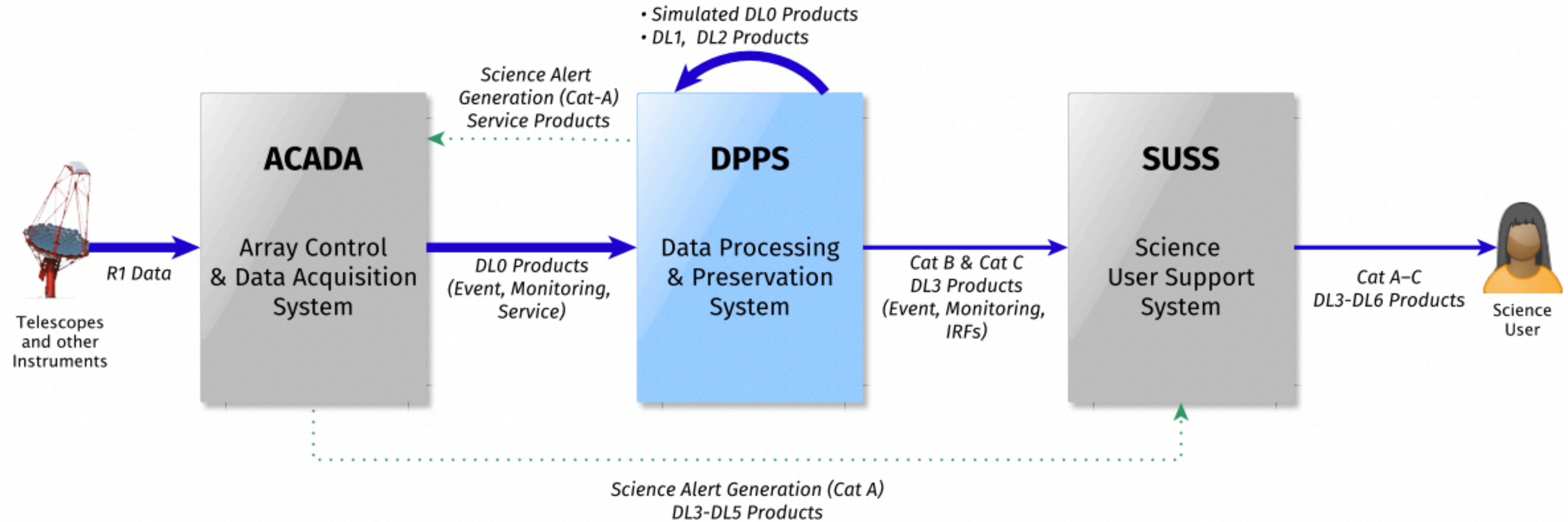
DPPS - Introduction

DPPS - Introduction

• Subsystems

source: DPPS Introduction, DPPS workshop at DESY (Zeuthen), 10-12th October, 2022
https://indico.cta-observatory.org/event/4313/contributions/35652/attachments/22328/31950/DPPS%20overview%202022.pdf