IOP: Birds eyes view

HUGO G. LABRADOR (CERN), 14TH JULY, CS3MESH4EOSC, TECHNICAL WORKSHOP

APIs and Protocols are different stuff

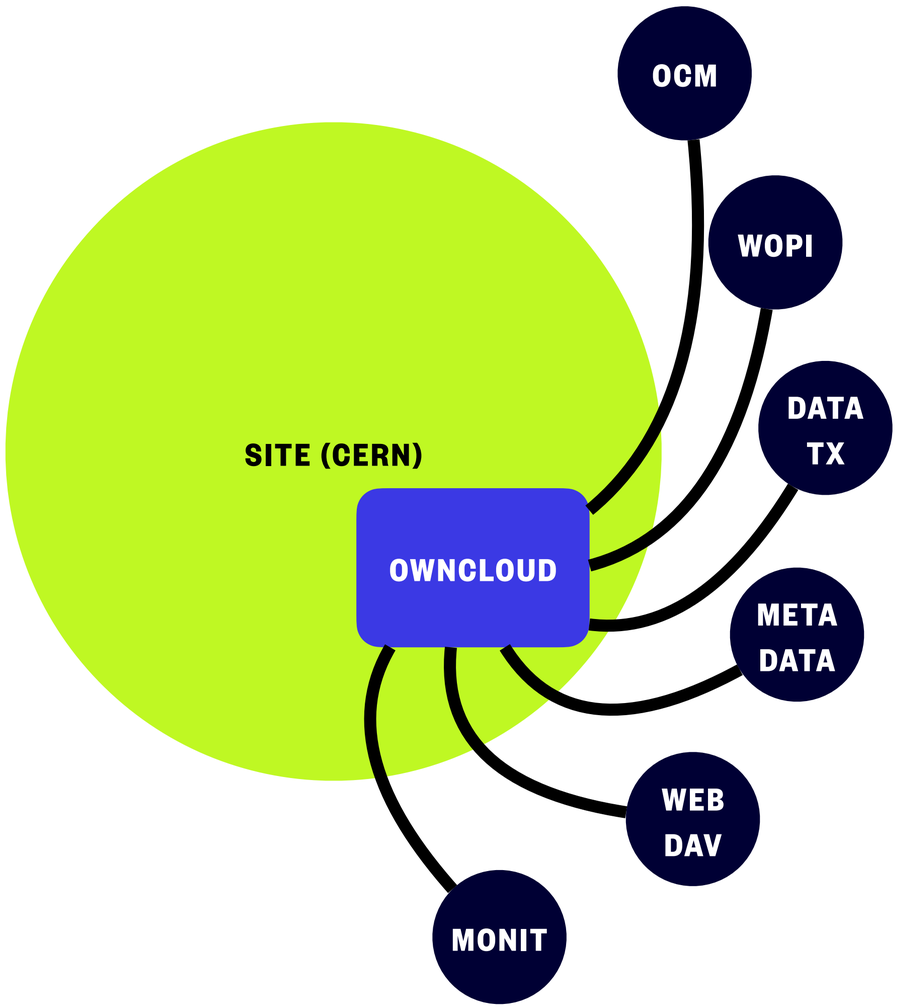
- **API** stands for Application Programming Interface, it refers to methods and data needed to interact with a component from an application you are programming
- A **Protocol** defines the way two entities communicate, the sequence of actions to achieve a goal, using one or more **APIs**

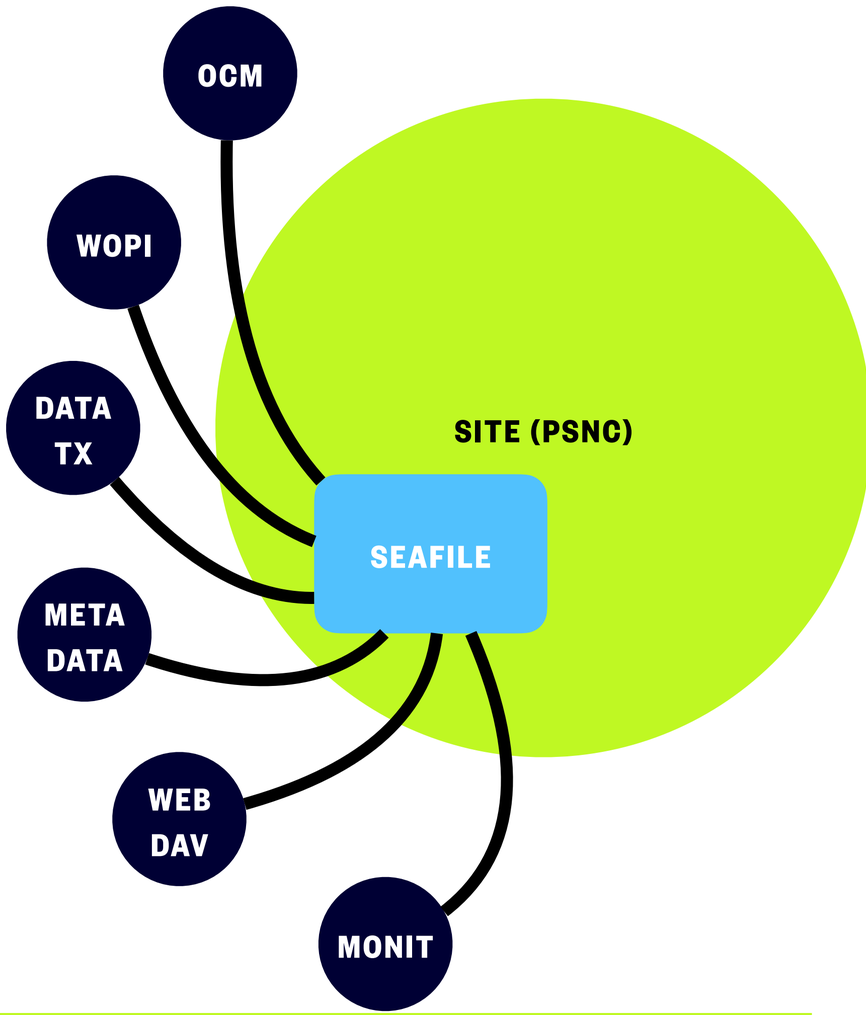
ScienceMesh Use-Cases

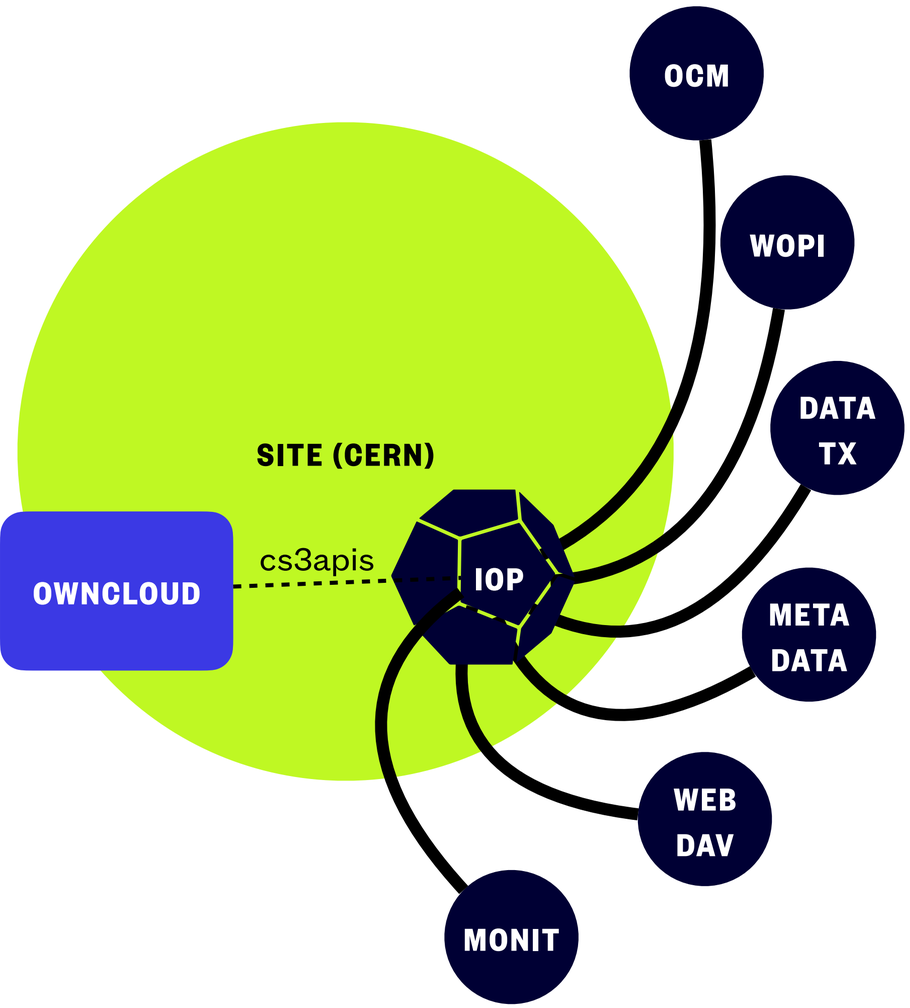
- Each use-case is backed up by one or more APIs and Protocols.
- **1) Data Science Environments**
- **2) Open Data Systems**
- **3) Collaborative Documents**
- **4) On-demand data transfers**

APIS that we need

- **OpenCloudMesh:** establishes a *rendezvous* to start exchanging "things" between sites using arbitrary protocols
  - **WebDAV** protocol is used to exchange data between sites
- **WOPI**: defines a protocol to allow concurrent and collaborative editing of **documents**, mostly used by Office-like applications like Microsoft 365/ Collabora
- API to perform on demand data transfers
- API to set rich metadata for research data management/digital repositories (tagging, ...)
- API for monitoring the health of the deployments
- ...

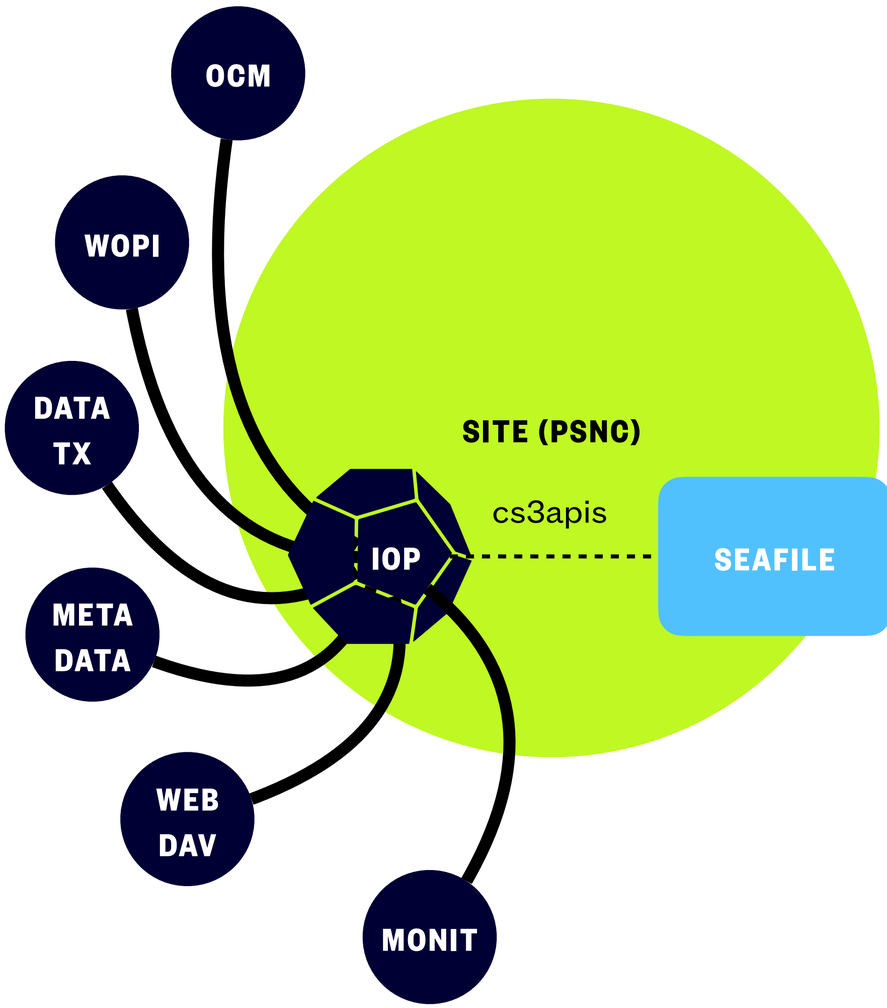

OCM
WOPI
DATA TX
SITE (CERN)
OWNCLOUD
META DATA
WEB DAV
MONIT

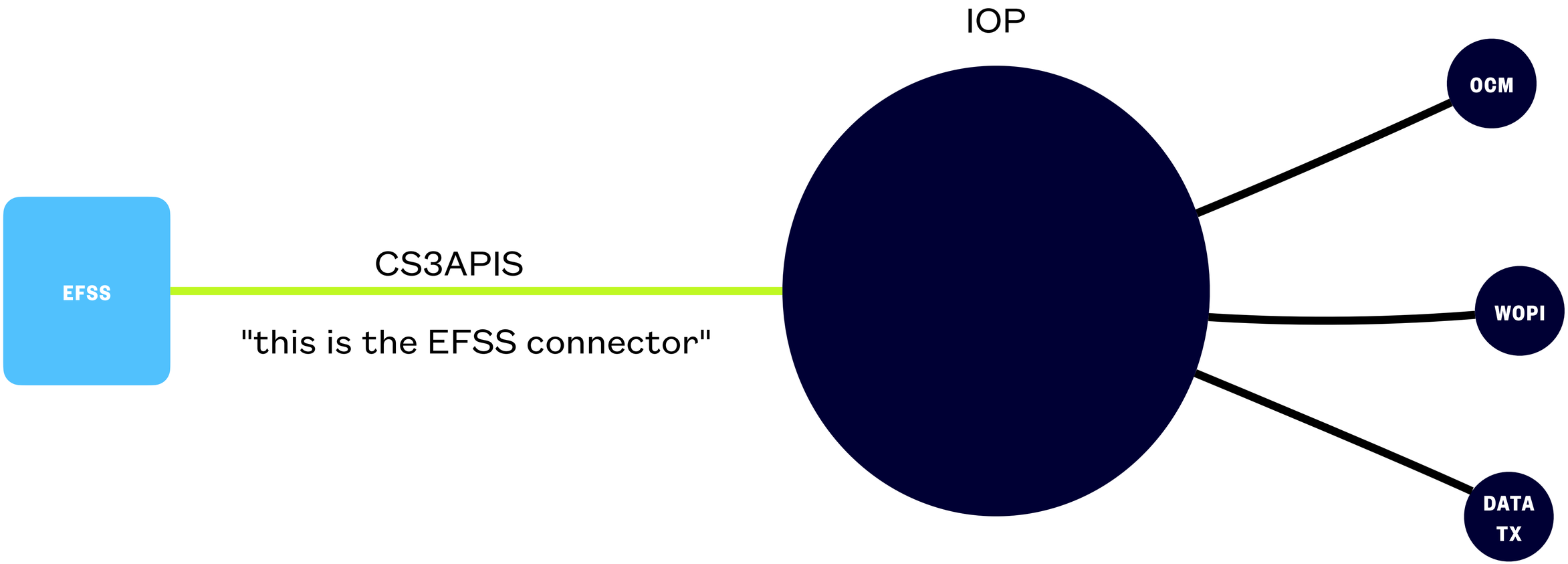

OCM
WOPI
DATA TX
SITE (PSNC)
SEAFILE
META DATA
WEB DAV
MONIT

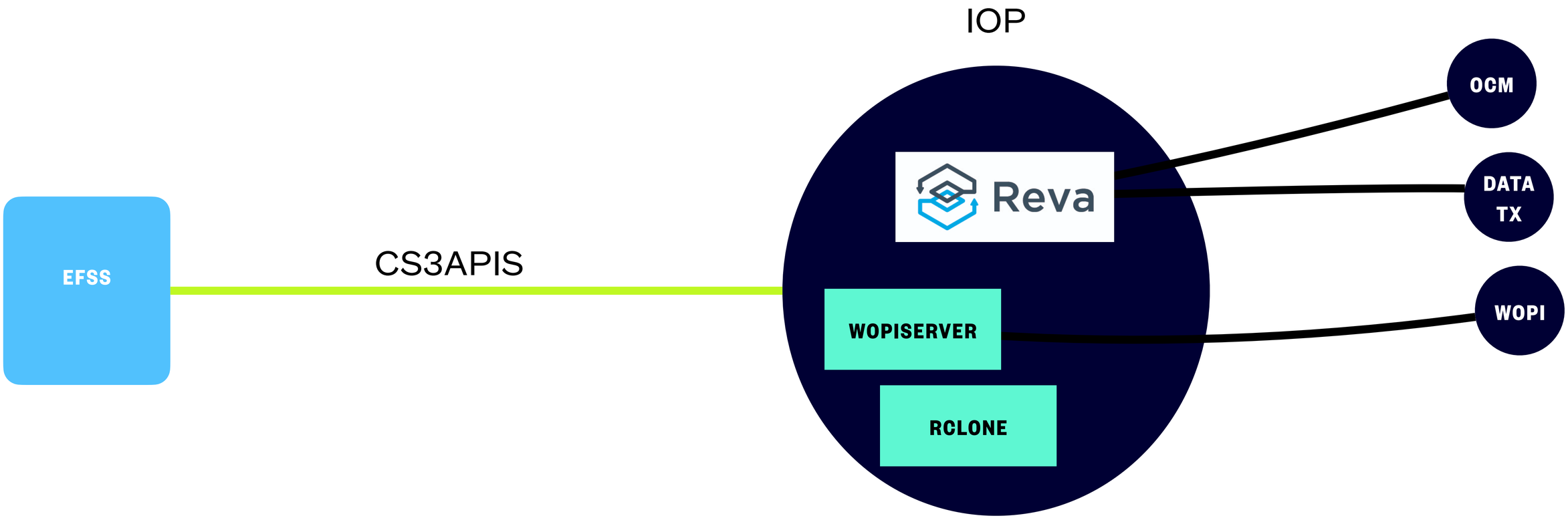


IOP is a middleware that adds the necessary APIS for interoperability and implements policies for healthy participation in the mesh

- timespan of the project (3 years) Feasible approach given
- Allows to gradually push protocols to vendors, no big bang approach
- Vendor-neutral, owned by community

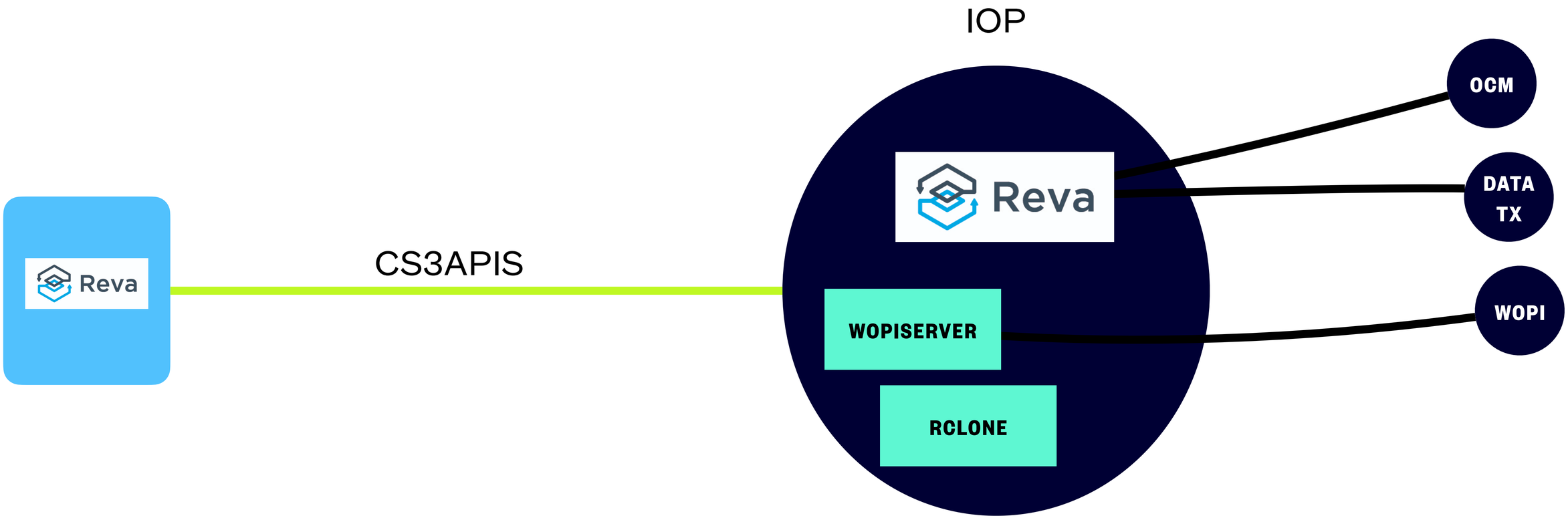



IOP
OCM
EFSS
CS3APIS
WOPI
"this is the EFSS connector"
DATA
TX



The IOP contains multiple components packaged into a K8S chart for easy deployment

```
helm install iop sciencemesh/iop \
  --set-file revad.configFiles.revad\\.toml=standalone.toml \
  --set-file revad.configFiles.users\\.json=users-cern.json \
  --set-file revad.configFiles.ocm-providers\\.json=providers.demo.json \
  -f custom-ingress.yaml
```

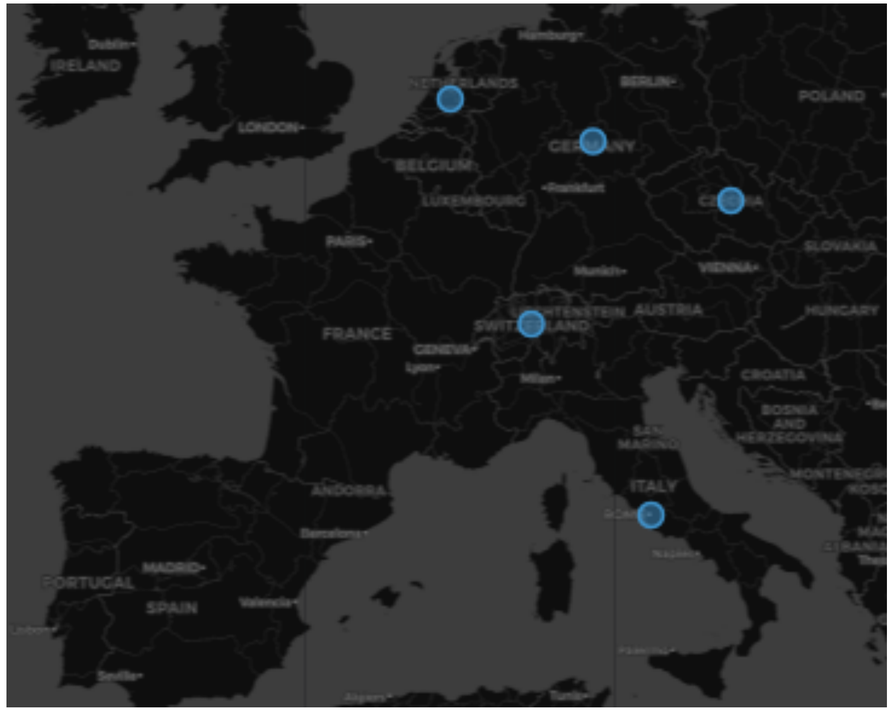


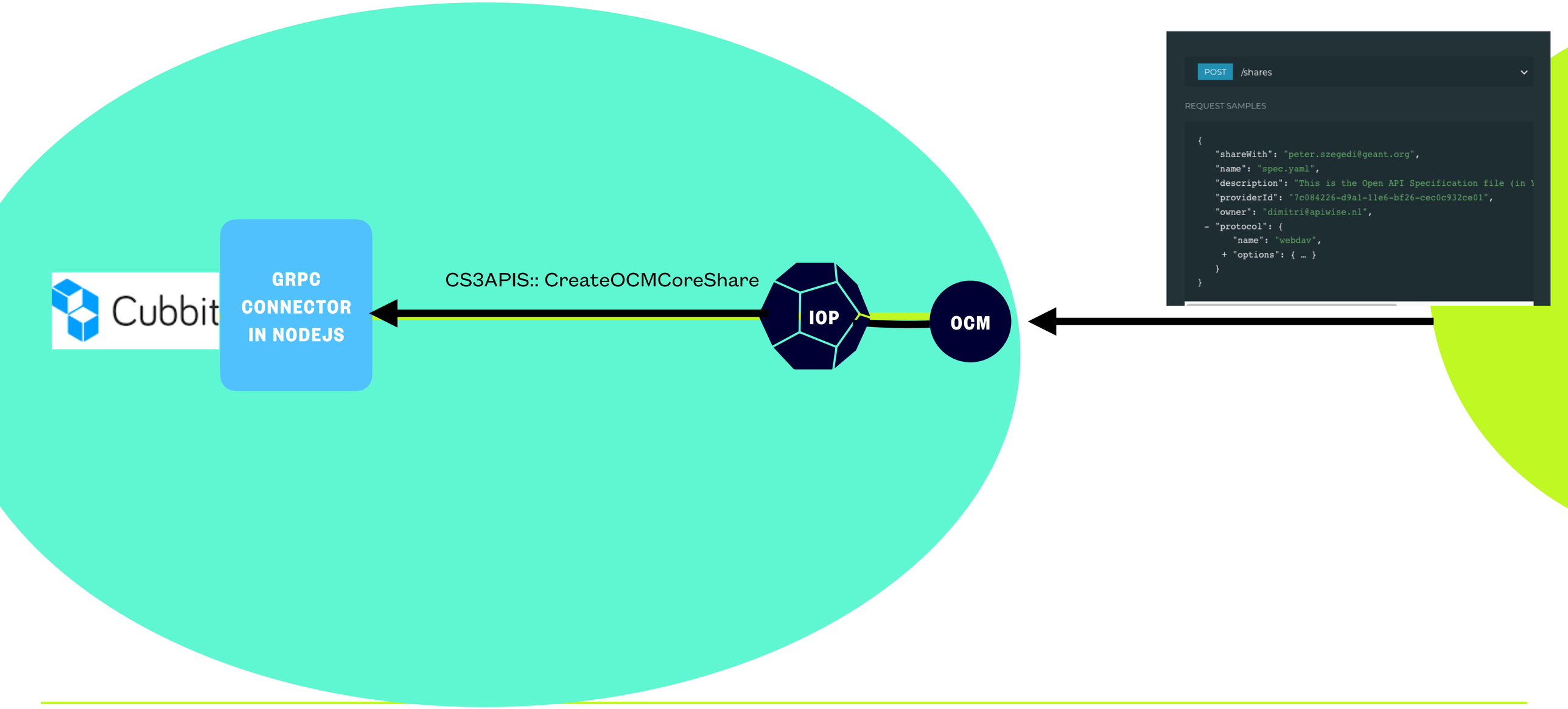
Reva has two purposes:

- Implements vendor-neutral APIs/Protocols for interoperability
- Reference implementation of the CS3APIS

Current State

- **Deploying IOP with artificial EFSS**
- **Testing OCM across deployments**
  - **from CLI**




Cubbit
GRPC
CONNECTOR
IN NODEJS
CS3APIS:: CreateOCMCoreShare
IOP
OCM
POST /shares
REQUEST SAMPLES
{
"shareWith": "peter.szegedi@geant.org",
"name": "spec.yaml",
"description": "This is the Open API Specification file (in
"providerId": "7c084226-d9a1-11e6-bf26-cec0c932ce01",
"owner": "dimitri@apiwise.nl",
- "protocol": {
"name": "webdav",
+ "options": { … }
}
}