

Resource Management Overview



EOX IT Services <https://eox.at>



EOfarm P.C. <http://eofarm.com>

Public Demo 2021-05-25

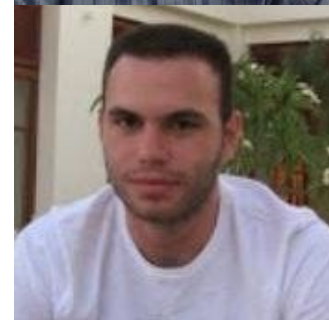


This work is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/).

Overview

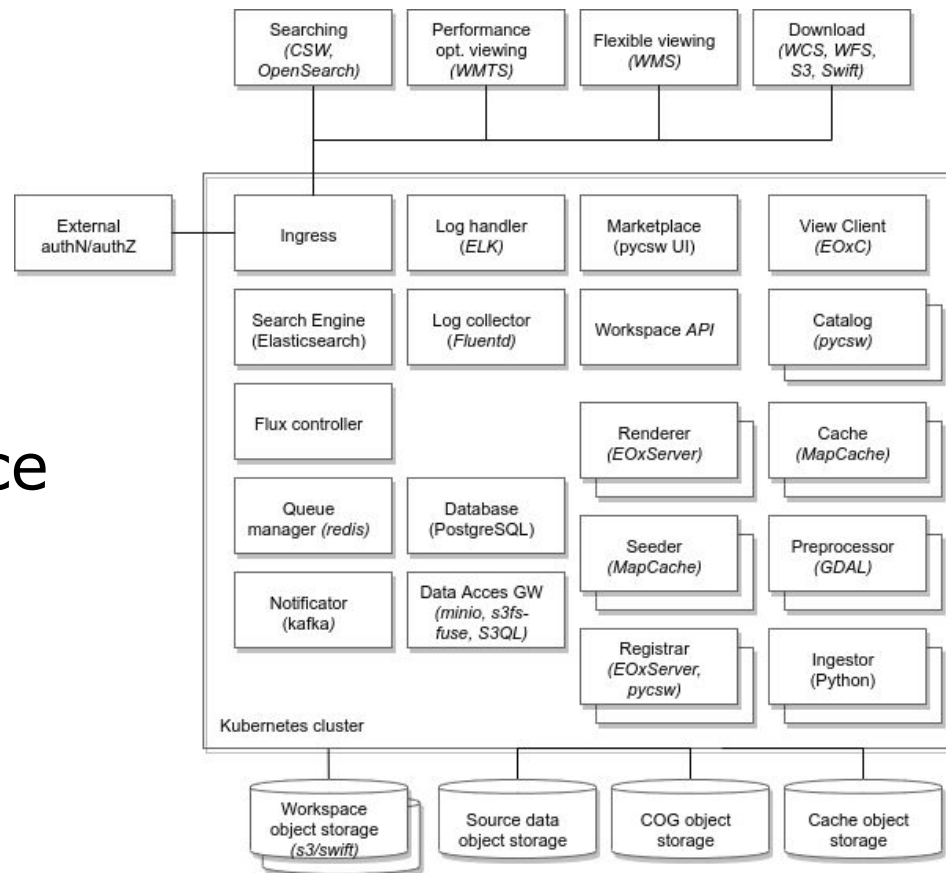
- Team
- Resource Management Architecture
- Resource Catalogue
- Data Access Service & Ingestion
- User Workspace
- Summary

Team



Resource Management Architecture

- Heavily based on [Python stack/ecosystem](#)
- Kubernetes mainly using helm charts
- Automation via flux
- System and per Workspace deployments



Resource Catalog Overview

- Discover/Search, Harvest, Distributed/Federated Search (user and system catalog)
- Metadata
- Standards
 - ISO 19115/19139
 - OGC CSW 2.0.2 / 3.0.0
 - Core Profile
 - ISO Application Profile
 - OGC OpenSearch Geo/Time, EO Profiles
 - OGC API - Records (in development)

Resource Catalog Technologies

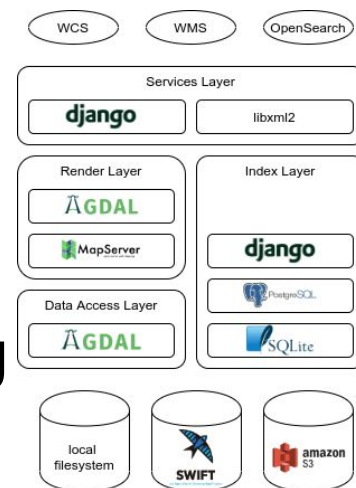
- [pycsw](#)
 - Python OGC Reference Implementation Server
 - Multiple metadata formats (ISO, DC, etc.)
 - Flexible backends (RDBMS)
- [OWSLib](#)
 - Swiss army knife Python client library for OGC Web Services
- [pygeometa](#)
 - Metadata composition
- [pygeoapi](#)
 - Python OGC Reference Implementation Server
 - evolving OGC API standards

Resource Catalogue Technologies

- PostGIS
 - PostgreSQL Spatial extension
 - OGC SFSQL
 - Metadata Store for pycsw, EOxServer
- Elasticsearch
 - Full Text Search Engine
 - Highly Distributed

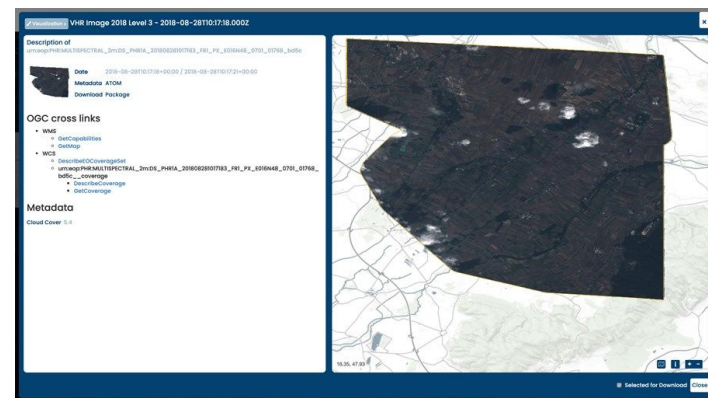
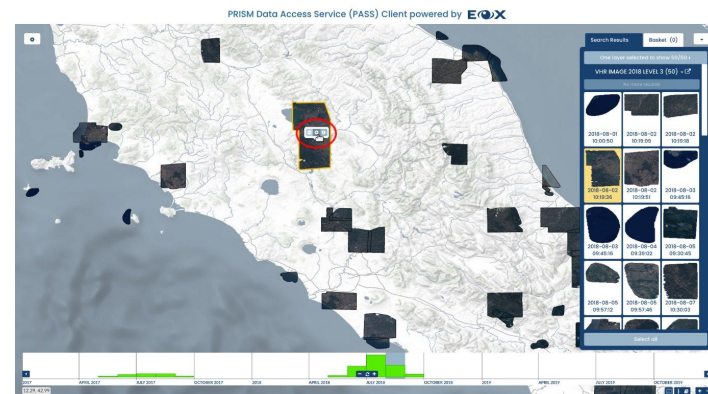
Data Access Service & Ingestion Overview

- View Server
- EOxServer as core
- Flexible WMS viewing interface (product outlines, masked validity, CQL)
- Performance optimized WM(T)S cached viewing
- Download interface - WCS or DSEO
- OGC OpenSearch Geo/Time catalog
- Ingestion (preprocessing products into COGs)
- Deployed in kubernetes cluster
- helm chart



Data Access Service & Ingestion Components

- Database - [PostgreSQL](#)
- Renderer - [EOxServer](#)
- Cache - [Mapcache](#)
- Registrar - [Python](#) + [EOxServer](#)
- Ingestor - [Python](#)
- Preprocessor - [GDAL](#)
- [Redis](#) queues
- Webclient based on [EOxC](#)
- [Elasticsearch](#)/[Fluentd](#)/[Kibana](#) logging stack

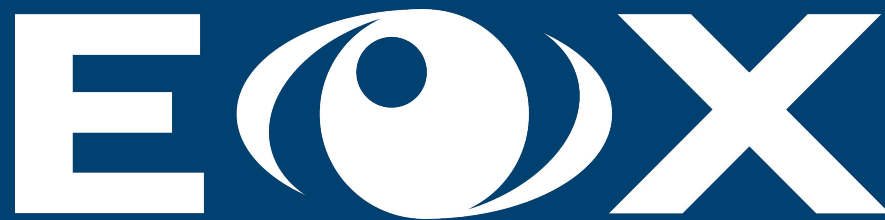


User Workspace

- EOEPKA Model
 - Object storage bucket
 - Data Access Service
 - Resource Catalog with federated system catalog
 - Deployment scenario(s) up to operator
- [Workspace API](#) based on FastAPI
 - create, delete, etc.
 - using “Bucket” claims via K8s CRD & K8s Secret
 - register data & processes
- [Bucket operator](#) for OpenStack

Summary

- Open Standards (OGC, ISO)
- Free and Open Source (FOSS)
 - Core components
 - Reference implementation
 - Geospatial (FOSS4G)
- Best of breed Python geospatial components
- Download, configure, and run
 - <https://github.com/EOEPCA/eoepca>
- Open communities
- Open to contributions



Thanks for your attention!



EOX IT Services <https://eox.at>



EOfarm P.C. <http://eofarm.com>

Public Demo 2021-05-25



This work is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/).