Dataverse in the European Open Science Cloud

Slava Tykhonov (DANS/CESSDA)
Philipp Conzett (UiT/CLARIN)
Marion Wittenberg (DANS/CESSDA)

Dataverse Community Meeting at Harvard University
20 June, 2019
SSHOC objective and deliverables

Objective

Development of a research data repository service on EOSC, for SSH institutions currently without such a facility for their designated communities

Deliverables

After 38 months: Data repository service running on EOSC

After 40 months: Report on principles of governance and sustainability of the data repository service
DataverseSSHOC project has two parallel tracks of the development:

- **Core development team** is working on the modification and extension of the Dataverse core functionality.
- **The application development** team will create new or will integrate existent tools that will be published on Dataverse App Store website.

Our goal is to build the distributed and mature data infrastructure based on sustainable microservices.
Development methodology

We follow SCRUM with quick but small updates, all new functionality will go through DTAP (development, testing, acceptance and production) pipeline:

local (own PC) – test (cloud) – acceptance (cloud) – production (cloud)
DataverseEU deployment in CESSDA Cloud

Docker Compose for the local development and testing
Kubernetes (K8s) for the deployment of services in the Cloud
(CI/CD pipeline with Jenkins and Helm)
Services in European Open Science Cloud (EOSC)

- EOSC requires the level 8 of maturity (at least)
- we need the highest quality of software to be accepted as a service
- clear and transparent evaluation of services is essential
- the evidence of technical maturity is the key to success
- the limited warranty will allow to stop out-of-warranty services
Testing process for the core and apps

Testing process follows the CESSDA maturity model
https://zenodo.org/record/2591055#.XKR6ny2B2u5

Important: every change of Dataverse functionality should be supplied with unit tests, changes of external functionality should get Selenium scenarios.

Goal: to score as high as possible according to CESSDA maturity model
Testing procedures

- Writing unit tests (Java) for core development team
- Integration tests (Selenium) for app development team
- Performance/stress tests (Apache JMeter) before Dataverse will go to Acceptance/Production
Dataverse App Store

Let’s build different services out of tools!

Data preview: DDI Explorer, Spreadsheet/CSV, PDF, Text files, HTML, Images, video render, audio, JSON, GeoJSON/Shapefiles/Map, XML

Interoperability: external controlled vocabularies (CESSDA CV Manager)

Data processing: NESSTAR DDI migration tool

Linked Data: RDF compliance including SPARQL endpoint

Federated login as a service (OAuth/Shibboleth in the same installation)
Applications maturity level

Every software package should follow the same CESSDA Maturity Model to be accepted as a service.

Must have: k8s infrastructure with upstream Docker images, warranty statement, documentation, unit tests, Selenium tests, jenkins pipeline

Running demonstration service will allow to create the connection to your own Dataverse
Multilingual support

DataverseEU will run Weblate as a service for the user interface, metadata schema and SOLR translation.

We’ve developed an experimental but adjustable pipeline for multilingual support that allows to download and synchronize all translations available in Dataverse Consortium github and provides easy access for translators to keep all properties up-to-date.
Weblate as Dataverse app
Controlled Vocabularies support
Use Cases

Who is going to benefit from SSHOC?

- European SSH researchers will be offered a Dataverse installation
- Many of the features to be developed in SSHOC will benefit also other Dataverse installations / communities
Use Case: Multilingual support

• Useful for the SSHOC Dataverse installation, e.g. French users can use French interface, Italian users can use Italian interface, etc.
• But also useful for existing Dataverse installations, e.g. DataverseNO
Use Case: Multilingual support

In DataverseNO we ideally need support for the two written standards of Norwegian:
- Bokmål
- Nynorsk

in addition to support for
- English
Use Case: Federated Log-in

Multiple authentication protocol support (enable use of both OAuth and Shibboleth in same installation)

For DataverseNO users outside Norway:

For DataverseNO users at Norwegian higher education and research institutions:

Log in with Feide

DataverseNO has requested you to log in with Feide.

Please, choose your affiliation:

UiT The Arctic University of Norway

Continue »
Use Case: Cloud support

We are considering to have DataverseNO being run in the cloud by UNINET, Norway’s national research and education network.

The Kubernetes approach in SSHOC will enable this kind of cloud deployment.
Use Case: CLARIN metadata compliance

Metadata for language resources and data exists in a multitude of formats.

To overcome this dispersion CLARIN has developed the Component MetaData Infrastructure (CMDI).

CMDI provides a framework to describe and reuse metadata blueprints of profiles.
Use Case: CLARIN metadata compliance

Will enable repositories like TROLLing (part of DataverseNO) to become fully CLARIN compliant.