The Global Dataverse Community Consortium

Supporting Dataverse repositories Around the World

Panel on Sustainable Architecture Options for Dataverse

Introduction

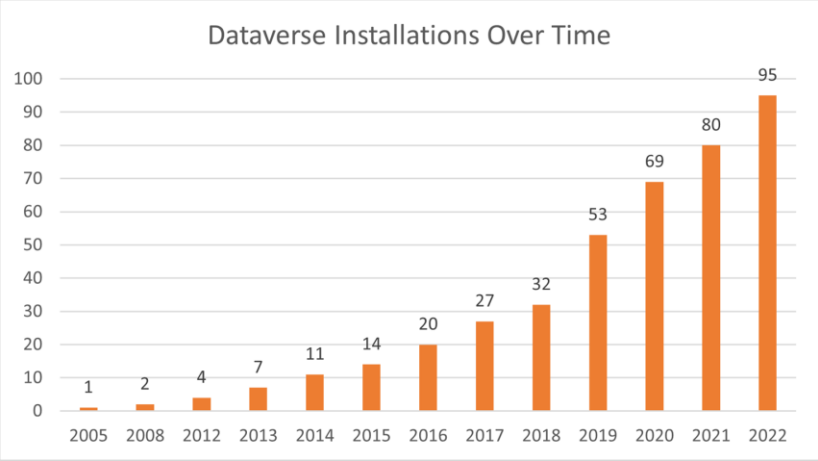
Philipp Conzett
UiT The Arctic University of Norway

Agenda

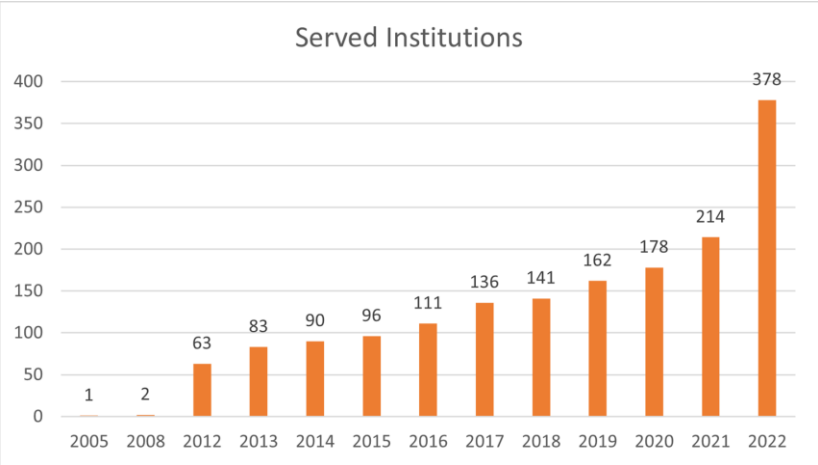
09:30	Introduction to the session and topic (**Philipp Conzett**, Senior Research Librarian, UiT The Arctic University of Norway)
09:45	Each panelist briefly (5-10 min.) introduces their take on preparation items 1-3 above: ● **Oliver Bertuch**, Research Software Engineer, FZ Jülich and Dataverse Core Team Member ● **Wim Hugo**, CTO, Data Archiving and Networked Services (DANS) ● **Stefano Iacus**, Director of Data Science and Product Research at the Institute for Quantitative Social Science, Harvard University ● **Rory Macneil**, CEO, RSpace ● **Jim Myers**, Senior Developer and Architect, GDCC and Dataverse Core Team Member
10:35	Panel discussion and questions/comments from other participants
10:50	Closing/Summing up/next steps
11:00	Coffee break

Background for this panel

The continuously growing popularity of the Dataverse software poses some challenges to the sustainable development of the Dataverse software and its ecosystem of associated tools and services.



(Based on https://iqss.github.io/dataverse-installations/, May 6, 2023)



(Based on https://iqss.github.io/dataverse-installations/ , and installation web pages May 6, 2023)

How to ensure sustainability?

- Working on open-source software sustainability, we need to address multiple aspects.
- The It Takes a Village project distinguishes between four main facets:
  - Governance
  - Technology
  - Resources
  - Community engagement

Figure: Arp, Laurie Gemmill & Megan Forbes. 2022. It Takes a Village: Open Source Software Sustainability. LYRASIS. Used under a CC BY 4.0 license.

Technology

Topics to be addressed:

- Decide on **architecture choices**. See, e.g., the examples used in the Dataverse Community Survey 2022:

Architecture Choice	Current Examples	Pros	Cons
a. Out of the box	• Authentication • Checksums • Embargo • Provenance • Versioning	• Things work more or less out of the box. • Has a few moving parts (e.g., one process, one app server, one database). As a result, it is easier to design, deploy, and test (system test, e2e test) the application. • Easy to manage transactions and data. sharing between features • Low operational complexity.	• Only configurable, cannot remove it. • Hard to adapt (fork or live with it). • Difficult to parallelize work among multiple teams. So, development scaling is challenging. • Granular scaling (i.e., scaling part of the application) is not possible. • Polyglot programming or polyglot databases are challenging.
b. Extension	• External controlled vocabularies • Previewers • Localization/ Internationalization	• Better development scaling as teams can work parallely on different features in a more autonomous way with little external dependency, thus good support for crowdsourcing. • Can be pluggable.	• Relies on community maintenance. • Difficult to coordinate. • Harder to sustain/sync the different parts. • Breaking changes. • Less seamless UI.
c. Loosely coupled integration (via API)	• Archivematica • Data Curation Tool • Open Journal System • Whole Tale	• Greatest degree of flexibility and freedom.	• Relies on external maintenance. • Same as choice b), but stronger.

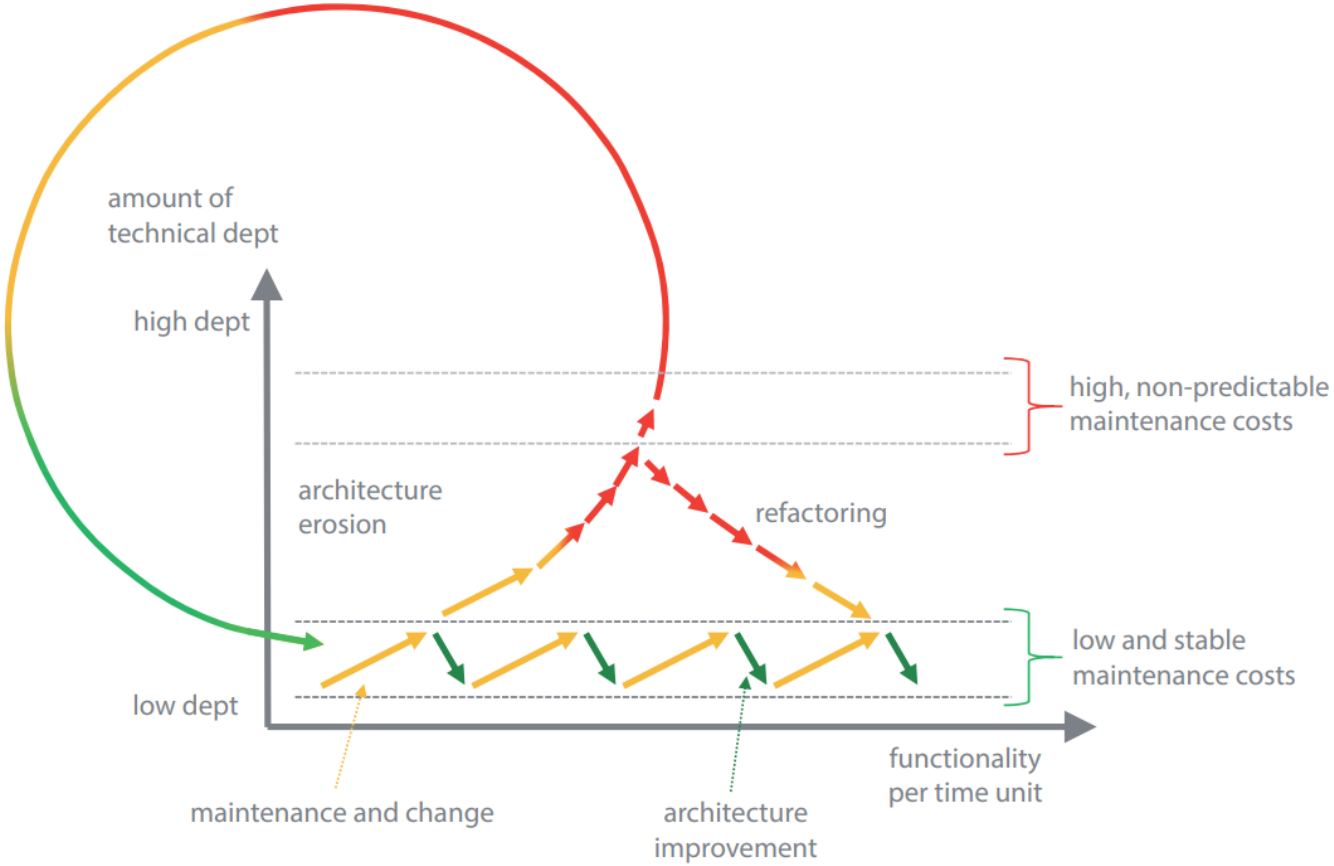
Adapted from Conzett, Philipp. 2022. Dataverse Community Survey 2022 – Report. Septentrio Reports. https://doi.org/10.7557/7.6872, p. 43-44.

Technology

Topics to be addressed:

- Decide on **architecture choices**
- Establish rules for **contribution**
- Manage **maintenance**
- Continuous **development**
- Avoid/mitigate **technical debt**
- **Refactoring** when needed
- Consider **re-architecture** if necessary

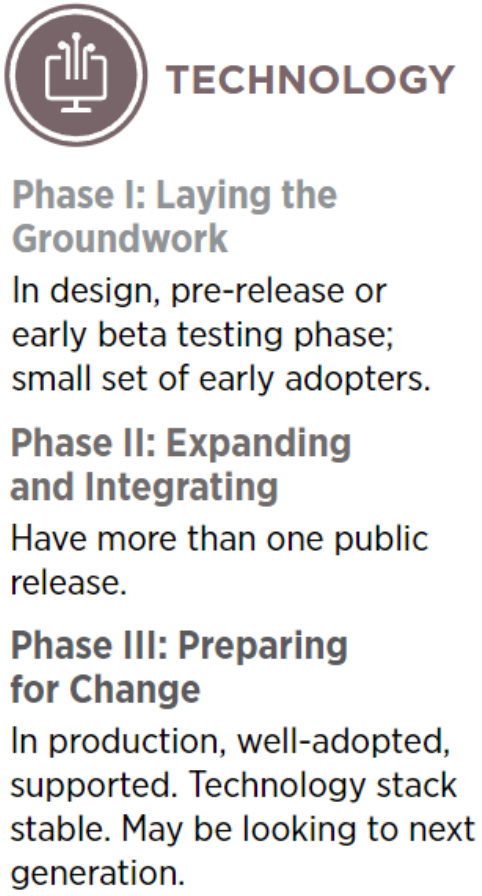
Technical debt and architecture erosion



Source: Lilienthal, Carola. 2019. Sustainable software architecture: analyze and reduce technical debt. dpunkt.verlag, p. 5. All rights reserved dpunkt.verlag

Sustainability throughout the life cycle

"... sustainability strategies evolve as the [Open-Source Software] life cycle progresses, technology advances, and community needs change." (Arp & Forbes. 2022, p. 7)



From: Arp, Laurie Gemmill & Megan Forbes. 2022. It Takes a Village: Open Source Software Sustainability. LYRASIS. Used under a CC BY 4.0 license.

Core goals in different phases

Phase I: Laying the Groundwork

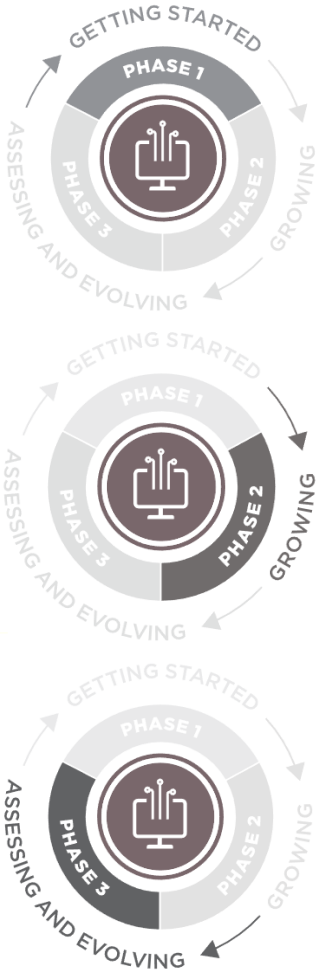
- Turn an idea for an application into a **viable product** that serves the needs of the community

Phase II: Expanding and Integrating

- **Refine the application**: identify and strengthen areas that are working well, identify gaps that can be filled with new features and functionality, and phase out elements that are not working.

Phase III: Preparing for Change

- Determine how the core application's technology stack and functionality will serve the future needs of the community; **plan ahead for expansion, integration, re-architecture, or retirement**



What about Dataverse?

- In different phases depending on facet (cf. Governance, Technology, Resources, Community Engagement) and specific issues?
- Let’s use this panel to address technical aspects of the sustainability of the Dataverse software.
- In the working group session on Dataverse sustainability after lunch, we’ll be addressing also other aspects of sustainability: Governance, Resources, and Community Engagement.

Goal of this panel

Address the pros and cons of possible architecture choices conceivable for the Dataverse software to ensure the sustainable development of the software, including providing sustainable support for

- integrations
- deployment modes
- storage options
- file management systems
- data types
- (meta)data structures
- adaptations
- etc.

Preparation items

The panelists and the audience were asked to

1. **identify and list/rank needs** in respect of extensions and improvements to Dataverse;
2. **express preferences** for the rework options and/or identify additional options;
3. **match requirements and needs with options** considering risk, costs, and benefits.

Welcome to our panelist!

- **Oliver Bertuch**, Research Software Engineer, FZ Jülich and Dataverse Core Team Member
- **Wim Hugo**, CTO, Data Archiving and Networked Services (DANS)
- **Stefano Iacus**, Director of Data Science and Product Research at the Institute for Quantitative Social Science, Harvard University
- **Rory Macneil**, CEO, RSpace
- **Jim Myers**, Senior Developer and Architect, GDCC and Dataverse Core Team Member