Building a Free, Open-Source Data Repository to Support Distributed Interdisciplinary Science

Edward Flathers, Paul Gessler, Erich Seamon
University of Idaho College of Natural Resources
flathers@uidaho.edu

<Open Source: Software and Science>

According to the Free Software Foundation, a program is free software if the program's users have the four essential freedoms:

- The freedom to run the program as you wish, for any purpose
- The freedom to study how the program works, and change it so it does your computing as you wish
- The freedom to redistribute copies so you can help your neighbor
- The freedom to distribute copies of your modified versions to others. By doing this you can give the whole community a chance to benefit from your changes.

[From http://www.gnu.org/philosophy/free-sw.html accessed 25 October 2015]

The principles of open science enjoy significant overlap with free software. The journal *Scientific Data*, a product of *Nature*, describes openness among its six founding principles:

"We believe scientists work best when they can easily connect and collaborate with their peers, *so Scientific Data* aims to:

- Offer transparency in experimental methodology, observation and collection of data
- Use open licenses that allow for modifications and derivative works
- Break down barriers to interdisciplinary research — facilitating understanding, connectivity and collaboration
- Ensure all interested parties — scientists, policy-makers, NGOs, companies, funders and the public — can find, access, understand and reuse the data they need."

[From http://www.nature.com/sdata/about/principles accessed 25 October 2015]

Clearly, the principles of freedom to re-use, produce derivative works, and redistribute are causes common to both the FOSS movement and proponents of open science.

<Standards>

Some of the organizations that have developed standards that are supported or implemented by the repository include:

The Consultative Committee for Space Data Systems

International Organization for Standardization

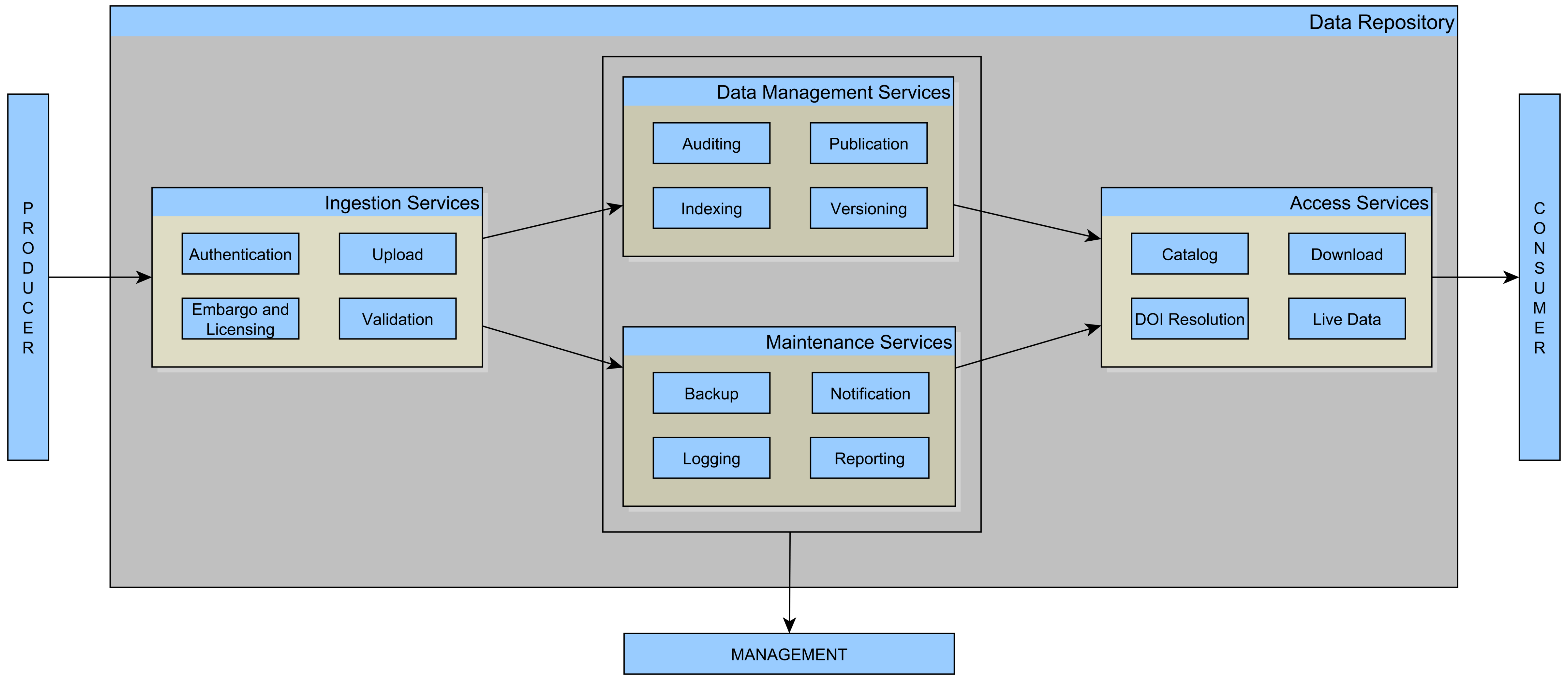
OGC Making location count.

<sword />

OPEN ARCHIVES

FederalGeographicDataCommittee fgdc

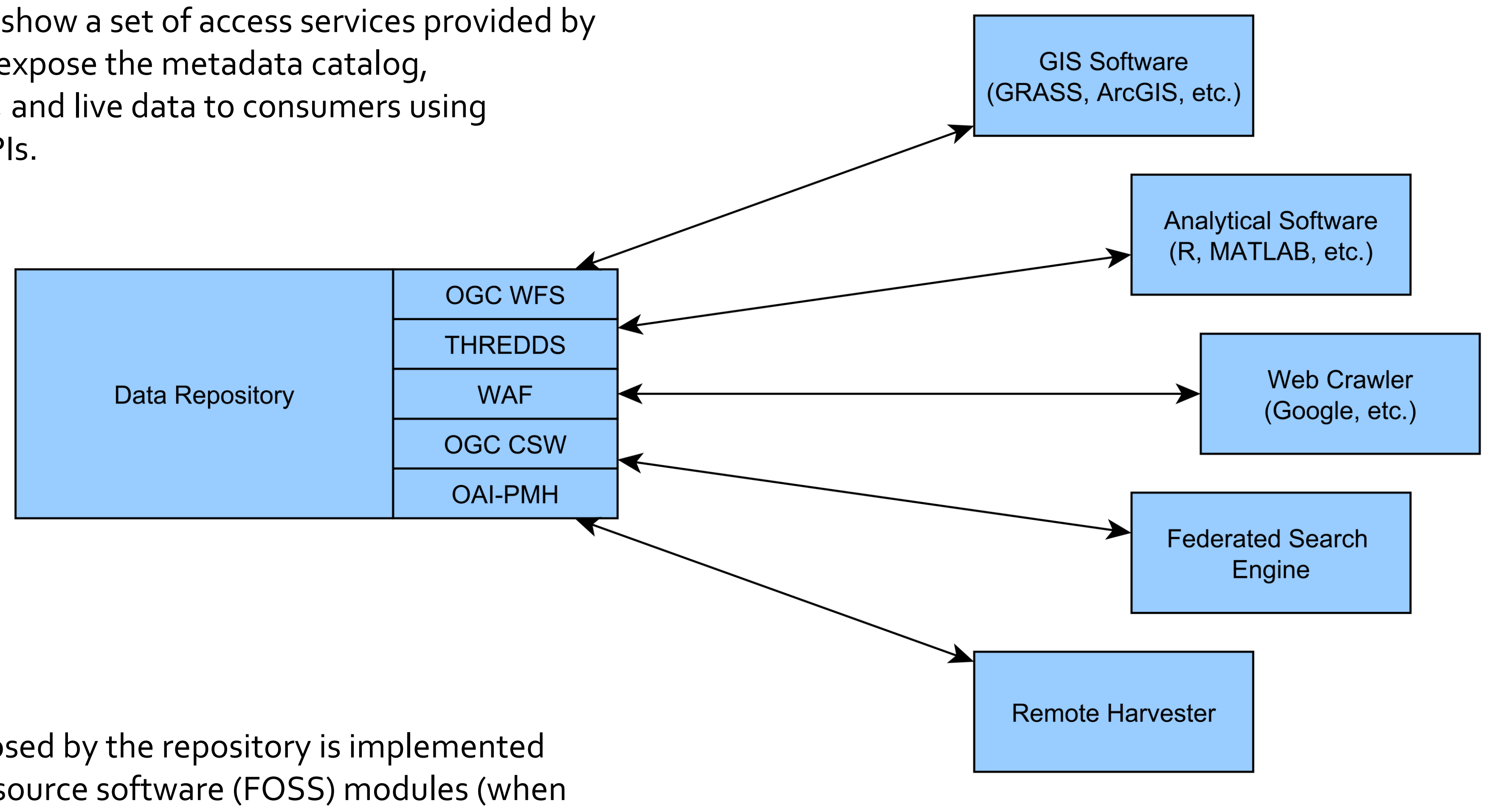
<The Repository Model>



After CCSDS, January. Reference model for an open archival information system (OAIS). CCSDS 650.0-B-1, Blue Book, 2002.

<Access Services>

In this example, we show a set of access services provided by the repository that expose the metadata catalog, downloadable data, and live data to consumers using standards-based APIs.


Data Repository
OGC WFS
THREDDS
WAF
OGC CSW
OAI-PMH
GIS Software (GRASS, ArcGIS, etc.)
Analytical Software (R, MATLAB, etc.)
Web Crawler (Google, etc.)
Federated Search Engine
Remote Harvester

Each service exposed by the repository is implemented using free, open-source software (FOSS) modules (when available). For example, the Open Geospatial Consortium Catalog Service for the Web (OGC-CSW) service is provided by a module called PyCSW, a FOSS project available on GitHub.

One benefit of adhering to standards during implementation is interoperability: client applications that support the standards will work with the repository despite designers having no foreknowledge of its existence.

Data consumers (above right) might be human or machine actors.

- Humans tend to interact with repositories using client software such as web browsers, GIS, and analytical software
- Machines tend to access repositories in order to harvest data or metadata

<Service-Oriented Architecture>

Service Oriented Architecture (SOA) is a design paradigm from the computer sciences that describes building modular, loosely-coupled software systems (Papazoglou). The modules, or services, that are deployed in such a system may exist in geographically disparate locales; they may be created and maintained by separate institutions or groups, and they may rely on entirely different computing hardware and software. The loose mode of coupling is usually organized by an Application Programming Interface (API) that defines the language and communication protocol that the service "speaks". As long as a service properly implements the requirements of the API, it can interoperate with other systems that speak its language.

Using a collection of modular services in place of a monolithic, one-size-fits-all repository software package allows simple replacement of individual system components. In turn, this can:

- Enhance system reliability
- Support interoperability with external systems
- Enable staggered rollout of new features
- Reduce dependence on pre-existing software
- Separate development into manageable tasks

Papazoglou, MP, and WJ Van Den Heuvel. 2006. "Service-Oriented Design and Development Methodology." International Journal of Web Engineering and Technology, 1–17.

<Summary>

Using free. open-source software for repository implementation

- Matches principles of openness and sharing between the free software and open science communities
- Supports FOSS, commercial, and custom data consumers
- Meshes with the SOA approach to modular systems

<Acknowledgments>

This material is based upon work that is supported by the National Institute of Food and Agriculture, U.S. Department of Agriculture, under award number 2011-68002-30191,. and with support from these organizations:

University of Idaho
REACCH Regional Approaches to Climate Change – Pacific Northwest Agriculture
Idaho EPSCoR
NIFA
USDA ars
Washington State University
NKN Northwest Knowledge Network
USGS science for a changing world
OSU Oregon State University