FLAT:
A CLARIN-compatible repository solution based on Fedora Commons

Paul Trilsbeek
The Language Archive
Max Planck Institute for Psycholinguistics
Nijmegen, The Netherlands
Paul.Trilsbeek@mpi.nl

Menzo Windhouwer
Meertens Institute
KNAW Humanities Cluster
Amsterdam, The Netherlands
menzo.windhouwer@meertens.knaw.nl

Abstract

This paper describes the development of a CLARIN-compatible repository solution that fulfils both the long-term preservation requirements as well as the current day discoverability and usability needs of an online data repository of language resources. The widely used Fedora Commons open source repository framework, combined with the Islandora discovery layer, forms the basis of the solution. On top of this existing solution, additional modules and tools are developed to make it suitable for the types of data and metadata that are used by the participating partners.

1 Introduction

The Language Archive (TLA) has been using an in-house built repository solution for more than 15 years now. During these years, this solution has grown into a rather complex set of tools and services that each take care of a specific function related to archiving, preservation, discoverability and usability of the resources contained in the repository. This complex set of tools and services has been developed by a large number of software developers over the years, using a variety of programming paradigms and frameworks. This makes the solution rather expensive to maintain.

The Meertens Institute has been creating collections of language resources for many years and typically developed custom exploration tools for each collection. It used an in-house developed tool for managing these, digital and physical, collections. However, there was no central repository framework for the preservation of the digital resources.

For both organisations, it made sense to develop a new repository solution that would fulfil the requirements of a CLARIN B-type centre (including the Data Seal of Approval requirements) and that would be less costly to maintain than the existing solution of TLA. Contrary to 15 years ago, there are now a number of widely used open source repository solutions available that one can use or that one can take as a basis for a more tailor made solution.

2 Selecting an existing solution as a basis

At the beginning of the development trajectory, various repository solutions were looked into in order to see whether they would fulfil four basic criteria that would have to be met in order to consider them to form the basis of what would later be called *FLAT* (Fedora Language Archiving Technology). These four criteria were:

1. the software needs to be open source
2. the software needs to be widely used
3. the software needs to be actively developed

4. the software needs to run on a Linux server

On the basis of these four criteria, four solutions were selected and investigated further to see to what extent they fulfilled further requirements:

1. Fedora Commons[1]
2. DSpace[2]
3. EPrints[3]
4. Greenstone[4]

Next these four repository solutions were evaluated on the following additional criteria:

- Main programming language: the programming language in which the largest part of the software is written
- Nested Collections: whether the software supports hierarchically nested collections of arbitrary depth
- Accommodate CMDI: whether the software can in principle support metadata in the CMDI format, even though additional work might be needed for search indexing or visualisation.
- Support Data Types: support the various data types that will likely need to be stored within the organisations (Audio, Video, Text, Images, Annotations on media and text)
- File format verification: whether the software can verify that ingested data files conform to their file format specification
- Checksums: whether the software stores checksums of ingested materials, for the purpose of (periodic) verification of possible data corruption
- Versioning: whether the software supports versioning of stored resources when a new version of a file is ingested
- Handle PID: whether the software supports the issuing of Handle persistent identifiers at the file level
- OAI-PMH: provide metadata for harvesting via the OAI-PMH protocol
- Access Control: provide extensive access control features to allow access to parts of the resources only to certain users or user groups
- LDAP: support connection to an LDAP user database for authentication
- Shibboleth: support the Shibboleth federated authentication solution
- Facet Search: whether the software provides a faceted search/browse function

	Fedora Commons	DSpace	EPrints	Greenstone
Main progr. language	Java	Java	Perl	Java
Nested collections	Yes	Somehow	No	No
Accommodate CMDI	Yes	Yes	Yes	Yes
Support Data Types	Yes	Yes	Yes	Yes
File format verification	Islandora/Hydra	Plug-in	No	No
Checksums	Yes	Yes	Yes	Yes
Versioning	Yes	Yes	Yes	No
Handle PID	Plug-in	Yes	No	No
OAI-PMH	Yes	Yes	Yes	Yes
Access Control	Yes	Yes	No	Yes
LDAP	Yes	Yes	Yes	No
Shibboleth	Plug-in	Yes	Yes	No
Facet Search	Islandora/Hydra	Yes	Plug-in	Yes

Both Fedora Commons and DSpace provide most of the required functionality, either natively, by means of some additional existing code (a Plug-in or some other addition), or via an additional access

[1] http://fedorarepository.org
[2] http://dspace.org
[3] http://www.eprints.org
[4] http://www.greenstone.org

layer in the case of Fedora Commons (Islandora[5] and Project Hydra[6]). Both EPrints and Greenstone have some limitations that would be difficult to overcome, in particular the lack of support for nested collections of arbitrary depth is a fundamental feature that lies at the core of the software. This is an important feature in particular for TLA, since its collections are very hierarchically organised and by flattening these structures (no pun intended), one would lose some important information about the relations between the objects in a collection.

The main difference between DSpace and Fedora Commons (Payette & Lagoze 1998) is that DSpace is a complete system that comes with a web front-end for the end user (actually, two of them), and that Fedora Commons is rather a system for the back-end of a repository, on top of which one can use a number of existing web front-ends or develop your own. For FLAT, the latter was deemed more suitable, taking the Islandora (Leggott 2009) access layer as the front end. Islandora consists of a number of modules for the widely used Drupal[7] content management system, along with a middleware component for handling the communication between the Fedora Commons repository and the Drupal front end. The modular setup makes it easier to develop specific functionality that is needed for the data and metadata types that are used by the FLAT partners and to keep those developments clearly separated from the Islandora and Fedora codebase.

Several CLARIN centres are using a Fedora Commons repository as well for storing their language resources and CMDI metadata. Most of them however use it mainly as a data store and do not provide an advanced access layer on top of it for discovering and visualising the archived resources. Instead, they either use the central metadata and content search engines provided by CLARIN or use the somewhat outdated Fedora *gsearch* metadata search engine. The HZSK CLARIN centre also use the Fedora/Islandora combination for their repository, however they have made some significant changes to the Islandora codebase itself, and these changes are not publicly available. FLAT has been set up as an open source project[8] on GitHub from the start and will most likely be used by other organisations besides the Meertens Institute and TLA in the coming year.

3 Extending the basis

The initial phase of the FLAT development trajectory was mainly about deploying and configuring Fedora/Islandora such that it could be used as a read-only repository viewer for CMDI collections. Ingesting materials was done by generating Fedora objects in FOXML format on disk and using batch ingest scripts that come with Islandora. The largest part of the work in the initial phase went into developing the conversion scripts, developing automated deployment scripts for the whole technology stack using the Docker[9] framework, and into developing a Drupal module that could be used to visualise CMDI metadata within the Islandora layer. The conversion scripts for the FOXML creation were inspired by code that was provided by the IDS CLARIN centre. After this initial phase, the focus was on indexing the metadata into a Solr[10] database for faceted searching, on looking into user authentication and access control, on installing and configuring an OAI-PMH server and on creating suitable graphical styles for both institutes. Figure 1 shows a screen shot of the faceted search interface of a FLAT prototype for TLA.

The largest part of the development in terms of newly written code started after that point when the main focus was on developing a new ingest front- and backend, suitable for ingesting CMDI-based collections from either command line scripts, exploration tools as developed by the Meertens Institute or from an easy to use graphical interface that is suited for self-archiving by end users. While Islandora does offer the option to ingest materials into a Fedora repository by means of a graphical web interface, this does require a certain understanding of how Fedora organises its objects and of the terminol-

[5] http://islandora.ca
[6] https://projecthydra.org
[7] https://www.drupal.org
[8] https://github.com/TheLanguageArchive/FLAT
[9] https://www.docker.com
[10] http://lucene.apache.org/solr/

ogy that Fedora uses. This makes the ingest procedure unnecessarily complex and time consuming. In FLAT a submission basically consists of a CMDI record and the resources it describes. When the repository receives a submission, it runs several checks to confirm that the submission is valid and complete before storing the CMDI record and the resources in the Fedora repository. These checks can be adapted and extended to match the business rules of the organisation.

The development of an easy to use graphical interface for self-archiving by end users poses a few challenges, mainly due to the large variation in complexity of the data sets that are to be archived. These vary from large multimedia language corpora with elaborate metadata descriptions and deeply nested hierarchies to a single zip file described by a single metadata record. The deposit workflows should ideally cater for data sets of all complexity, while still making the deposit of the simplest data sets not unnecessarily difficult. A first prototype of such a deposit solution is ready and a production-ready version is targeted for the fall of this year. It (optionally) uses the OwnCloud[11] self-hosted cloud solution for handling uploads of large amounts of data (large numbers of files and/or large files), which is both reliable as well as convenient for the user, since it closely resembles commercial cloud solutions that they are familiar with, such as Dropbox. CMDI metadata editing is possible in the deposit tool itself via web-based forms, albeit initially only for a limited number relatively simple CMDI profiles. Editing of more complex profiles will only be worked on after the first production-ready release.

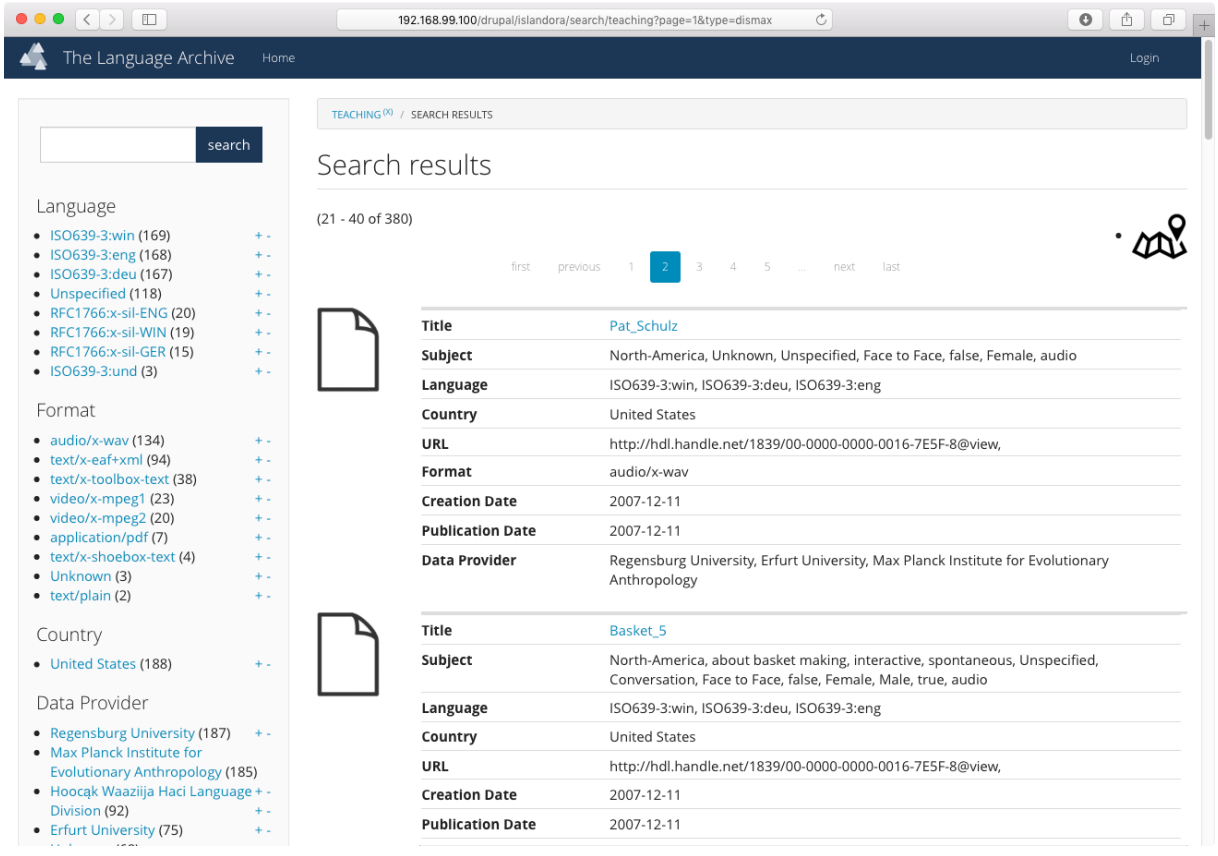

Figure 1. Faceted search interface in a FLAT prototype for The Language Archive

4 Future developments

The current FLAT developments are done on the basis of the latest stable Islandora release, which is built for the legacy Fedora Commons 3. Both Fedora Commons 4, which was released in December 2014, as well as the next Islandora generation that works together with it (codename *CLAW*[12]) are

[11] https://owncloud.org
[12] https://github.com/Islandora-CLAW/CLAW

complete rewrites from the ground up, introducing major changes to the underlying technology. While we expect to be using the legacy Fedora Commons 3 for at least two years until the development of Islandora *CLAW* has advanced far enough, migration will be required at some point. Migration tools will be provided by the community, but any additional modules and code will need to be rewritten to some extent. Major changes like these are not unique to Fedora/Islandora though and do not occur that frequently. After the migration, FLAT should be relatively easy to maintain for many years to come.

5 Conclusions

The combination of the Fedora Commons repository solution with the Islandora discovery layer provides a solid yet flexible basis for developing a CLARIN-compatible repository solution (Windhouwer et al 2016) for archiving language-related data sets with CMDI metadata. The modular setup and the possibility to override certain functionality in Drupal makes it relatively easy to separate any additional code from the core open source solutions as provided by the communities, which helps in reducing maintenance costs.

With FLAT, we are combining the existing functionality of the Fedora/Islandora combination and supplement this with a number of additional modules as well as custom developed ingest tools to provide a solution that is suitable for our various use cases, including self-archiving by end users.

The relatively large technological changes between versions 3 and 4 of Fedora Commons and the associated Islandora will require a fair amount of development work to be done on FLAT once the new combination is ready for production use, however we do not expect a similar effort to be needed for many years after that.

Abbreviations and glossary of terms

CLARIN: Common Language Resources and Technology Infrastructure

CMDI: Component MetaData Infrastructure

Data Seal of Approval (DSA): assessment method for trusted digital repositories.

FOXML: Fedora Object XML, format for specifying properties of objects in a Fedora Commons repository

LDAP: Lightweight Directory Access Protocol

OAI-PMH: Open Archives Initiative - Protocol for Metadata Harvesting

PID: Persistent IDentifier, such as Digital Object Identifier (DOI) or Handle.

References

Leggott, Mark A., “Islandora: a Drupal/Fedora Repository System”, 4th International Conference on Open Repositories, Atlanta, Georgia, 2009

Payette, S., and Lagoze, C., “Flexible and Extensible Digital Object and Repository Architecture (FEDORA)”, European Conference on Research and Advanced Technology for Digital Libraries, Lecture Notes in Computer Science, Springer, 1998

Windhouwer, M., Kemps-Snijders, M., Trilsbeek, P., Moreira, A., Van der Veen, B., Silva, G., and Von Rhein, D., “FLAT: constructing a CLARIN compatible home for language resources”. 10th International Conference on Language Resources and Evaluation, Portorož, Slovenia, 2016.