FLAT: Constructing a CLARIN Compatible Home for Language Resources

Menzo Windhouwer,[a] Marc Kemps-Snijders,[a] Paul Trilsbeek,[b] André Moreira,[b] Bas van der Veen,[a] Guilherme Silva,[b] Daniel von Reihn[b]

[a]Meertens Institute
Amsterdam, The Netherlands
[b]The Language Archive – Max Planck Institute for Psycholinguistics
Nijmegen, The Netherlands
E-mail: {Menzo.Windhouwer, Marc.Kemps.Snijders, Bas.van.der.Veen}@meertens.knaw.nl,
{Paul.Trilsbeek, Andre.Moreira, Guilherme.Silva, Daniel.vonRhein}@mpi.nl

Abstract

Language resources are valuable assets, both for institutions and researchers. To safeguard these resources requirements for repository systems and data management have been specified by various branch organizations, *e.g.*, CLARIN and the Data Seal of Approval. This paper describes these and some additional ones posed by the authors' home institutions. And it shows how they are met by FLAT, to provide a new home for language resources. The basis of FLAT is formed by the Fedora Commons repository system. This repository system can meet many of the requirements out-of-the box, but still additional configuration and some development work is needed to meet the remaining ones, *e.g.*, to add support for Handles and Component Metadata. This paper describes design decisions taken in the construction of FLAT's system architecture via a mix-and-match strategy, with a preference for the reuse of existing solutions. FLAT is developed and used by the a Institute and The Language Archive, but is also freely available for anyone in need of a CLARIN-compliant repository for their language resources.

Keywords: CLARIN, repository system, component metadata

1. Introduction

A repository, which provides access to language resources, is not only a valuable asset for its home organization but also for the whole community of humanities scholars. Often these resources can be considered treasures of cultural heritage. This is showcased by the recent recognition of the UNESCO Memory of the World Register for collections in The Language Archive (The Language Archive, 2015).

There are many ways to setup a repository ranging from a homegrown solution to open source or commercial systems used by many organizations. The Language Archive (MPI for Psycholinguistics, 2016) and the Meertens Institute (Meertens Institute, 2016a) have both a long history in digital archiving, which started when general purpose repository systems were not yet available. Departing from such homegrown solutions, these institutes recently started the construction of FLAT (Fedora Language Archiving Technology); [1] a new repository system based on open source components with custom extensions for specific purposes.

2. Requirement Analysis

To safeguard these valuable language resources and to serve the interested researchers well, requirements for repository systems and data management have been specified by various branch organizations. In the construction of FLAT these requirements, mainly influenced by the CLARIN e-infrastructure (CLARIN ERIC, 2016b), which also implies passing the certification procedures of the Data Seal of Approval (DSA, 2016), are taken into account next to the needs of the home institutions. Many of these requirements deal with organizational issues, but also a lot of them result in technical requirements to be supported by the repository used. This section focuses mainly on the latter.

2.1 CLARIN Requirements

CLARIN recognizes various center types, where the most natural center type for an organization with a repository of language resources is the B type, *i.e.*, a service-providing centre. A B centre needs to fulfill a wide range of requirements (CLARIN ERIC, 2012). These are the technical ones that impact the design of FLAT:[2]

[CLARIN-B-2] *Centres need to adhere to the security guidelines, i.e. the servers need to have accepted certificates*: FLAT's web services should have no problem to be accessed via HTTPS.

[CLARIN-B-3] *Centres need to join the national identity federation where available and join the CLARIN service provider federation to support single identity and single sign-on operation based on SAML 2.0 and trust declarations*: FLAT should be able to run on a server in the CLARIN service provider federation and should support single-sign-on based on SAML 2.0.

[CLARIN-B-5] *Centres need to offer component based metadata (CMDI) that make use of elements from accepted registries such as ISOcat in accordance with the CLARIN agreements, i.e. metadata needs to be harvestable via OAI-PMH*: FLAT should support CMDI (CLARIN ERIC, 2016c) as a metadata format and provides the metadata via OAI-PMH.

[1] Actually FLAT is not limited to language resources, so Fedora Long-term Archiving Technology is also a valid interpretation of the acronym. The meaning of other abbreviations used in this paper can be found in Appendix A.

[2] See Appendix B for the full list of requirements for CLARIN B centres.

[CLARIN-B-6] *Centres need to associate PIDs records according to the CLARIN agreements with their objects and add them to the metadata record*: FLAT should assign PIDs to every resource and metadata record in the repository.

2.2 DSA Requirements

The CLARIN B centre requirements also require the participation in a DSA or MOIMS-RAC quality assessment. For now FLAT focuses on the DSA requirements (DSA, 2014). These are the ones that impact the design of FLAT:[3]

[DSA-10] *The data repository enables the users to discover and use the data and refer to them in a persistent way*: FLAT should provide ways to search for resources, to navigate the collection hierarchy and to refer to the discovered resources using handles.

[DSA-11] *The data repository ensures the integrity of the digital objects and the metadata*: FLAT should maintain and produce checksums for digital objects and CMDI metadata.

[DSA-12] *The data repository ensures the authenticity of the digital objects and the metadata*: FLAT should version both digital objects and metadata, which means that it is always possible to retrieve originally deposited object.

[DSA-13] *The technical infrastructure explicitly supports the tasks and functions described in internationally accepted archival standards like OAIS*: FLAT will carry on the tradition of Live Archives (DAM-LR, 2007), which extends OAIS (CCSDS Secretariat, 2012).

2.3 Other Requirements

As both The Language Archive and the Meertens Institute have already vast digital collections and a long history of technological development they pose additional technical requirements to FLAT:

[Home-1] FLAT should support arbitrary deep collection hierarchies.

[Home-2] FLAT should support Handles as PIDs.

[Home-3] FLAT should work with arbitrary CMDI profiles.

[Home-4] FLAT should provide resource level access control facilities, allowing access rules to be independently specified for each resource regarding both specific user and/or user groups.

[Home-5] FLAT should allow collection management to review submissions before resources are actually ingested in the repository.

[Home-6] FLAT should allow system management to determine the location of resources on persistent storage, *e.g.*, from fast access times to secure tape drives.

[Home-7] FLAT should allow storing arbitrary relationships between resources, *e.g.*, between a physical fiche and the scan of it.

[Home-8] FLAT should provide entry points for interaction with the current research infrastructure, *i.e.*, the Virtual Research Environments (VREs) used by the researchers.

All these requirements are taken into account in the overall system architecture of FLAT and the selection of readily available components.

2.4 Related Work

CLARIN's center registry (CLARIN ERIC, 2016a) shows that many centers have based their repository on Fedora Commons (DuraSpace, 2016a) or DSpace (DuraSpace, 2016b). The LINDAT group at Charles University in Prague has made their DSpace-based solution generic (UFAL, 2016). A growing number of CLARIN centers use it as their repository system. Unfortunately this solution does not meet all the specific institutional requirements, *e.g.*, support for arbitrary deep collection hierarchies or arbitrary CMDI profiles.

3. System Architecture

The base implementation of FLAT (The Language Archive, 2016) is delivered by the Fedora Commons repository system (DuraSpace, 2016a), which meets some requirements out of the box, *e.g.*, ensuring integrity and authenticity of digital objects [DSA-11, DSA-12]. Other requirements can be relatively easily supported by common extensions, *e.g.*, OAI-PMH and search facilities [CLARIN-B-5, DSA-10]. This combined with the power of the Drupal-based Islandora (Islandora Community, 2016) for the online user interface provides a good starting point for FLAT. However, to meet the very CLARIN specific requirements (*i.e.*, support for CMDI and persistent identifiers) additional development work needs to done. The next sections will discuss the configuration of Fedora Commons, Islandora and FLAT's specific system components. An overview is depicted in Figure 1.

3.1 Fedora Commons

Fedora Commons has a very flexible digital object model that can be configured to meet specific needs. For FLAT two content models, taken from Islandora, provide the base models: the compound and collection content models. Each CMDI record corresponds to one compound [CLARIN-B-5]. Members of the compound consist of objects for all the resources described by the record. Compounds can be parts of collections, which themselves can be part of collections again, thereby allowing for nested collection hierarchies of arbitrary depth [Home-1]. Every object, be it a CMDI record or a resource, has a PID in the form of a Handle. Such a handle resolves to the Fedora Commons API call to retrieve the object's main data stream [CLARIN-B-6, DSA-10, Home-2].

Fedora Commons uses RDF to state relationships, *e.g.*, between a compound or collection and its members. This mechanism can also be used for other repository specific relationships [Home-7].

[3] See Appendix C for the full set of guidelines from the Data Seal of Approval.

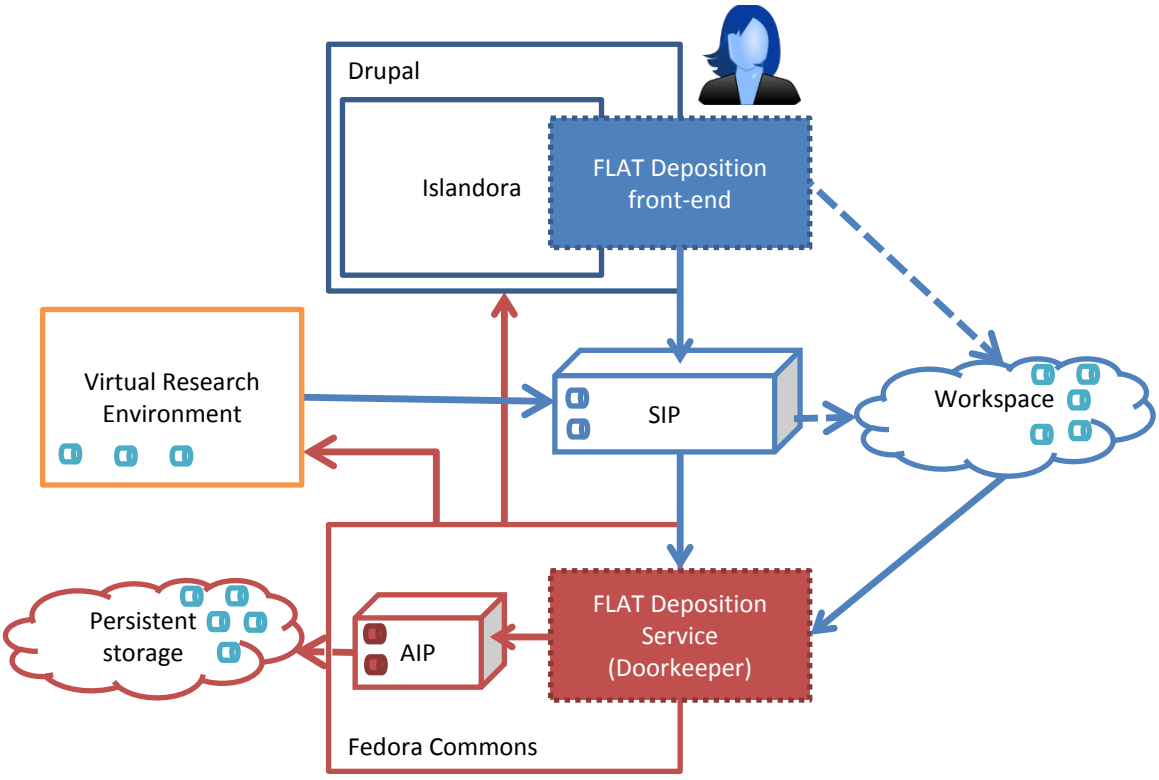

Figure 1: FLAT overview

FLAT stores resource-related data streams outside Fedora Commons in so-called external data streams. This makes it possible to differentiate between different resource usage scenarios by employing different storage facilities for different types of resources [Home-6], *e.g.*, videos can be available on faster media so they can be streamed out quickly.

OAI-PMH is supported by configuring a well-known extension for Fedora Commons based on Proai (Cornell University and Fedora Commons, Inc., 2009), which is configured to provide CMDI records upon request [CLARIN-B-5].

To support metadata search, two different indexing strategies are being used [DSA-10]:

1. The Language Archive uses the generic indexing service of Fedora Commons, where available facets are dynamically determined through analysis of CMDI profiles and records in the repository [Home-3].
2. The Meertens Institute uses a separate indexing service it developed for CLARIN-NL (CLARIN-NL, 2016) and Nederlab (Nederlab, 2016), which can handle arbitrary CMDI profiles and records [Home-3]. This makes the full metadata information available in the index. The resulting index is also used by VREs that are designed to support other use cases requiring other information fields [Home-8].

Access control is implemented in Fedora Commons using XACML policies (OASIS, 2016). This policy language is very flexible and allows for the specification of fine-grained access policies regarding individual users and/or groups for each resource [HOME-4]. The access rules already used by the institutes are translated into XACML policies.

3.2 Islandora

FLAT currently uses Islandora to provide users with both search, based on the search indexes described above, and browse facilities. However, this dependency is less severe than the dependency on Fedora Commons, *i.e.*, it is possible to use another user interface without losing the main facilities of FLAT, *e.g.*, support for CMDI and Handles. Islandora can be extended with so called solution packs. For FLAT, a CMDI solution pack has been developed, which can be enabled in Islandora to render the CMDI metadata records [CLARIN-B-5, Home-3].

Drupal (Drupal.org, 2016) supports a Shibboleth-based single-sign-on extension and both Fedora Commons and Islandora have proven to run on a server configured for the CLARIN Service Provider Federation without any problem [CLARIN-B-2, CLARIN-B-3].

3.3 Deposition

Before ingesting new resources into FLAT, a series of validation steps are necessary, *e.g.*, to check validity of the CMDI record and integrity of resources. Some of this functionality could be provided out of the box by Islandora, *e.g.*, file type checking using the FITS (OpenScholar, 2016) solution packs. However, Islandora ingests new objects directly into the Fedora Commons repository, while in FLAT a temporary workspace is required where further ingest is halted until collection managers have reviewed the contents. Also the archives have their own specific needs, *e.g.*, to trigger additional content-based index actions to support access from VREs [Home-8]. To enable this, a flexible deposit tool and service, known as the FLAT Doorkeeper, is constructed, which accepts a submission (also known as a Submission

Figure 2: The FLAT instance at the Meertens Institute (2016b)

Information Package (SIP) in OAIS), which is uploaded via a SWORD API, and pushes it through a sequence of steps. These steps are mainly wrappers around general purpose components, *e.g.*, FITS for file type checking and Xerces2 (The Apache Software Foundation, 2012) for XML validation. In addition, the deposition steps include manual validation [Home-5], triggering local index services and the actual deposition in Fedora Commons (as an Archival Information Package (AIP) in OAIS terminology) [DSA-13].

This Doorkeeper is the main entrance to FLAT and is the safeguard to maintain a consistent repository. Two different approaches for interaction with the FLAT Doorkeeper are being pursued:

1. The Language Archive develops an online user interface, which can be used by researchers to deposit their language resources accompanied by their CMDI records.
2. At the Meertens Institute a more tight integration with data production processes is pursued, *e.g.*, VREs such as Nederlab (Nederlab, 2016) or current survey/crowdsourcing environments [Home-8]. Here researchers can use dedicated expert interfaces to manipulate and analyze specific kinds of data, and the results of these analyses can be ingested into the FLAT repository using the deposition service.

3.4 System Integration

Due to the number of software components used by this architecture as well as its high degree of flexibility, the deployment and configuration of this system can become fairly complex. To overcome this and boost deployment simplicity, FLAT is currently offered as a freely available Docker machine image (Docker, 2016; The Language Archive, 2016), allowing it to be easily installed without extensive knowledge of underlying platforms like Fedora.

4. Conclusions and Future Work

Using the FLAT system architecture described above, both the Meertens Institute and The Language Archive can meet their own and the researchers' requirements, as presented by CLARIN, to provide a safe home for the valuable language resources entrusted to them. Read-only instances of FLAT have been successfully deployed and

have been able to provide access to more than half a million resources described by just over 110,000 CMDI metadata records in The Language Archive, and a quarter of a million resources described by as many CMDI records at the Meertens Institute (see Figure 2). The deposition tool and service are currently under construction to be followed later by an online user interface. FLAT will thus become another option for a repository system that supports the CLARIN requirements (and more) out-of-the-box.

Appendix A: Abbreviations

AIP	Archival Information Package
API	Application Programming Interface
CLARIN	Common Language Resources and technology INfrastructure
CMDI	Component MetaData Infrastructure
DAM-LR	Distributed Access Management for Language Resources
DSA	Data Seal of Approval
ERIC	European Research Infrastructure Consortium
FITS	File Information Tool Set
FLAT	Fedora Language Archiving Technology
HTTPS	HyperText Transfer Potocol Secure
MOIMS	Mission Operations & Information Management System
OAI	Open Archives Initiative
OAIS	Open Archival Information System
OASIS	Organization for the Advancement of Structured Information Standards
PID	Persistent IDentifier
PMH	Protocol for Metadata Harvesting
RAC	Repository Audit and Certification
RDF	Resource Description Framework
SAML	Security Assertion Markup Language
SIP	Submission Information Package
TLA	The Language Archive
UNESCO	United Nations Educational, Scientific and Cultural Organization
VRE	Virtual Research Environment
XACML	eXtensible Access Control Markup Language
XML	eXtensible Markup Language

Appendix B: Requirements for CLARIN Centres

Source: (CLARIN ERIC, 2012)

[CLARIN-B-1] Centres need to offer useful services to the CLARIN community and to agree with the basic CLARIN principles (own architecture choice, explicit statement about quality of service, usage of persistent identifiers, adherence to agreed formats, protocols and APIs).

[CLARIN-B-2] Centres need to adhere to the security guidelines, i.e. the servers need to have accepted certificates.

[CLARIN-B-3] Centres need to join the national identity federation where available and join the CLARIN service provider federation to support single identity and single sign-on operation based on SAML2.0 and trust declarations. In case all resources at a centre are open, setting up a Service Provider is optional.

[CLARIN-B-4] Centres need to have a proper and clearly specified repository system and participate in a quality assessment procedure as proposed by the Data Seal of Approval or MOIMS-RAC approaches.

[CLARIN-B-5] Centres need to offer component based metadata (CMDI) that make use of elements from accepted registries such as ISOcat in accordance with the CLARIN agreements, i.e. metadata needs to be harvestable via OAI PMH.

[CLARIN-B-6] Centres need to associate PIDs records according to the CLARIN agreements with their objects and add them to the metadata record.

[CLARIN-B-7] Each centre needs to make clear statements about their policy of offering data and services and their treatment of IPR issues.

[CLARIN-B-8] Each centre needs to make explicit statements to the CLARIN boards about its technological and funding support state and its perspectives in these respects.

[CLARIN-B-9] Centres need to employ activities to relate their role in CLARIN to the research community in order to guarantee a research based status of the infrastructure and allow researchers to embed their services in their daily research work.

[CLARIN-B-10] Centres that are offering infrastructure type of services (A or E) need to specify their services for CLARIN and the terms of giving service.

[CLARIN-B-11] Centres are advised to participate in the Federated Content Search with their collections by providing an SRU/CQL Endpoint. This content search is especially suitable for textual transcriptions and resources.

Appendix C: The DSA Guidelines 2014-2015

Source: (DSA, 2014)

Guidelines Relating to Data Producers:

[DSA-1] The data producer deposits the research data in a data repository with sufficient information for others to assess the scientific and scholarly quality of the research data and compliance with disciplinary and ethical norms.

[DSA-2] The data producer provides the research data in formats recommended by the data repository.

[DSA-3] The data producer provides the research data together with the metadata requested by the data repository.

Guidelines Related to Repositories:

[DSA-4] The data repository has an explicit mission in the area of digital archiving and promulgates it.

[DSA-5] The data repository uses due diligence to ensure compliance with legal regulations and contracts.

[DSA-6] The data repository applies documented

processes and procedures for managing data storage.

[DSA-7] The data repository has a plan for long-term preservation of its digital assets.

[DSA-8] Archiving takes place according to explicit workflows across the data life cycle.

[DSA-9] The data repository assumes responsibility from the data producers for access to and availability of the digital objects.

[DSA-10] The data repository enables the users to utilize the research data and refer to them.

[DSA-11] The data repository ensures the integrity of the digital objects and the metadata.

[DSA-12] The data repository ensures the authenticity of the digital objects and the metadata.

[DSA-13] The technical infrastructure explicitly supports the tasks and functions described in internationally accepted archival standards like OAIS.

Guidelines Related to Data Consumers:

[DSA-14] The data consumer must comply with access regulations set by the data repository.

[DSA-15] The data consumer conforms to and agrees with any codes of conduct that are generally accepted in higher education and research for the exchange and proper use of knowledge and information.

[DSA-16] The data consumer respects the applicable licences of the data repository regarding the use of the research data.

Bibliographical References

CCSDS Secretariat (2012). *Reference Model for an Open Archival Information System (OAIS).* Washington: Consultative Committee for Space Data Systems (CCSDS) http://public.ccsds.org/publications/archive/650x0m2.pdf (accessed 9 March 2016).

CLARIN ERIC (2012). *Centre requirements* (revised version) http://hdl.handle.net/1839/00-DOCS.CLARIN.EU-77 (accessed 1 March 2016).

CLARIN ERIC (2016a). *CLARIN Centre Registry* https://centres.clarin.eu/ (accessed 1 March 2016).

CLARIN ERIC (2016b). *CLARIN Infrastructure* http://clarin.eu/(accessed 24 February 2016).

CLARIN ERIC (2016c). *Component Metadata* http://www.clarin.eu/content/component-metadata (accessed 24 February 2016).

CLARIN-NL (2016). *CLARIN-NL* http://clarin.nl/ (accessed 1 March 2016).

Cornell University and Fedora Commons, Inc. (2009). *Proai Documentation* http://proai.sourceforge.net/ (accessed 1 March 2016).

DAM-LR (2007). *Live Archives Initiative* http://www.mpi.nl/dam-lr/lra-flyer/ (accessed 1 March 2016).

Docker (2016). *Docker - Build, Ship, and Run Any App, Anywhere* https://www.docker.com/ (accessed 10 March 2016).

Drupal.org (2016). *Drupal - Open Source CMS* https://www.drupal.org/ (accessed 9 March 2016).

DSA (2014). *The Guidelines 2014-2105* http://datasealofapproval.org/en/information/guidelines/ (accessed 1 March 2016).

DSA (2016). *Home | Data Seal of Approval* http://datasealofapproval.org/en/ (accessed 1 March 2016).

DuraSpace (2016a). *Fedora Repository* http://fedora-commons.org/ (accessed 1 March 2016).

DuraSpace (2016b). *DSpace* http://dspace.org/ (accessed 1 March 2016).

Islandora Community (2016). *Islandora Website* http://islandora.ca/ (accessed 1 March 2016).

Meertens Institute (2016a). *Welcome to the Meertens Institute* http://www.meertens.knaw.nl/cms/en/ (accessed 1 March 2016).

Meertens Institute (2016b). *Het Archief* http://archief.meertens.knaw.nl/ (accessed 14 March 2016).

MPI for Psycholinguistics (2016). *The Language Archive* https://tla.mpi.nl/ (accessed 1 March 2016).

Nederlab (2016). *Nederlab* http://www.nederlab.nl/ (accessed 1 March 2016).

OASIS (2016). *OASIS eXtensible Access Control Markup Language (XACML) TC* https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml (accessed 9 March 2016).

OpenScholar (2016). *File Information Tool Set (FITS)* http://projects.iq.harvard.edu/fits (accessed 1 March 2016).

The Apache Software Foundation (2012). *The Apache Xerces™* http://xerces.apache.org/ (accessed 1 March 2016).

The Language Archive (2015). *UNESCO Memory of the World Register to recognize collections in The Language Archive - The Language Archive* https://tla.mpi.nl/tla-news/unesco-memory-of-the-world-register-to-recognize-collections-in-the-language-archive/ (accessed 1 March 2016).

The Language Archive (2016). *FLAT* https://github.com/TheLanguageArchive/FLAT (accessed 24 February 2016).

UFAL (2016). *LINDAT/CLARIN Digital Repository Based on DSpace* https://github.com/ufal/lindat-dspace (accessed 24 February 2016).