



LEAF - Linking and Exploring Authority Files

IST-2000-26323

Interface Control Document

| | |
|--------------------|------------------------------|
| DATE | 23.04.2003 |
| AUTHOR | K. Majcen, H. Vallant |
| QUALITY ASSURANCE | R. Bull |
| WORK PACKAGE | 6 |
| KEYWORDS | interfaces, FTP, OAI, Z39.50 |
| DELIVERABLE NUMBER | 6.3 |

DOCUMENT HISTORY

| Version | Date | Comments | Status | Distribution |
|---------|-----------|--|--------------|-----------------|
| 0.1 | 5.12.2001 | First internal draft of the structure | Draft | JR's LEAF team |
| 0.2 | 29.7.2002 | Update of interface section | Draft | JR's LEAF team |
| 0.3 | 3.2.2003 | Included information for comments on ADD | Draft | LEAF consortium |
| 1.0 | 7.3.2003 | Added information from technical meeting in Berlin (February 7 th 2003) | Draft for QA | LEAF consortium |
| 1.1 | 9.4.2003 | Minor corrections | Final for QA | Rob Bull |
| 1.2 | 23.4.2003 | QA comments | Final | LEAF consortium |

Table of Contents

| | |
|--|----------|
| Table of Contents | 2 |
| 1 Introduction | 3 |
| 1.1 Scope..... | 3 |
| 1.2 Document Overview | 3 |
| 2 System context | 3 |
| 3 Interfaces | 4 |
| 3.1 Harvesting local authority files | 4 |
| 3.1.1 Prerequisites..... | 4 |
| 3.1.2 Harvesting..... | 5 |
| 3.1.3 Updates by searching | 5 |
| 3.2 Connecting the LEAF system and the LEAF Conversion utility..... | 5 |
| 3.2.1 Conversion from local record formats into EAC format | 5 |
| 3.2.2 Conversion from EAC format into the export formats defined for LEAF..... | 5 |
| 3.3 Connecting from external systems..... | 5 |
| 3.3.1 Get results for a search with given names | 5 |
| 3.3.2 Get detailed information for a particular result record | 5 |
| 3.3.3 Use case MALVINE..... | 5 |
| 3.4 Connecting to external systems..... | 5 |
| 3.5 Maintaining the connections to external resources | 5 |
| 3.5.1 New configuration available..... | 5 |
| 3.5.2 Server status list | 5 |
| 3.5.3 Configuration download interface | 5 |
| 3.5.4 Server status list | 5 |
| Figures | 5 |
| Tables | 5 |
| Definitions, Acronyms and Abbreviations | 5 |
| References | 5 |

1 Introduction

1.1 Scope

This document is the Interface Control Document (ICD) prepared for the Project LEAF (Linking and Exploring Authority Files; project number IST-2000-26323). This document was done under Work Package 6 (*Functional Specifications of the Demonstrator*) of the LEAF project (see [DoW]).

The purpose of this document is to ensure that the integration of the LEAF components developed by different partners of the LEAF consortium can be performed without too many problems.

The intended **readership** of this document are all **project participants** who are involved in the software development of LEAF components.

1.2 Document Overview

Chapter 1 is this introduction. The system overview and the various interfaces which exist between components but also external systems are shown in chapter 2. The details of the interfaces are described in chapter 3.

2 System context

The overall architecture of the LEAF system is shown in this chapter. Several components can be extracted from that. Between those components interfaces exist. Furthermore external systems like MALVINE will make use of the biographic search service provided by the LEAF system and the data found through that. An interface is provided for that purpose.

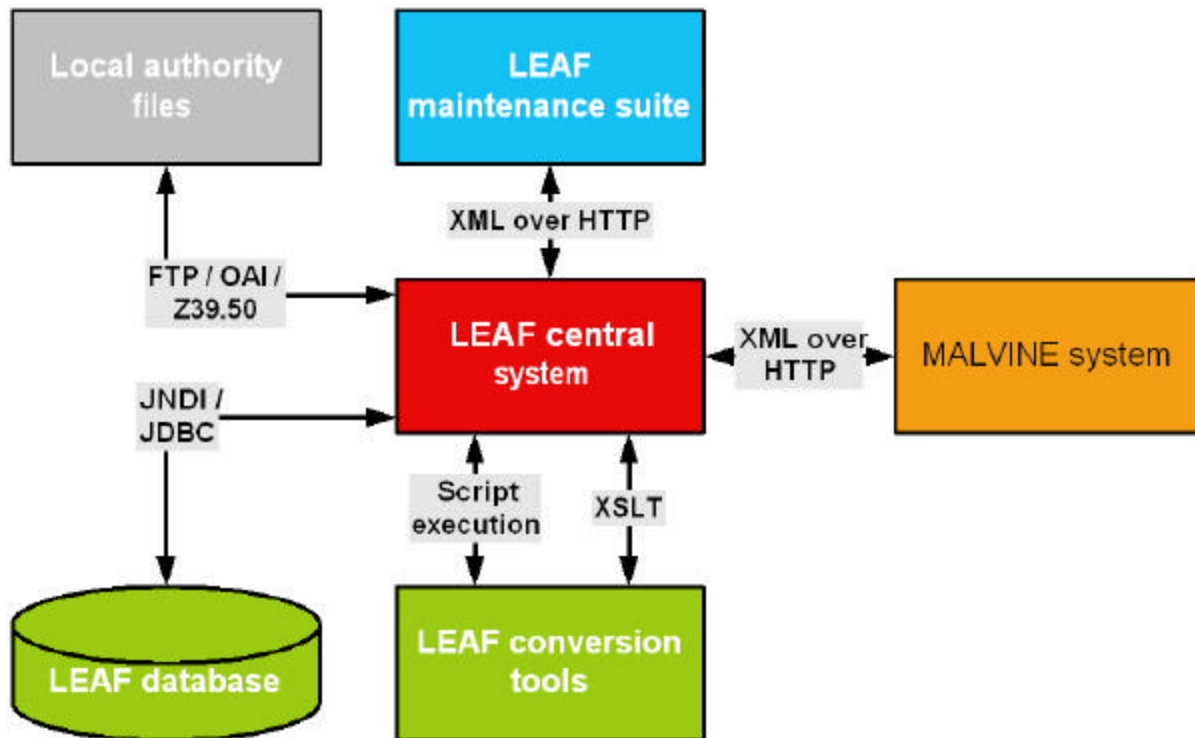


Figure 1: The interfaces from / to the LEAF central system

3 Interfaces

This chapter describes in detail the interfaces which exist for the LEAF central system. The following table lists the internal and external interfaces of the LEAF central system.

| Interface | Description / comment |
|------------------------|--|
| LEAF database | Internal database interface → not described in this document |
| LEAF conversion tools | Used for the import and export of records in various formats |
| MALVINE system | Allows to use some LEAF search functions from MALVINE |
| Local authority files | Used for harvesting of records |
| LEAF maintenance suite | Administrative details of LEAF systems |

Table 1: Internal and external interfaces

3.1 Harvesting local authority files

Harvesting can mean either of two things in the LEAF context:

- harvesting for purpose of insert / update / delete data
- updates by searching (insert at user query time; maintenance of included records will be done on a regular time basis)

3.1.1 Prerequisites

| Protocol | Archive side | LEAF server side |
|--------------|--|--|
| FTP-Download | <ul style="list-style-type: none"> ▪ FTP server ▪ facility for compressing (gzip, winzip) ▪ all records or updated/new records allowed ▪ any format the conversion utility supports | <ul style="list-style-type: none"> ▪ FTP client ▪ facility for uncompressing (gunzip, winzip) ▪ Conversion utility must support incoming format |
| FTP -Upload | <ul style="list-style-type: none"> ▪ FTP client ▪ facility for compressing (gzip, winzip) ▪ all records or updated/new records allowed ▪ any format the conversion utility supports | <ul style="list-style-type: none"> ▪ FTP server ▪ facility for uncompressing (gunzip, winzip) ▪ Conversion utility must support incoming format |
| OAI | <ul style="list-style-type: none"> ▪ OAI server ▪ Data must be in an XML format which the conversion utilities support and compliant to the OAI rules ▪ Updated records allowed ▪ Information about deleted records as defined in the OAI message or use of the user interface for manual removal of records | <ul style="list-style-type: none"> ▪ OAI client ▪ Conversion utility must support incoming format |
| Z39.50 | <ul style="list-style-type: none"> ▪ Z39.50 server (search & retrieval) ▪ any format the conversion utility supports and compliant with Z39.50 definitions ▪ The record converted to EAC must contain an 'ID' element or attribute. ▪ The Z39.50 server must provide an attribute for the search for the ID. | <ul style="list-style-type: none"> ▪ Z39.50 client (search & retrieval) ▪ Conversion utility must support incoming format |

Table 2: Prerequisites on archive and LEAF server side

3.1.2 Harvesting

The harvesting can be done by protocols like FTP or OAI (protocol defined at [OAI]). Of course other protocols could be thought about as well but only the mentioned ones are described here. During the implementation of the offline system mechanisms for the protocols will be used at the server in coordination with the local sites. To include recent data these mechanisms have to be triggered in a defined way or repositories have to be monitored by some listening procedure.

For an FTP upload the archive only has to use an **FTP client** (used manually or triggered by some update mechanism) connecting to **port 21** of the **LEAF server**. The archive's FTP client connects to the LEAF server and transfers the new file. The archive is assigned a **user name**, a **password** and a **file name** for uploads during integration into the LEAF framework. Furthermore the archive has to tell if it provides **full exports** or **updates**.

For FTP download the archive has to provide an **FTP server** where the new files are stored. The **LEAF server** connects to a predefined **port** (the default is 21) of the archive's server. Then the LEAF server downloads the new file. The archive has to provide a **server name**, a **user name**, a **password** and a **file name** for downloads during integration into the LEAF framework. Furthermore the archive has to tell if it provides **full exports** or **updates**.

For the use of OAI the archive has to provide an **OAI server** from where the new records can be harvested. The **LEAF server** connects to a predefined **port** of the archive's server. Then the LEAF server downloads the new file. The archive has to provide **server name & port** during integration into the LEAF framework. Furthermore the archive has to provide **updates/deletes** but not **full exports**.

3.1.3 Updates by searching

For integration of resources which can only provide the records via Z39.50 or which do not allow to harvest data as a whole a Z39.50 server may be used. This means that a search on the LEAF system is also targeted against a Z39.50 talking OPAC server which can provide biographic information. The biographic information requested in detail by the user will be added to the system and will be linked afterwards. The records will be searched for in regular time intervals to have current data available.

For the use of Z39.50 the archive has to provide a **Z39.50 server** where searches can be targeted to. The **LEAF server** connects to a predefined **port** (the default is 210) of the archive's server. Then the LEAF server performs search and retrieval operations on that server. This is also done when updating records. The archive has to provide a **server name & port** and a **database name** during integration into the LEAF framework.

Services which are used:

- Init
- Search
- Present
- Close

Used attribute set:

- BIB-1 (1.2.840.10003.3.1)

Used record syntaxes are:

- Marc21 – formerly USmarc <http://www.loc.gov/marc/> (1.2.840.10003.5.10)
- MAB2-PND/GKD <http://www.ddb.de/cgi-bin/bermudix.pl?url=professionell/mab.htm> (1.2.840.10003.5.16)
- Unimarc <http://www.ifla.org/VI/3/p1996-1/sec-uni.htm> (1.2.840.10003.5.1)
- SUTRS <http://cweb.loc.gov/z3950/agency/asn1.html#RecordSyntax-SUTRS> (1.2.840.10003.5.101) including correct EAC according to the LEAF rules (as used in the conversion utilities)

A general prerequisite for the above formats is the support by the conversion utilities (python scripts and XSL style sheets).

3.2 Connecting the LEAF system and the LEAF Conversion utility

Two kinds of functionality have to be supported by the conversion utilities ([ConvTool]) to allow integration into the LEAF system:

- Conversion from local record formats into LEAF EAC format
- Conversion from EAC format into the local format of the record or into the export formats defined for LEAF

Any limitations given for the latter conversions will be documented in the results of the work package 4 "Data Representation Study".

3.2.1 Conversion from local record formats into EAC format

The conversion from local record formats into EAC format (described in [RND]) is used in 2 different places / functions in the LEAF system:

- before inserting / updating the harvested records (part of offline functionality)
- before inserting / updating records acquired through searches in remote systems (part of online functionality)

These conversion functions are provided by several Python scripts (www.python.org). The integration of these functions into the LEAF system can therefore be done in either of 2 ways: connecting the conversion utility directly to the LEAF system or running the conversion utility in some batch mode.

- For the first approach the Jython interpreter (www.jython.org) can be used which allows to run the Python scripts directly in the Java environment.
- The second approach means that the Python interpreter needs to be installed on the server hosting the LEAF system.

For purpose of modularity the second approach was chosen. This will allow to exchange the scripts with some other software later on or even drop this functionality on server side in case that all involved archives can provide their records in EAC format.

As a consequence of the second approach the records to be converted are stored temporarily into the file system of the server. Following that the conversion utility will be run against the stored file(s) producing the appropriate conversion results as file(s) as well. These file(s) are picked up again for insert / update of the information in the LEAF repository where appropriate linking operations can take place afterwards.

Anyhow the functionality provided in either of the 2 above mentioned cases has to foresee the following interface:

Input parameters:

- Name of input directory
- Name of file including the records in local format (e.g. 'SBB_20021108.dat')
- Identification of the local format (e.g. 'ONB')
- Number of records per output file (e.g. 1000 records per output file resulting in 2 output files – one with 1000 and the other with 967 records – if 1967 records were included in the file with the records in local format)
- Maximum file size in kB (e.g. when a file reaches the size of 100kB a new file will be created)
- Name of output directory
- Name of output file(s)
- Name of error file

Output:

- Number of output file(s)

- Result file(s) with the names created from the output file name and a serial number (e.g. 'ONB_20021108_1.xml', 'ONB_20021108_2.xml' ...)
- Notification of conversion problems in the defined error file specifying the line number in the original file and an indication of what was wrong with the input or the conversion

3.2.2 Conversion from EAC format into the export formats defined for LEAF

The conversion from EAC formatted records into one of the export formats defined for LEAF (in [MRAD-2]) will be done via the use of XSL style sheets. The formats to be provided to LEAF by the conversion utilities contain

- MAB2
- UNIMARC
- MARC21
- Local format of the particular record

UoB which works on the conversion utilities has announced that a potential loss of information may be introduced due to not 100% perfect crosswalks between the different formats and the EAC format seems to be the only reliable one.

3.3 Connecting from external systems

The purpose of this interface is to provide functions which allow to query the LEAF system from external systems (e.g. the MALVINE system) and to extract information which is necessary to query bibliographic sites with specific person identifications.

The connection provided for external systems is given through an XML over HTTP interface. This interface will allow to implement workflows in external systems making use of the LEAF search functionality. Therefore the interface will provide the following functions:

- Get results for a search with given names
- Get detailed information for a particular result record (person)

3.3.1 Get results for a search with given names

Attributes (where at least one has to be filled) for this kind of search are:

- Name
- Date of birth
- Date of death
- ID of (national) authority file (as available)
- LEAF record ID

The results provide the relevant parts from EAC including:

- ID of the LEAF record
- Name
- Date of birth
- Date of death

3.3.2 Get detailed information for a particular result record

The attribute to get the detailed information for a record are :

- ID of the LEAF record

The result provides the EAC record from the LEAF system.

3.3.3 Use case MALVINE

The general procedure for MALVINE using the above described interface is:

- The MALVINE user opens a MALVINE window for the search for persons in the bibliographic search form
- The MALVINE user enters and submits his (biographic) query to MALVINE
- The MALVINE system queries LEAF to receive a brief list of identified persons (see chapter 3.3.1)
- The MALVINE user chooses an entry in the brief list of the resulting persons in case he wants to see more about that person
- In case the user has chosen a particular person in the brief list MALVINE views the details of the resulting person to the user utilising LEAF (see chapter 3.3.2)
- The MALVINE user chooses one person from the brief list or the particular person selected in the previous steps to return to the MALVINE search page
- The MALVINE search page will use the (local OPAC) IDs of the chosen persons for the searches on the different OPAC systems; in case that an archive was not in the list returned from LEAF a search using the name as selected will be done on that corresponding OPAC

Of course the example above is possible for other applications in the same way. Other features of LEAF (e.g. registration ...) will not be possible at MALVINE level.

3.4 Connecting to external systems

The LEAF system foresees that information about a data provider's online system can be used to get relevant information for a particular person record from that data provider's online system. The way this is done is performed in several steps:

- The data provider has to check the feasibility of this approach in his specific situation (URL available for searching, identification of particular records for that URL available in the records structure)
- The general details (URL for searching e.g. <http://www.hmc.gov.uk/nra/searches/Pldocs.asp?P=>) of the identified service (if exists) are added to the provider configuration within the maintenance suite
- The central LEAF system extracts the general information about the provider's system from the configuration provided by the maintenance suite
- The central LEAF system extracts the identification for particular records from the resulting records' SYSKEY (e.g. 26486)
- The central LEAF system creates a click able link from the above general and particular information (e.g. <http://www.hmc.gov.uk/nra/searches/Pldocs.asp?P=26486> pointing to John Smith).

3.5 Maintaining the connections to external resources

The purpose of this interface is to provide mechanisms which allow to store information about sites integrated in the harvesting process of LEAF which need to be configured. This means that information has to be given on how to access the servers (address, port ...) as well as available attributes if foreseen for a protocol. The maintenance of the above information is done in the maintenance suite ([CompTool]).

3.5.1 New configuration available

Whenever any kind of new data provider registers successfully to the LEAF maintenance suite the latter will trigger an event at the LEAF system telling that an updated configuration is available. For that purpose a URL will be called with parameter name of the configuration file to be downloaded.

The LEAF system is then allowed to download the new / updated configuration file from a URL provided by the LEAF maintenance suite.

3.5.2 Server status list

Whenever the LEAF maintenance suite detects that there is a change of availability at one or more data providers' server the LEAF maintenance suite will trigger an event at the LEAF system telling that an updated server status list is available. For that purpose a URL will be called with parameter the file name of the online server list to be downloaded.

The LEAF system is then allowed to download the server status list from a URL provided by the LEAF maintenance suite.

3.5.3 Configuration download interface

The connection provided at the LEAF maintenance agency is given through a web interface which allows the LEAF system to download an XML file with the information needed.

Information to be included:

- **FTP**
 - Organisation
 - format (for conversion scripts and style sheets)
 - Host
 - User
 - Password
 - file names for data file and version file
 - all records Y/N
 - zipped Y/N
- **Z39.50**
 - Organisation
 - format (for conversion scripts and style sheets)
 - Host
 - Port
 - database name
 - attribute combinations (especially the combination for identifying one particular record)
 - record syntax
- **OAI**
 - Organisation
 - format (for conversion scripts and style sheets)
 - Host
 - Port

3.5.4 Server status list

The connection provided at the LEAF maintenance agency is given through a web interface which allows the LEAF system to download an XML file with the information needed.

Information to be included:

- **Name of the server**
- **Online / offline status**

Figures

Figure 1: The interfaces from / to the LEAF central system

3

Tables

Table 1: Internal and external interfaces4

Table 2: Prerequisites on archive and LEAF server side4

Definitions, Acronyms and Abbreviations

The following list provides the definitions and abbreviations used within this document.

- ADD** Architectural Design Document
- BIB-1** Attribute set for use attributes in Z39.50
 - DT** Document Template
 - DTG** Document Template Guidelines
- EAC** Encoded Archival Context (Initiative)
- FTP** File Transfer Protocol
- HTTP** Hypertext Transfer Protocol
- ICD** Interface Control Document
- IST** Information Society Technologies
- JDBC** Java Database Connectivity
- JNDI** Java Naming and Directory Interface
- kB** Kilo-Byte
- LEAF** Linking and Exploring Authority Files
- MAB2-PND/GKD** "Maschinelles Austauschformat für Bibliotheken"-Personennamen/
Körperschaftsnamen
- MALVINE** Manuscripts and Letters via Integrated Networks in Europe
- MARC(21)** (Concise) Machine-Readable Cataloguing
 - OAI** Open Archives Initiative (Protocol)
 - OPAC** Online Public Access Catalogue
- SUTRS** Simple Unstructured Text Record Syntax
- UNIMARC** Universal MARC Format
 - URL** Uniform Resource Locator
- USMARC** US MARC Format
 - WP** Work Package
 - XML** Extensible Markup Language
 - XSL** Extensible Stylesheet Language
 - XSLT** XSL Transformation
- Z39.50** Information Retrieval (Z39.50): Application Service Definition and Protocol Specification

Partner Acronyms:

- BL** British Library
- BN** Biblioteca Nacional
- CNS** Crossnet Systems Ltd.
- DLA** Deutsches Literaturarchiv
- FDÖP** Forschungsstelle und Dokumentationszentrum für Österreichische Philosophie

- GSA** Goethe- und Schiller-Archiv
- IMEC** Institut Mémoires de l'Edition Contemporaine
- JRS** JOANNEUM RESEARCH
- NUK** National and University Library, Ljubljana, Slovenia
- ÖNB** Österreichische Nationalbibliothek
- RA** Riksarkivet
- SBB** Staatsbibliothek zu Berlin
- SNL** Swiss National Library
- UCM** Biblioteca de Universidad Complutense de Madrid
- UoB** University of Bergen

References

The following list shows references that are made within this document.

- [ADD]** Architectural Design Document; LEAF Deliverable D6.2; JRS
- [CompTool]** Model Compatibility Design; LEAF Deliverable D9.1; CNS
- [ConvTool]** Report on the XML encoding and conversion tools for the name data; LEAF Deliverable D4.3; UoB
- [CM]** Coverage Matrix; LEAF Deliverable D6.1; JRS
- [DoW]** LEAF ANNEX 1, "Description of Work"
- [FTP]** RFC 959: File Transfer Protocol;
<http://www.w3.org/Protocols/rfc959/Overview.htm>; March 3rd 2003
- [MRAD-1]** José Borbinha (BN), Nuno Freire (BN); Model Requirements Analysis Document – Part 1; LEAF Deliverable D5.1 v1.0; May 8th, 2002
- [MRAD-2]** Hans-Jörg Lieder (SBB), Max Kaiser (ÖNB); Model Requirements Analysis Document – Part 2; LEAF Deliverable D5.1 v1.0; May 14th, 2002
- [OAI]** Open Archives Initiative <http://www.openarchives.org>; November 29th 2002
- [RND]** Gunnar Karlsen (UoB); Report on a Recommended Name DTD; LEAF Deliverable D4.1 v1.0; December 15th, 2001
- [XML-Schema]** XML Schema, <http://www.w3.org/XML/Schema>; Mai 13th 2002
- [XSL]** Extensible Stylesheet Language (XSL); Version 1.0; W3C Recommendation 15 October 2001; <http://www.w3.org/TR/xsl>; November 29th 2002
- [XSLT]** XSL Transformations (XSLT); Version 1.0; W3C Recommendation 16 November 1999; <http://www.w3.org/TR/xslt>; November 29th 2002
- [Z39.50]** Z39.50 Maintenance Agency Page; <http://lcweb.loc.gov/z3950/agency/>; March 3rd 2003