

reUSE  
reUSE Digital Master Files of Printed Material  
**Contract No. 11173**

**D1.3 Estonian Demonstrator**  
Version 1.0

National Library of Estonia  
Krista Kiisa, Liina Enok, Urmas Sinisalu  
March 2005

1	Introduction.....	3
2	Technical Documentation .....	4
2.1	Overview of System Architecture.....	4
2.1.1	Hardware.....	4
2.1.2	Software .....	4
2.1.3	Fedora .....	6
2.2	Publisher interface .....	9
2.3	Object Management Interface.....	10
2.4	Search interface.....	12
2.5	Depositor management interface .....	13
2.6	File processing .....	14
2.7	Metadata.....	14
2.7.1	Fedora object.....	14
2.7.2	NLE defined metadata .....	15
2.8	Backup procedures.....	15
2.9	User Management Interface.....	15
2.10	Ingest Buffer Database.....	15
2.11	FTP Management.....	23
3	Workflow models .....	24
3.1	Ingest.....	24
3.2	Preservation.....	25
3.3	Access .....	26
4	Submission Agreement .....	27
5	Käsiraamat väljaandjale (Handbook for publishers).....	31
5.1	Mida pakub DIGitaalne ARhiiv <i>DIGAR</i> . (Introduction to Services provided by DIGital ARchive <i>DIGAR</i> ).....	31
5.2	Mida kogume (What to Deposit. Deposit Formats).....	32
5.3	Kuidas liituda (How to Join the Archive).....	33
5.4	Originaalfailide üleandmise kord: (Submission procedures).....	34
5.5	Juhendmaterjalid (Guides).....	35
5.5.1	Kasutusjuhend failide transportimiseks FTP abil (Guide for Uploading the Files).....	35
5.5.2	Juhend failide kirjeldamiseks ning kasutuspiirangute määramiseks (Guide for Filling Descriptive Metadata Form and Applying Access Restrictions) .....	37

## **1 Introduction**

Deliverable D1.3 Estonian Demonstrator gives an overview of the ongoing implementation activities at reUSE project at the end of WP1.

The paper aims to show the results of setting up a digital repository in the National Library of Estonia for collecting, preserving and making available digital original files of public sector publications in order to assure long-term preservation of the content.

The paper consists of:

- Technical descriptions of the implemented system
- Workflow models
- Submission agreement
- Handbook

The first two parts of the document are in English.

Materials addressed to the publishers are in Estonian and are not translated into English at this stage. These materials are handbook with detailed guidelines for submitting files and the agreement template for establishing stable relations between public sector bodies regulating access rights and intellectual property rights.

## **2 Technical Documentation**

### **2.1 Overview of System Architecture**

The digital repository system of NLE is built on Fedora software version 2.0. Major components of the system are (See Figure 1):

- Fedora server version 2.0 as underlying digital object repository. Fedora server includes embedded Tomcat application server with SOAP extension (Axis).
- NLE object management interface. This interface interact with the FEDORA repository via SOAP interfaces provided by Fedora's Management-API
- Search interface. This interface interacts with the FEDORA repository via SOAP interfaces provided by Fedora's Access-API.
- MySQL database for ingest buffer. This database is used by publisher interface and object management interface.  
The same MySQL server also hosts Fedora's internal database for object storage.
- Publishers interface. Enables to describe files in Publisher's ftp-directory
- User Management interface.

#### **2.1.1 Hardware**

- Fujitsu-Siemens PRIMERGY TX300r 2-way Rack
  - 2 x Intel Xeon DP 3,06Ghz/1MB L3 cache processor
  - 2GB DDR RAM PC2100 ECC memory
  - 2 x 10/100/1000Mbps Ethernet LAN
  - 4 channel U320 SCSI RAID Controller
  - Internal Tape Drive LTO2 Ultrium 200/400GB
  - 2 x 36GB hot swap U320 SCSI HDD (system disks)
- Fujitsu-Siemens PRIMERGY SX30 - external disk system
  - 2 x U320 SCSI channels
  - 14 x 146GB hot swap HDD (data disks)

System disks are configured into RAID 1 and data disks into RAID 5.

#### **2.1.2 Software**

- Debian GNU/Linux 3.1 ('sarge'), kernel 2.6.8.1
- MySQL 4.1.10 for Ingest Buffer database and Fedora database
- Apache 2.0.53, PHP 4.3.10 for Depositor Management interface
- Fedora 2.0
  - Jakarta-tomcat-5.0.28
- Jakarta Tomcat 5.5.7 hosts publishers interface, object management interface and search interfaces.
- Java 1.50\_02
- ProFTP 1.2.10-10
- Bacula 1.36.1-1 for backup
- Clamav 0.83-3 for virus check
- Plop 2.1.0 for PDF processing
- Xpdf 3.00-12 for PDF processing

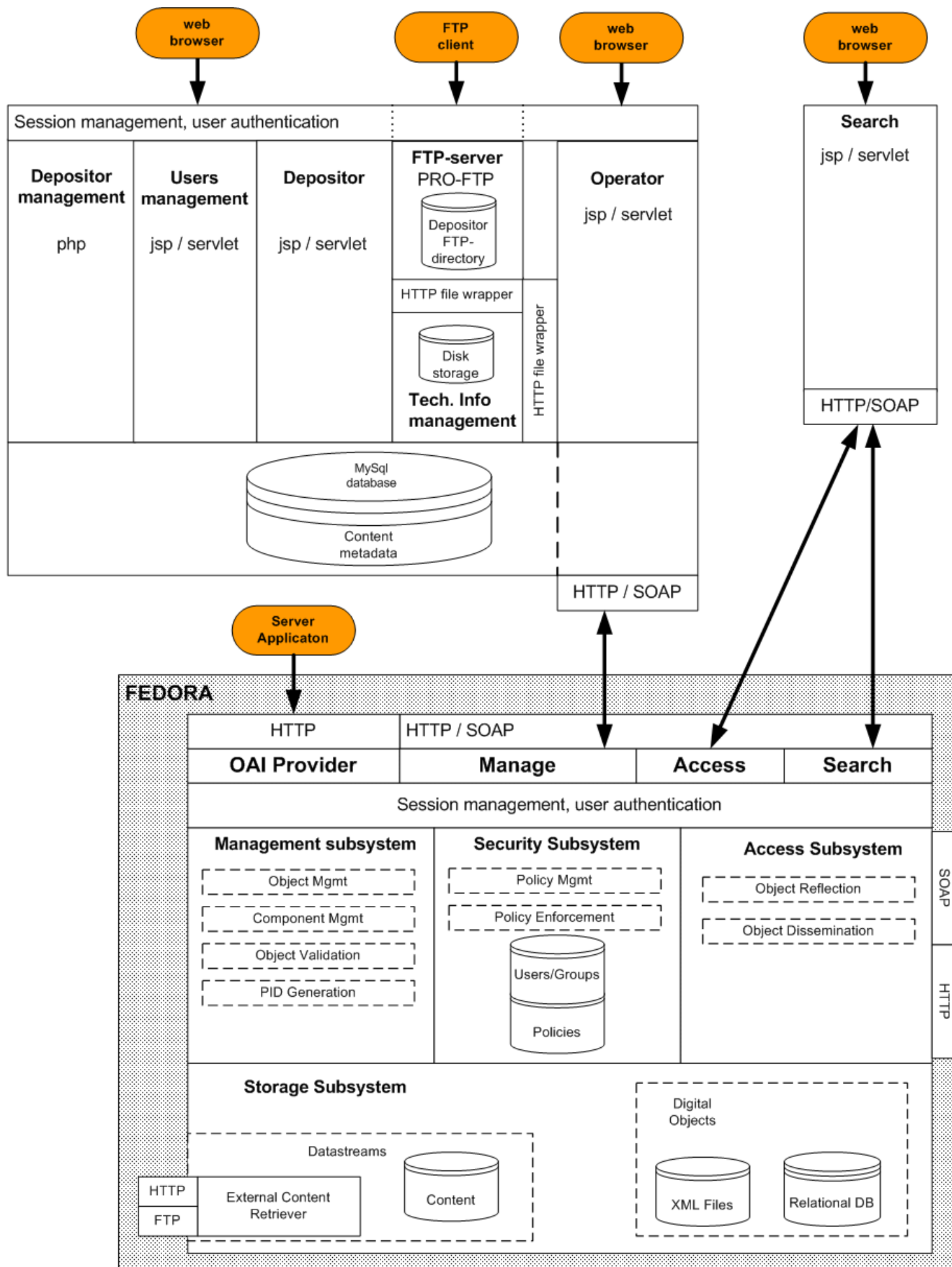


Figure 1 System Architecture Overview

### 2.1.3 Fedora<sup>1</sup>

#### 2.1.3.1 *Major Features of the Fedora Repository*

- **Open Source** - The Fedora repository system is open source software licensed under the Mozilla Public License<sup>2</sup>.
- **Flexible Digital Object Model** - The Fedora digital object model provides the flexibility to create kinds of objects including documents, images, electronic books, multi-media learning objects, datasets, metadata, and more. Please refer to the Fedora Digital Object documentation for more details.<sup>3</sup> The model supports the aggregation of one or more content items as "Datastreams". The bytestream content of a Datastream can be any media type and can be either stored locally in the repository, or referenced by a digital object (i.e., content stored outside the repository). The Fedora object model also provides a mechanism (known as a Disseminator) for associating services with an object to produce dynamic or computed content from digital objects.
- **Content Versioning** - Any modifications made to a Datastream or Disseminator through the Fedora management interface (API-M) will automatically result in the creation of a new version of that Datastream or Disseminator. The Fedora object contains a record of all versions, thereby creating a history of how objects changed over time. Additionally, Fedora maintains an audit trail record of the nature of the object change events. Access to prior versions is available by adding a date-time stamp to requests on the Fedora Access service (API-A and API-A-LITE).
- **XML Ingest and Export** - Digital objects can be submitted to a Fedora repository as XML-encoded files that conform to either the Fedora Object XML (FOXML)<sup>4</sup> schema or an extension of the Metadata Encoding and Transmission Standard (METS)<sup>5</sup> schema. In future releases of Fedora other formats will be supported including MPEG2/DIDL and METS 1.3
- **XML Storage** - By default, Fedora digital objects are stored in a Fedora repository as XML-encoded files that conform to Fedora Object XML (FOXML) format. As previously mentioned, objects can be both ingested and exported in other XML formats. Content bytestreams that are aggregated as Datastreams in a digital object are stored in their native formats in the repository persistent storage area.
- **Object-to-Object Relationships** - Fedora provides the ability to assert object-to-object relationships. Object-to-Object relationship metadata is a way of asserting various kinds of relationships among Fedora objects including the notion that an object is a member of a collection, is considered a part of

---

<sup>1</sup> For full Fedora System documentation refer to <http://www.fedora.info/download/2.0/userdocs/>

<sup>2</sup> <http://www.mozilla.org/MPL/MPL-1.1.html>

<sup>3</sup> <http://www.fedora.info/download/2.0/userdocs/digitalobjects/index.html>

<sup>4</sup> <http://www.fedora.info/definitions/1/0/foxml1-0.xsd>

<sup>5</sup> <http://www.fedora.info/definitions/1/0/mets-fedora-ext.xsd>

another object, is a derivation of another object, is a description of another object, or is equivalent to another object. A default set of common relationships is defined in the Fedora relationship ontology, although other community or user-defined relationships may also be asserted. These relationships are automatically indexed by Fedora as part of the Fedora Resource Index.

- **Access Control and Authentication** - Release 1.2 includes a simple form of access control to provide access restrictions based on IP address. IP range restriction is supported in both the Management and Access APIs. In addition, the Management API is protected by HTTP Basic Authentication. In the upcoming Fedora 2.1 release we will introduce a new Access Control and Authentication module that will include the ability to enforce fine-grained access control policies expressed using XACML! (Winter/Spring 2005)
- **Simple Search** - Fedora automatically creates two indexes of the repository. The default search index is a simple index of the repository searchable by object properties and object Dublin Core elements. As of Fedora 2.0 there is the new RDF-based Resource Index that includes more information about objects plus object-to-object relationships. Both of these indexes are searchable via REST-based web service interfaces.
- **RDF-based Resource Index** - As of Fedora 2.0 there is the new RDF-based Resource Index that includes more information about objects plus object-to-object relationships. Both of these indexes are searchable via REST-based web service interfaces.
- **OAI Metadata Harvesting Provider** - The OAI Protocol for Metadata Harvesting is a standard for sharing metadata across repositories. Every Fedora digital object has a primary Dublin Core record that conforms to the oai\_dc schema<sup>6</sup>. This metadata is accessible using the OAI Protocol for Metadata Harvesting, v2.0.
- **Migration Utility** - A new migration utility is provided to perform mass export and mass ingest of objects. At the core, the migration utility is built upon two newly enhanced command-line functions: fedora-export and fedora-ingest. Used together, these two command line functions can support a variety of scenarios involving moving or copying objects between repositories. The utility is general-purpose in that it can be used to copy or move objects among repositories for many reasons including upgrading from previous releases of Fedora. Client access to the migration comes in two forms: (1) run fedora-export and fedora-ingest at the command line, or (2) use the Fedora Administrator client to invoke these.
- **Batch Utility** - The Fedora repository system includes a Batch Utility as part of the Fedora Administrator client that enables the mass creation and modification of Fedora digital objects.

---

<sup>6</sup> [http://www.openarchives.org/OAI/2.0/oai\\_dc.xsd](http://www.openarchives.org/OAI/2.0/oai_dc.xsd)

- **Reporting Utility** - A reporting utility is provided providing different management views of the contents of the Fedora repository.

### 2.1.3.2 *Fedora Repository Web Service Interfaces*

The interface to the system consists of three open APIs that are exposed as web services:

- **Management API (API-M)** – defines an interface for administering the repository. It includes operations necessary for clients to create and maintain digital objects and their components. API-M is implemented as a SOAP-enabled web service.
- **Access API (API-A)** – defines an interface for accessing digital objects stored in the repository. It includes operations necessary for clients to perform disseminations on objects in the repository and to discover information about an object using object reflection. API-A is implemented as a SOAP-enabled web service.
- **Access-Lite API (API-A-Lite)** – defines a lightweight version of the Fedora Access Service that is implemented as a REST-based web service that can be invoked with a simple URL syntax.
- **Management-Lite API (API-M-Lite)** – this interface is under development. The intent is to provide a lightweight version of the Fedora Management Service implemented as a REST-based web service that can be invoked with a simple URL syntax. Currently, there is only one operation implemented, which is the getNextPID. A full version of this interface may be provided in a future release, depending on user demand.
- **Search API (part of API-A-Lite)** – this interface provides a simple search of the repository. The search operates upon the default index for the repository, which contains object properties and the default Dublin Core record for the repository. This search interface is intended as a basic field search of the object registry. It is expected that other services will be used to create to index Fedora objects at a finer-grained level than provided by the default search.
- **Resource Index Search API** – this is a new interface as of Fedora 2.0 that provides searching of the new Resource Index. The Resource Index is an RDF-based index of the Fedora repository that includes the following for each digital object:
  1. Object properties
  2. Object-to-object relationships
  3. Metadata about datastreams and disseminations
  4. Default Dublin Core record



## 2.2 Publisher interface

Interface for content providers (publishers). This interface helps publishers to organize uploaded files. Files have to be uploaded separately by using a FTP client. Each publisher has its' own FTP-directory. There is no web-upload possibility in the current version of the system, but this feature is under consideration for future development.

Main functionalities:

- Creating and describing digital objects. The web-form includes five fields for adding minimal metadata to the object: Title, Author(s), Publication Year, Copyright statement and Access restriction. Publisher can edit one object at a time
- Linking uploaded files to object and assigning linked files properties (comment, sorting order, access restriction)
- Sending object to processing. Publisher cannot send objects directly to Fedora digital object repository. After completing the object metadata and file relations are written into Ingest Buffer database and file processing scripts are invoked.

When publisher logs in to web interface, an empty object is created by the system. System inserts empty rows to the object tables in MySQL database and displays this empty object through web interface to be updated by the publisher.

Publisher may update object metadata and link uploaded files to object. Until the publisher does not call "Send data" function by pressing the corresponding button in the web interface the digital object is still in "DRAFT" state and resides in MySQL database.

Publisher may log out and may log in later to start new session – the draft's data is still there in editable form. Publisher can work only with last draft; he cannot work with multiple objects (drafts).

When publisher has completed object metadata he can send the object to processing. System then only changes status of the object and makes this object visible to operator and to inner file processing procedures. Objects are not ingested to the Fedora repository in this stage yet.

Publisher cannot edit objects once he has sent them to processing. After sending the object publisher can only see that the fact of sending object to processing has taken place from "history" list.

Architecture of publisher application is very similar to operator application, except it does not contain the components for communication to Fedora.

## 2.3 Object Management Interface

Interface for operators (librarians). Main functionalities:

- Create new publication objects.
- Edit objects entered by publishers.
- Send objects to Fedora repository.
- Edit objects already stored in Fedora repository

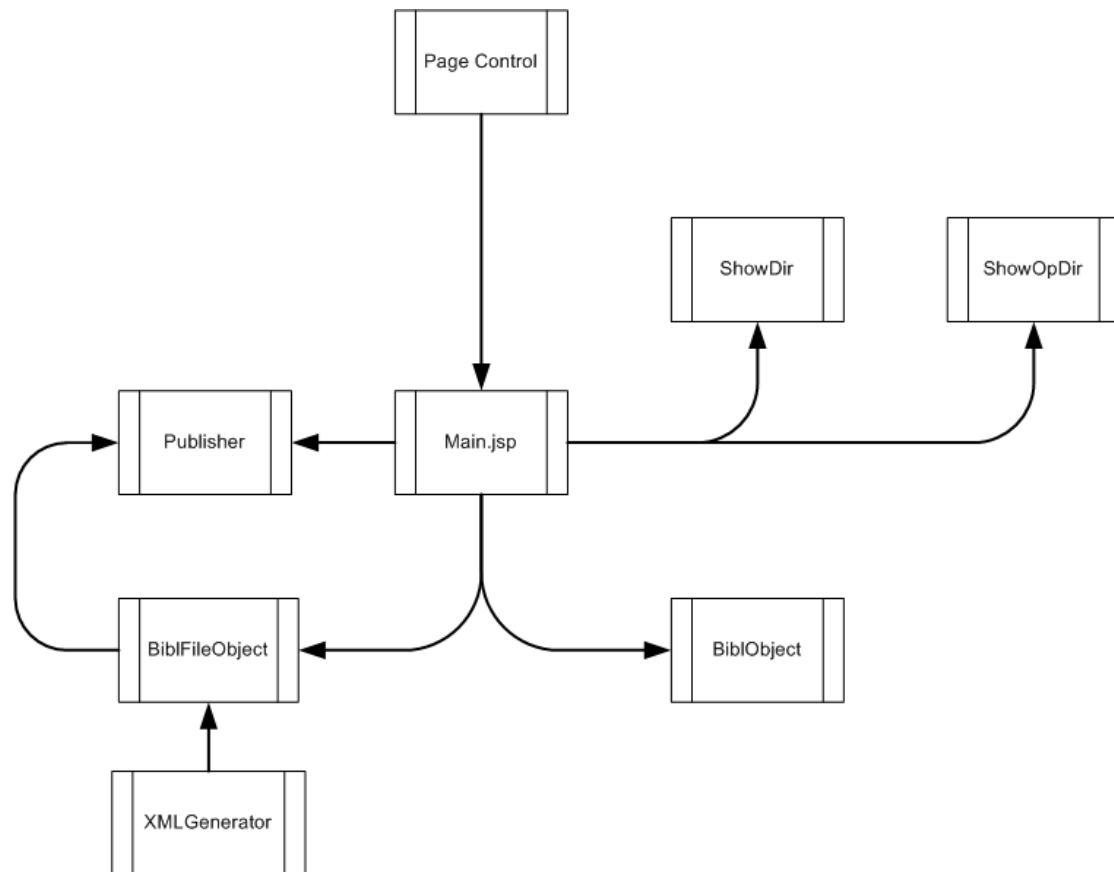
After logging in to the interface operator gets a listing of all the objects send to the Ingest buffer by Depositors. The interface also shows the date when the object was created and the status of each object in the processing chain together with the results of processing. Object management interface also shows the listings of depositors' FTP-directories and operators own FTP-directory. Operator can add files to object or remove files from object. When operator adds file then he must determine the type of file processing. There are two types of file processing: 1) gathering technical data; 2) gathering technical data and converting file to new formats. Each file linked to object must at least go through the step of gathering technical metadata.

When operator decides that object is ready for ingesting to Fedora repository "Send to Fedora" function is called by pressing corresponding button in the interface. Object metadata is gathered from Ingest Buffer database, Fedora XML-object is generated and object is ingested to repository.

Operator can also choose to delete object, this is alternative end state of object's lifecycle.

**Table 1 Operator application's JavaBean components**

BiblObject	Represents digital object in Java application. Mainly uses BIBL_OBJECT table in MySQL database. Also uses Fedora repository for data access for those objects that are already in repository.
BiblFileObject	Represents file, included in digital object. Uses BIBL_FILE table in database and Fedora repository (for those files that are already in repository)
Publisher	Represents information uploader entity (also called as „publisher“), uses DEPOSITOR table in database.
ShowDir	Contains presentation logic for browsing directories (filesystem browser)
ShowOpDir	Contains presentation logic for browsing operator directories (filesystem browser)
XMLGenerator	Contains logic for generating XML data (from database). Also contains logic for ingesting generated XML-data to Fedora repository.



**Figure 2 Object management application**

Operator's application architecture follows widely used MVC (Model View Controller) pattern. Application is a combination of three types of components: servlets, java server pages (JSP) and JavaBean components.

Servlet named „PageControl“ receives all incoming requests and acts like input filter and access controller. „PageControl“ servlet provides authentication services and forwards user requests to JSP called „main.jsp“.

„main.jsp“ acts like controller and presentation component for application, this jsp contains logic for calling business component methods. „main.jsp“ contains also presentation logic.

There are four JavaBean components (used mainly by „main.jsp“), those components contains business logic and data access logic. Separate data access component is planned but not currently in use.

## 2.4 Search interface

Query interface for different types of users (mainly external users). This interface uses Fedora API-A to process queries and display query results. This is an enhanced version of Fedora search interface<sup>7</sup>. The Search Interface provides a mechanism for searching and browsing the Fedora repository. Upon ingestion, metadata from the Fedora System Metadata section and the Dublin Core (DC) Metadata section of the object are indexed in a relational database, and may be searched using this interface. The DC Metadata section is a XML Metadata datastream in the object, where the *Datastream ID* is DC, and the XML conforms to the oai\_dc schema<sup>8</sup>. The search interface provides both simple and advanced searching via a web form included with the repository software. All queries are case insensitive. Simple Search enables queries of words and phrases occurring anywhere in that object's indexed metadata fields. Advanced Search enables fielded searching across any combination of metadata elements using string comparison operators ( = and ~ ) for string fields, and value comparison operators ( =, >, ≥, <, ≤ ) for date fields (dc:date fields may be treated as both). The wildcards, \* and ? may be used in any string-based query. Dublin Core metadata can also be harvested via OAI-PMH 2.0. Anonymous reader only has access to “viewing files”. Nobody, but the depositor has access to the original files.

---

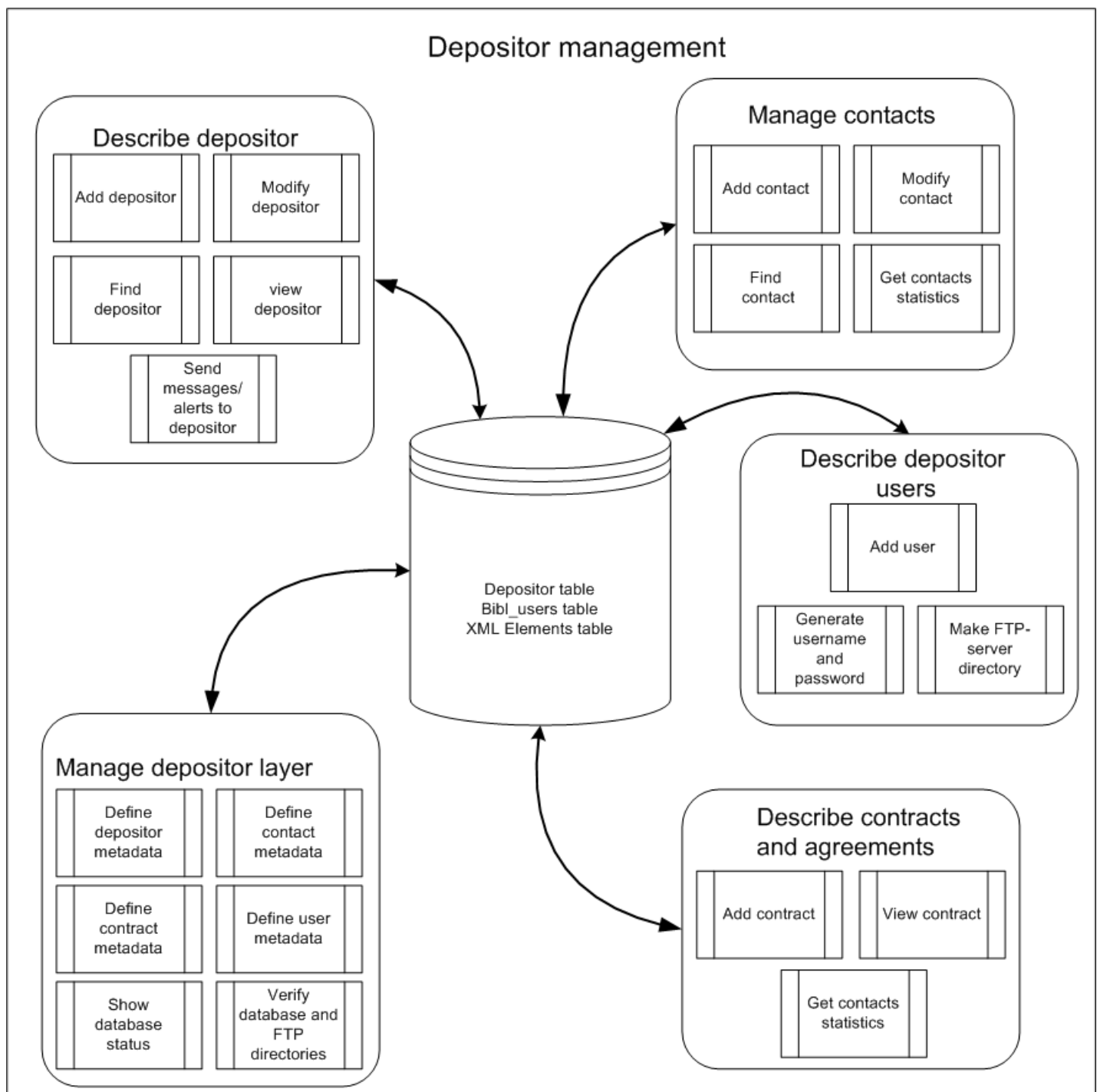
<sup>7</sup> <http://www.fedora.info/download/2.0/userdocs/server/webservices/search/index.html>

<sup>8</sup> [http://www.openarchives.org/OAI/2.0/oai\\_dc.xsd](http://www.openarchives.org/OAI/2.0/oai_dc.xsd).

## 2.5 Depositor management interface

This is a PHP-MySQL interface for managing depositor accounts. Main functionality:

- Adding new depositors into database
- Modifying existing depositors' data
- Creating FTP- and web-user accounts for depositors



**Figure 3 Depositor Management**

## 2.6 File processing

File processing procedures include utilities for:

- moving and copying files on local filesystem. (movefile.pl) When Send Data function is called from publisher interface or object management interface a script is started that moves files from publisher's FTP-directory to dedicated Ingest buffer directory. Files are renamed according to pattern DepositorID\_ObjectID\_FileID
- converting files – creating new files from the submitted original files. File conversion scripts are run by cronjob. PLOP 2.1.0 is used for creating web-optimized files.
- reading technical metadata from files. Scripts run by cronjob gather technical metadata. Standard Linux utilities are used for reading generic information (size, checksum). Xpdf 3.00-12 is used for gathering information from PDF files.

At the end of processing those processes save some processing results in database. Object states “PROCESSED BY SYSTEM – OK” and “PROCESSED BY SYSTEM – ERROR” are essentially the composite states – they are derived from states of individual files, processed by routines.

When all files are processed correctly – then object state may be “PROCESSED BY SYSTEM – OK”. When at least one file processing didn't succeed – then object may be in “PROCESSED BY SYSTEM – ERROR”-status.

## 2.7 Metadata

Digital objects are stored as Fedora Object XML files. The Fedora extension of METS will continue to be supported as an ingest and export format. In upcoming releases, Fedora will also support other formats for ingest and export such as METS 1.4 and MPEG21/DIDL.

### 2.7.1 Fedora object

Fedora object is an XML object that conforms to the FOXML schema<sup>9</sup>. The basic components of a Fedora digital object are:

- **PID:** a persistent, unique identifier for the object
- **Object Properties:** a set of system-defined descriptive properties that are necessary to manage and track the object in the repository.
- **Datastream(s):** the component in a Fedora object that represents MIME-typed content item. An object can have one or more Datastreams. The content of a Datastream can be either data or metadata, and this content can either be stored internally in the Fedora repository, or stored remotely (in which case Fedora holds a pointer to the content in the form of a URL). Every object has one Dublin Core metadata datastream by default.
- **Disseminator(s):** the component in a Fedora object that associates an external service with the object for the purpose of providing extensible views of the object or of its datastream content. An object can have zero or more Disseminators.

---

<sup>9</sup> <http://www.fedora.info/definitions/1/0/foxml1-0.xsd>

### 2.7.2 NLE defined metadata

Within Fedora Digital object NLE will store descriptive, technical and administrative metadata as internal XML datastreams:

- Descriptive metadata
  - The default Dublin Core metadata record that conforms to the oai\_dc schema<sup>10</sup>
  - If necessary an extended DublinCore record is created for more precise description
  - Each file within the object has a technical metadata record consisting of 5 compulsory fields for generic data (size, MD5, mimetype, format, date created) and several format specific fields. Format specific fields for PDF are tagged, pagecount, pagesize, fonts\_embedded. The list of data elements may be extended depending on a specific format.
  - The structure of the object (file order, file status) is described in filegroup datastreams. Each filegroup has status attribute with value either original, archival or viewing. Each file element within filegroup has Order attribute.
  - There is rights metadata record containing data about the owner of the object (depositor) and access restrictions either on object or file level.
  - Relations to other objects such as for example belonging to a certain collection are defined in Fedora reserved RELS-EXT datastream.

## 2.8 Backup procedures

- Backup on a second server located in house. MySQL database and files are copied periodically to the second server.
- Regular tape backup (Bacula 1.36.1-1)

## 2.9 User Management Interface

Servlet application for managing users of the system's Publisher interface, Object Management Interface and Depositor Management Interface. Functionalities are creating new users, assigning user privileges, modifying user privileges.

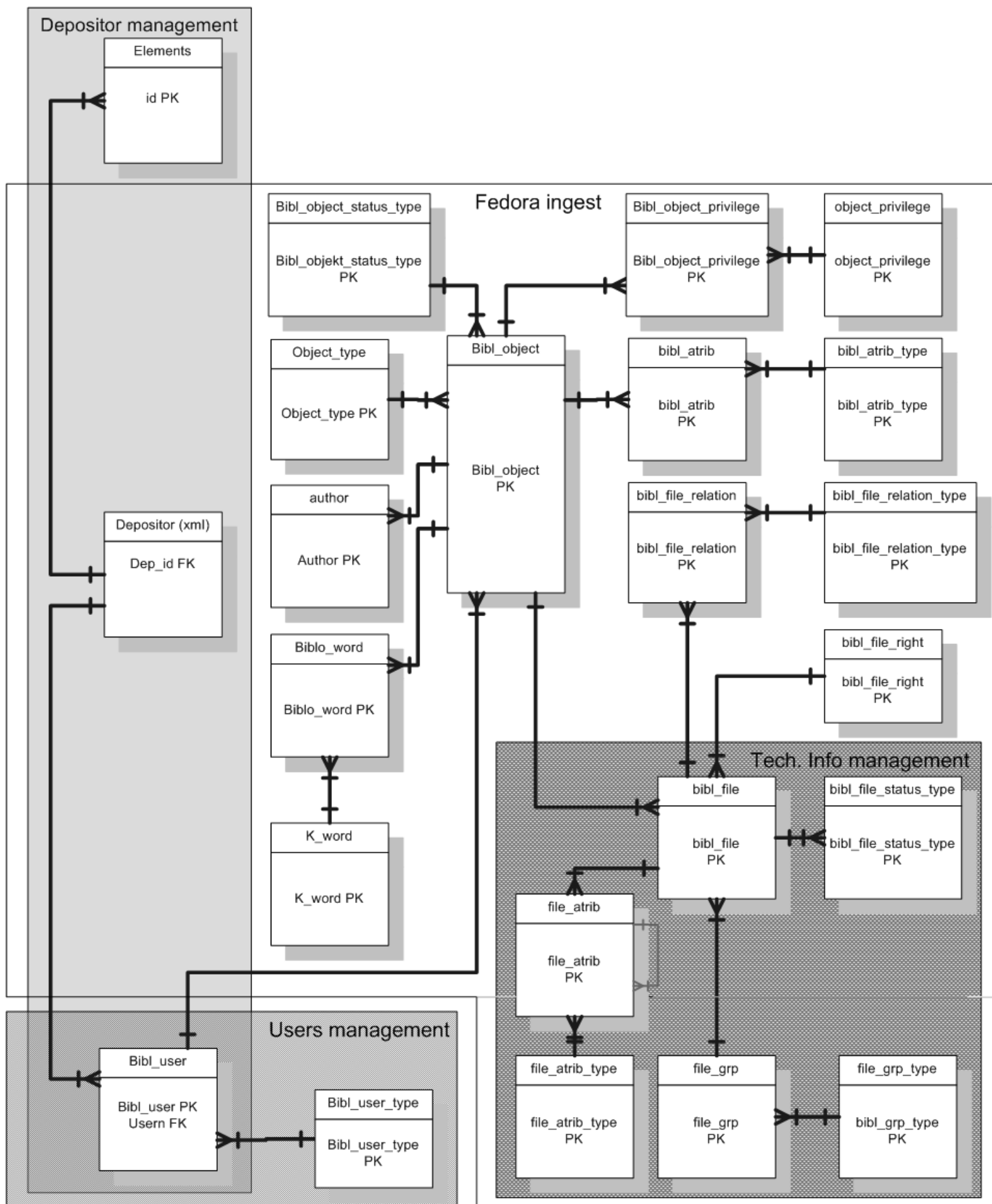
## 2.10 Ingest Buffer Database

This is MySQL database for gathering and organizing input data (from publishers) needed for digital object formatting (XML) and exporting them to Fedora. Main table is BIBL\_OBJECT - table for publication data. BIBL\_OBJECT and some related tables (AUTHOR, BIBL\_FILE) may be filled by publishers after uploading publication files to ftp-server.

Operator's interface will operate with same tables as publisher's interface (mainly BIBL\_OBJECT, AUTHORS, BIBL\_FILE), allowing operators to complete publisher's input and generate XML-objects for Fedora export.

---

<sup>10</sup> [http://www.openarchives.org/OAI/2.0/oai\\_dc.xsd](http://www.openarchives.org/OAI/2.0/oai_dc.xsd)



**Figure 4 Ingest Buffer Database**



Column/tables that are not commented are not in use at the moment or their names are self-describing. all primary keys are counters and named after name of the tables.

#### Table/Column

##### Table Name: author

Authors, persons related to publication

Table Column Name	Description
author	primary key
first_name	Authors first name
last_name	Authors last name
bibl_object	reference to publication (table BIBL_OBJECT)

##### Table Name: bibl\_atrib\_type

Table Column Name	Description
bibl_atrib_type	Primary key
name	Type name
description	Description of type
bibl_atrib_class	
orderb	
uniq	

##### Table Name: bibl\_atrib

Attribute table for publications (table BIBL\_OBJECT) and publication files (table BIBL\_FILE). Dynamic data-structure for publications and files - each publication or file may have different set of attributes. Those attributes are for example technical data for files.

Table Column Name	Description
bibl_atrib	Primary key
bibl_object	references to publication , to attribute „owner“ (table BIBL_OBJECT) - filled if this is publications attribute
value_text	attribute value
atrib_name	Attribute type name
bibl_atrib_type	references to attribute type (table BIBL_ATRIB_TYPE)

##### Table Name: bibl\_collection - not in use at the moment

Table Column Name	Description
bibl_object	
name	
bibl_collection	

##### Table Name: bibl\_collection\_part - not in use at the moment

Table Column Name	Description
bibl_object_collection	
bibl_collection	
bibl_collection_part	

**Table Name:** bibl\_file

- publication file. Related to publication (table BIBL\_OBJECT)

Table Column Name	Description
bibl_file	primary_key
id	
file_grp	
bibl_file_right	Reference to file usage rule (table BIBL_FILE_RIGHT)
status	File processing status
bibl_object	reference to publication (table BIBL_OBJECT)
desc_object	reference to file description (same table BIBL_OBJECT is used for that)
file_catalog	
name	file name
created	created date
created_by	created by user (reference to user table)
orderb	order

**Table Name:** bibl\_file\_relation

relation table, stores relations that exist between two tables (for example when one file derives from other)

Table Column Name	Description
bibl_file_relation	primary key
bibl_file	reference to one file (table BIBL_FILE)
description	relation description
bibl_file1	reference to other related file (table BIBL_FILE)
bibl_file_relation_type	reference to relation type (table BIBL_FILE_RELATION_TYPE)

**Table Name:** bibl\_file\_relation\_type

Table Column Name	Description
bibl_file_relation_type	primary key
name	name of the relation type

**Table Name:** bibl\_file\_right

Table Column Name	Description
bibl_file_right	primary key
name	name of the file right

**Table Name:** bibl\_file\_status\_type

Table Column Name	Description
bibl_file_status_type	primary key
name	name of the status
description	description

**Table Name:** bibl\_object

Table for publication data, filled originally by publishers, in some cases filled by operators. Most important/ central table in this database.

Table Column Name	Description
bibl_object	primary key
object_type	reference to object type (table OBJECT_TYPE)
bibl_object_status_type	Reference to object status type (table BIBL_OBJECT_STATUS_TYPE)
file_catalog	
title	publications title
_number	publications number
_year	issue date
_format	publications format
_pages	publications page count
created	created date
created_by	created by user (reference to user table)
updated	update date
updated_by	updated by user (reference to user table)
fedora_id	
fedora_event	
fedora_sender	
status	current status of object
status_date	last status change date
publisher	reference to publisher (table PUBLISHER)
parent_object	
file_type	
keywords	
publishername	publishers name (filled if different from referenced publisher)
publisher_date	
copyright	

**Table Name:** bibl\_object\_privilege – not in use at the moment

Table Column Name	Description
status	
created_by	
updated	
updated_by	
bibl_object_privilege	
created	
object_privilege	
bibl_object	

**Table Name:** bibl\_object\_status\_type

Table Column Name	Description
bibl_object_status_type	Primary key, type code
name	Type name

**Table Name:** bibl\_user

Table Column Name	Description
bibl_user	Primary key
created	
status	
last_name	
updated_by	
created_by	
updated	
root_catalog	
id	
first_name	
passw	password
bibl_user_type	
usern	username

**Table Name:** bibl\_user\_type

Table Column Name	Description
bibl_user_type	Primary key, type code
name	Type name

**Table Name:** biblo\_word – not in use at the moment

Table Column Name	Description
bibl_object	
value_text	
k_word	
biblo_word	

**Table Name:** depositor

Table Column Name	Description
id	Primary key
dep_id	Depositors code
element	
value	
type	
add_date	
user	
jrk	
param	
kuva	
namespace	
grupp	

**Table Name:** elements

Table Column Name	Description
id PK	element identifier (Unique element id)
display	Screen name of element
xmlname	Element - XML name
namespace	Element - XML namespace
datatype	Type of value
kohustuslik	Element is obligatory or not
tase	Element level
attribute	xml element attribute
description	Element description
tip	Element alt text in screen
kuva	HTML page name
rights	Element is read only or not
user_id	User_id who added the element
kp	added date
Jrk_nr	Element order in screen
length	Element length
kuva_len	Element screen length
retype	Element is repeatable or not

**Table Name:** file\_atrib

Table Column Name	Description
parent_atrib	
atrib_name	
atrib_class	
atrib_level	
file_atrib_type	
value_text	
biol_file	
file_atrib	

**Table Name:** file\_atrib\_type

Table Column Name	Description
file_atrib_type	Primary key, type code
name	Type name
description	
file_atrib_class	
orderb	

**Table Name:** file\_catalog

Table Column Name	Description
id	
cat_location	
file_catalog	

**Table Name:** file\_grp

Table Column Name	Description
file_grp	Primary key
file_grp_type	Reference to FILE_GRP_TYPE
name	File group name
orderb	order

**Table Name:** file\_grp\_type

file group type (V – files for presentation, O – original files, A – archive files)

Table Column Name	Description
file_grp_type	Primary key, type code
name	Name of type

**Table Name:** k\_word – not in use at the moment

Table Column Name	Description
k_word	
estonian_w	
english_w	

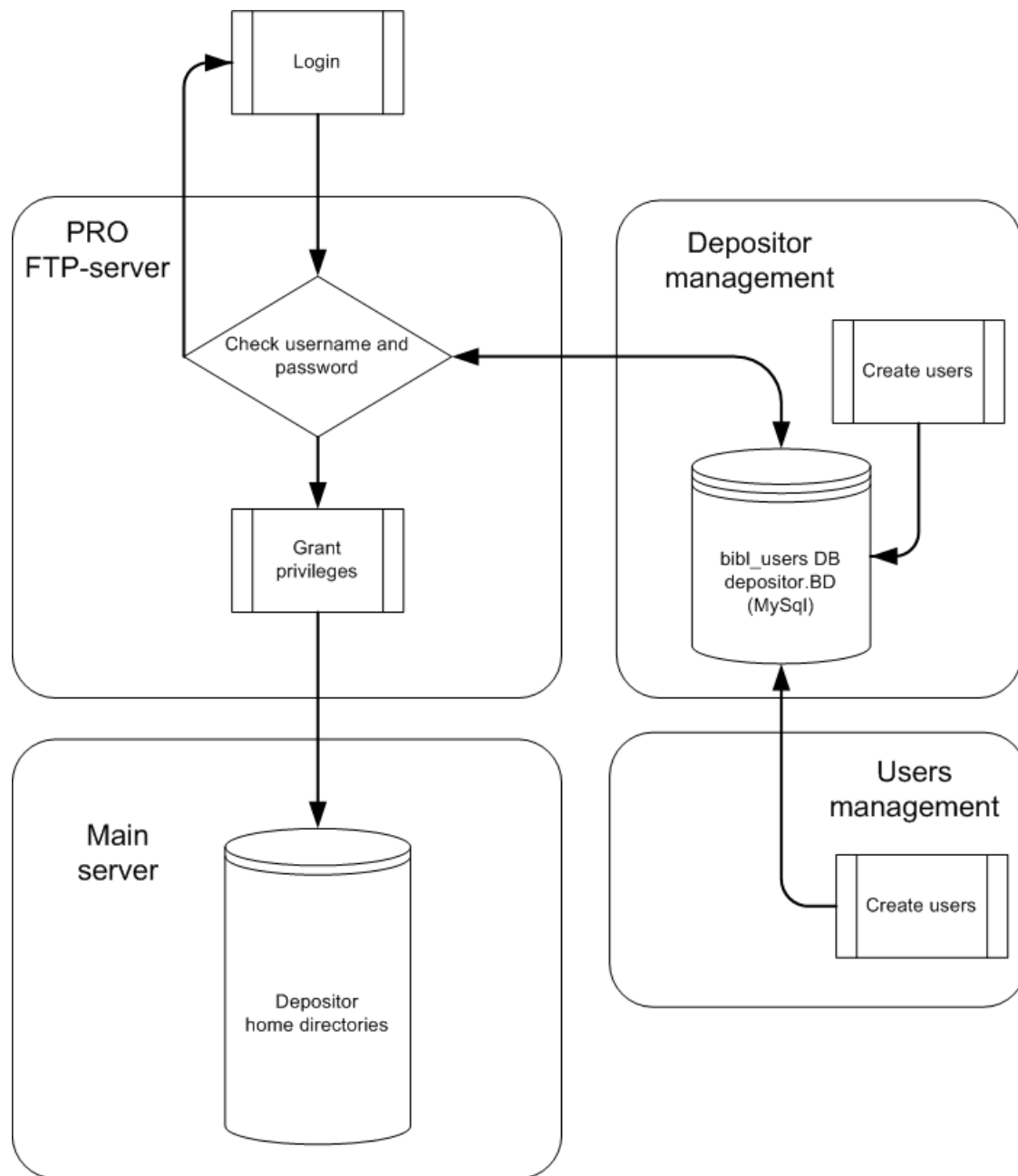
**Table Name:** object\_privilege – not in use at the moment

Table Column Name	Description
object_privilege	
name	
description	

**Table Name:** object\_type

Table Column Name	Description
object_type	Primary key
name	Name of type
description	

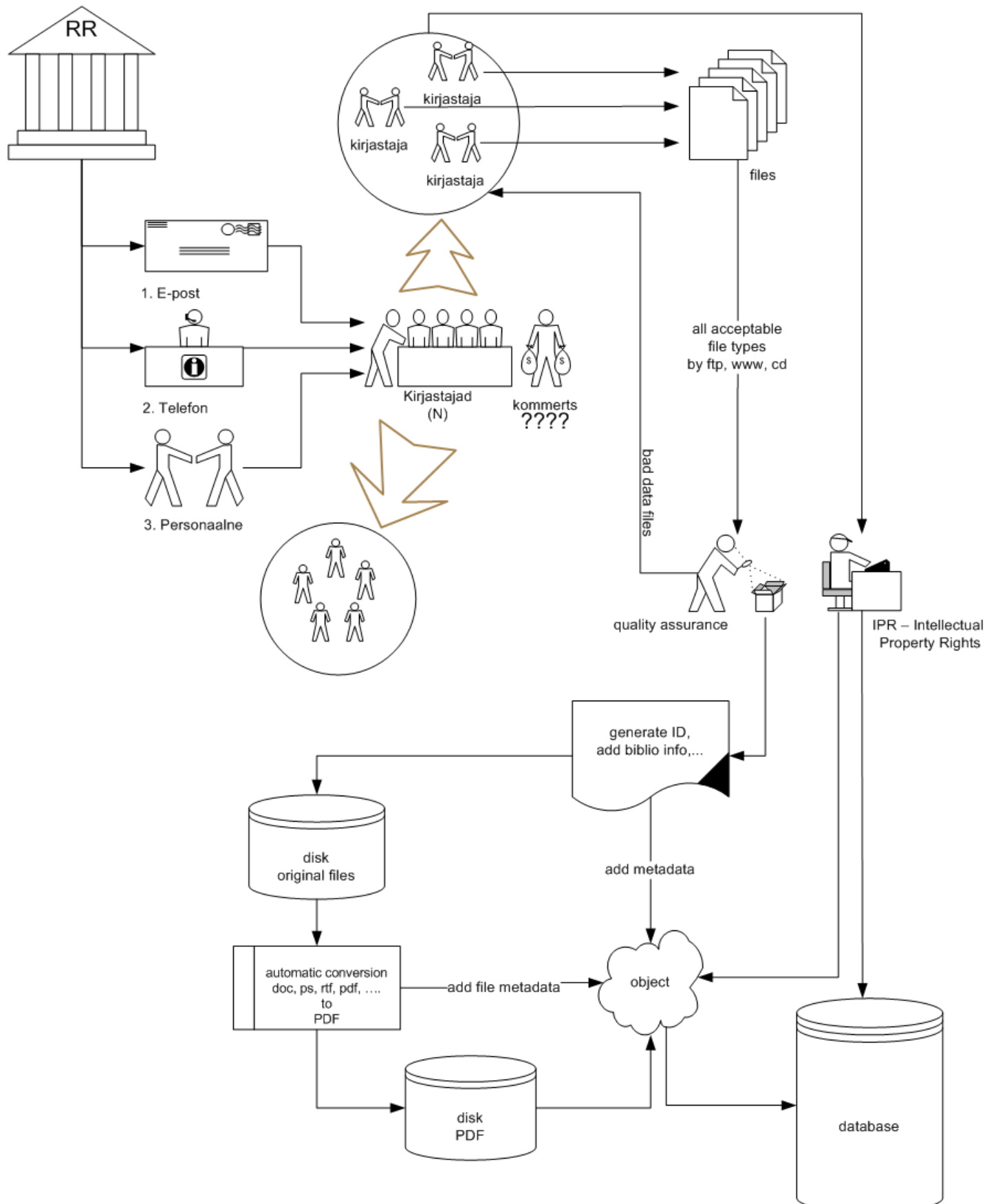
## 2.11 FTP Management



**Figure 5 FTP Management**

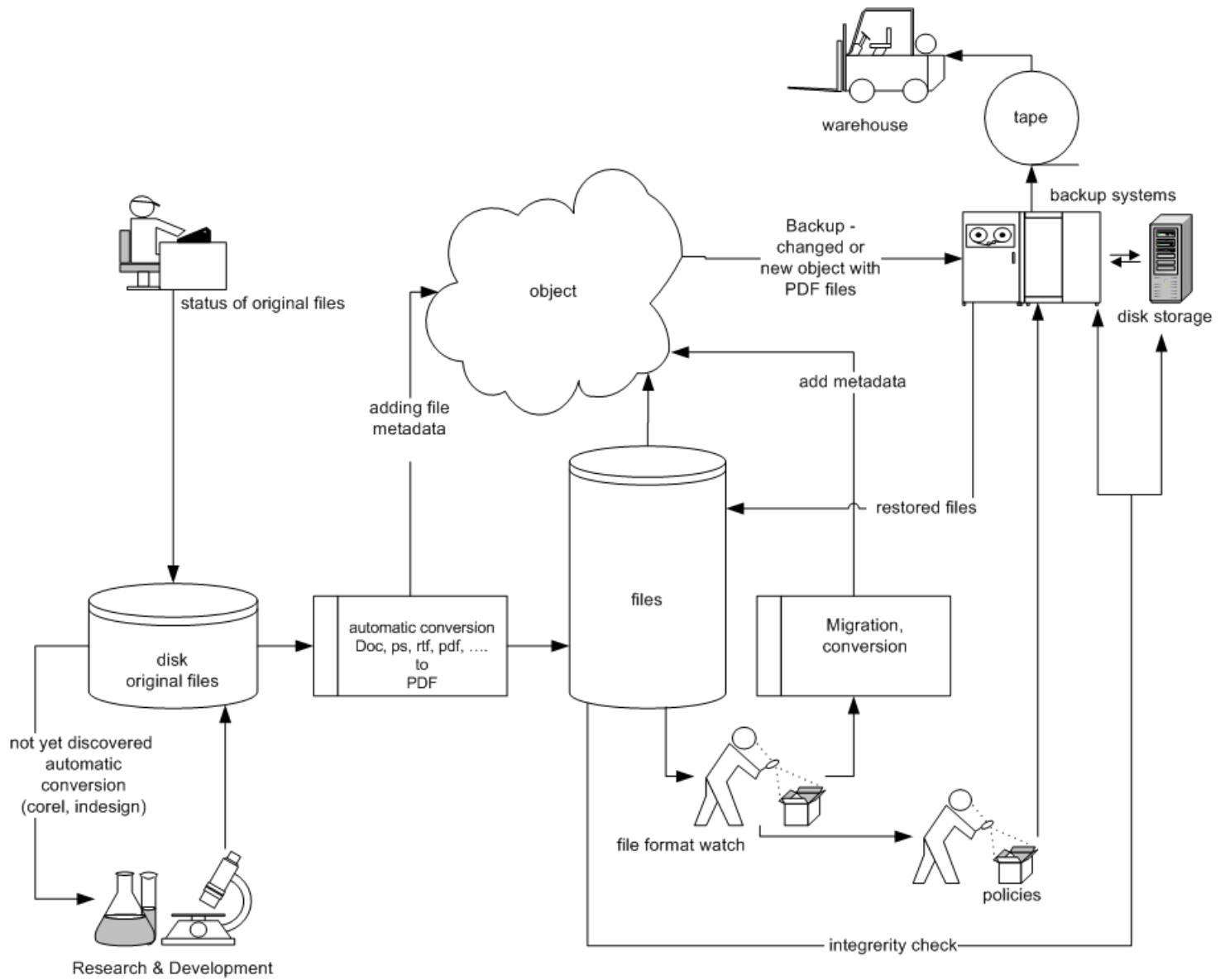
### 3 Workflow models

#### 3.1 Ingest

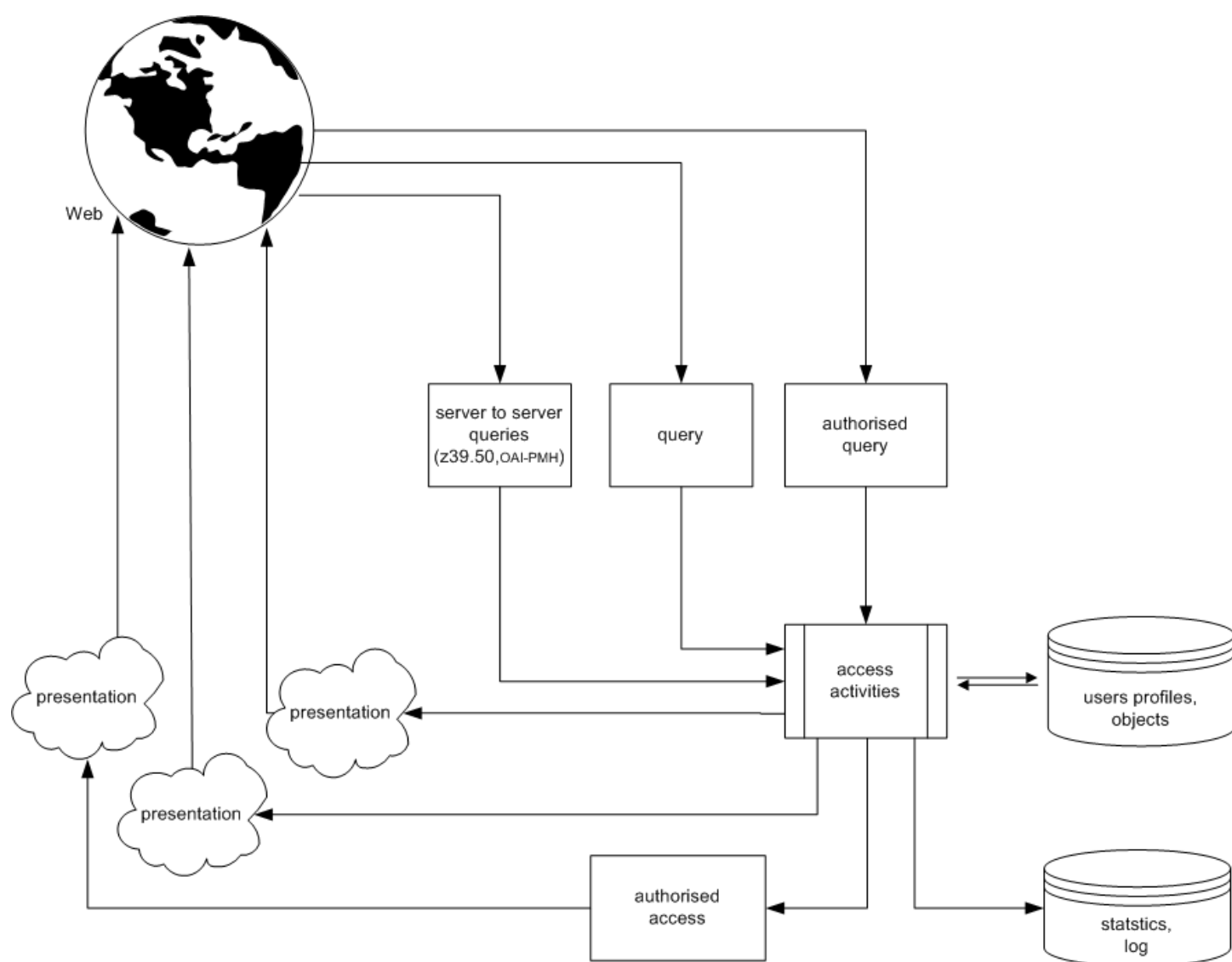




### 3.2 Preservation



### 3.3 Access



## **4 Submission Agreement**

### **DIGAR TEENUSE KASUTUSLEPING**

**Leping sõlmitakse Eesti Rahvusraamatukogu (Arhiivipidaja) ja originaalfailide loovutaja (Deposiitor) vahel.**

#### **Lepingu eesmärk:**

Deposiitori soov tagada oma väljannete pikaajaline säilivus, jätkuv taaskasutus ning kättesaadavus Internetis.

Arhiivipidaja soov pakkuda Depositorile pikaajalise arhiveerimise teenust, tagada arhiveeritud väljaande autentsus ja terviklikkus ning võimaldada arhiveeritud väljaannetele juurdepääs elektronkataloogi ESTER ja Eesti Rahvusraamatukogu elektrooniliste andmebaaside kaudu Internetis.

#### **1. Mõisted**

1.1. Lepingus kasutatakse mõisteid alljärgnevas tähenduses:

- 1.1.1. Arhiivfail – originaalfailist teisendatud fail mis kuulub pikaajalisele arhiveerimisele;
- 1.1.2. Arhiivipidaja - Eesti Rahvusraamatukogu;
- 1.1.3. Depositor – originaalfailide loovutaja;
- 1.1.4. Elektrooniline andmebaas – RRI poolt koostatud bibliograafiline andmebaas, mille kaudu jõuab kasutaja infootsingu sooritamisel kasutusfailini;
- 1.1.5. ESTER – Eesti raamatukogude elektronkataloog;
- 1.1.6. DIGAR – Rahvusraamatukogu digitaalne arhiiv;
- 1.1.7. Kasutusfail - arhiivfailist teisendatud lugejafail;
- 1.1.8. Kasutuspiirang – arhiveeritavale väljaandele või selle osadele vaba juurdepääsu tõkestamine Internetis;
- 1.1.9. Originaalfail – digitaalne fail, mille deposiitor lähetab arhiivi;
- 1.1.10. Võrguväljaanne - identifitseeritav informatsiooni kogum, mis on tehtud üldsusele kättesaadavaks Internetis;
- 1.1.11. Väljaanne – Depositori poolt publitseeritud teos.

#### **2. Lepingu objekt**

- 2.1. DIGAR teenuse kasutuslepingu objektiks on väljaande koostamise käigus loodud trükieelsed failid ning võrguväljaanded, mida arhiiv käsitleb kui originaalfaile.
- 2.2. Depositor annab lepingus sätestatud tingimustel Arhiivipidajale tasuta loa väljaande originaalfailide reprodutseerimiseks arhiiv- ja kasutusfailideks ning saadud koopiate edasiseks kasutamiseks.
- 2.3. Arhiivfailid on pikaajalise säilitamise objektiks. Juurdepääsuõigus arhiivfailidele on Depositoril ja arhiivi tehnilisel personalil.
- 2.4. Kasutusfailid tehakse kättesaadavaks elektronkataloogi ESTER ja elektrooniliste andmebaaside kaudu juhul kui Depositor pole kohaldanud neile kasutuspiiranguid

- 2.5. Kasutuspiirangu kohaldamise vajadusest mitte teatamise korral tehakse väljaande kasutusfailid üldsusele kättesaadavaks Interneti vahendusel. Kasutuspiirangud kehtivad ja on Arhiivipidajale kohustuslikud ainult elektroonilisel levitamisel üle Interneti. Rahvusraamatukogu kohtvõrgus töötavatele lugejaarvutitele Deposiiitori poolt kehtestatud kasutuspiirangud ei laiene.
- 2.6. Deposiiitori poolt Raamatukogule antav luba väljaande kasutamiseks on oma sisult lihtlitsents.
- 2.7. Deposiiitor kinnitab, et tal on kõik õigused käesolevat lepingut sõlmida ja väljaande reprodutseerimiseks ja kättesaadavaks tegemiseks lihtlitsents anda.
- 2.8. Juhul kui Arhiivipidaja vastu esitatakse nõue DIGARi kaudu edastatud teoste autoriõiguste või muu õigushüve rikkumise tõttu vastutab Arhiivile tekkinud kahjude eest Deposiiitor.

### **3. Deposiiitori õigused ja kohustused**

#### **3.1. Deposiiitoril on õigus:**

- 3.1.1. kasutada Rahvusraamatukogu digitaalse arhiivi DIGAR teenust tasuta teenusena.
- 3.1.2. saada tagasi arhiivi saadetud originaalfailid nende esialgsel kujul ja/või arhiivfaile, mida on tehnoloogia arengust tuleneva informatsiooni säilimise ja jätkuva kasutatavuse huvides arhiivis töödeldud ja muudetud.

#### **3.2. Deposiiitor kohustub:**

- 3.2.1. saatma uue väljaande publitseerimise eel arhiveerimisele kuuluva väljaande originaalfailid arhiivi vastavalt Arhiivipidaja poolt kehtestatud ja DIGARi koduleheküljel avaldatud originaalfailide üleandmise korrale või teavitama arhiivi uue väljaande täpsest asukohast oma serveris või veebilehel.
- 3.2.2. vastutama arhiivi üleantavate materjalide nõuetekohase kirjeldamise eest vastavalt Arhiivipidaja poolt kehtestatud ja DIGARi koduleheküljel avaldatud korrale.
- 3.2.3. teavitama Arhiivipidajat väljaande või selle teatud osadele kohaldatavad kasutuspiirangutest autorite või autoriõiguste omajate poolt.
- 3.2.4. viivitamatult teavitama Arhiivipidajat väljaandega seotud õiguslikest vaidlustest ja juhtudest mil ilmneb, et Deposiiitoril ei olnud õiguslikku alust lubadaväljaannete reprodutseerimine ja üldsusele kättesaadavaks tegemine DIGAR kaudu.

### **4. Arhiivipidaja õigused ja kohustused**

#### **4.1. Arhiivipidajal on õigus:**

- 4.1.1. salvestada, reprodutseerida, säilitada ja kättesaadavaks teha Deposiiitori poolt saadetud või Deposiiitori poolse teavitamise järgselt viimase serverist kopeeritud võrguväljaandeid ja neis sisalduvaid autoriõigusega kaitstavaid teoseid.
- 4.1.2. muuta arhiivfailide vorminguid ja tehnilisi parameetreid, arhiivfailides talletatud informatsiooni pikaajalise ja turvalise säilitamise eesmärgil.
- 4.1.3. lõpetada Deposiiitori poolt edastatud väljaannete kättesaadavaks tegemine juhul kui neis sisalduvate teoste või muu materjali osas

käivad õiguslikud vaidlused või kui teosed või muu materjal on vastuolus heade tavadega.

- 4.1.4. arhiveeritud väljaande kasutamisest või selle sisust tulenevate kolmandate isikute nõuded Arhiivipidajale pöörata Depositiitori vastu juhul kui Depositiitor on ilma õigusliku aluseta või pahatahtlikult andnud väljaanded arhiveerimisele.
  - 4.1.5. säilitada koopia väljaande arhiivfailist ka juhul kui Depositiitor nõuab hoiuleantud väljaande tagastamist või kui väljaanne on kohtuotsuse või kokkuleppel antud kolmandatele isikutele.
  - 4.1.6. Kui kolmas isik taotleb omandi või autoriõiguste tunnustamist hoiuleantud väljaandele ning esitab Arhiivipidaja vastu väljaande väljaandmise või kasutamise lõpetamise nõude, ei või Arhiivipidaja väljaannet temale ilma Depositiitori nõusolekuta välja anda või kasutamist lõpetada. Arhiivipidaja võib väljaande tagastada Depositiitorile või lõpetada väljaande kättesaadavaks tegemise DIGAR kaudu. Arhiivipidaja võib keelduda hoiuleantud failide Depositiitorile välja andmast, kui väljaanne on arestitud või kui Depositiitori vastu on esitatud hoiuleantud väljaande omandi või autoriõiguste tunnustamise hagi.
  - 4.1.7. Arhiivipidaja ei vastuta arhiivfailide kaotsimineku, hävimise, kahjustumise või mitteõigeaegse tagastamise eest, kui ta on arhiivfailide hoidmisel ilmutanud samasugust hoolt, nagu ta rakendab oma arhiivfailide hoidmisel.
- 4.2. Arhiivipidaja kohustub:
- 4.2.1. looma tarkvaralahendused arhiivfailide ning nende juurde kuuluvate bibliograafiliste, tehniliste ja administratiivsete metaandmete arhiveerimiseks digitaalarhiivis.
  - 4.2.2. arhiveerima Depositiitori poolt lähetatud väljaande hiljemalt 7 päeva jooksul pärast deposiitori poolt lähetatud originaalfaili kirjeldava informatsiooni edastamist Arhiivipidajale.
  - 4.2.3. teavitama Depositiitorit väljaande arhiivfailide jõudmisest pikaajalisele arhiveerimisele.
  - 4.2.4. teostama pidevat süsteemset seiret arhiivfailide seisundi üle, et tagada arhiveeritud informatsiooni pikaajaline säilimine.
  - 4.2.5. jälgima ja arvestama Depositiitori poolt kehtestatud kasutuspiiranguid kasutusfailide kättesaadavaks tegemisel elektronkataloogi ESTER ja elektrooniliste andmebaaside vahendusel.
  - 4.2.6. teavitama lugejaid deposiitori poolt kohaldatud kasutuspiirangute olemasolust teatud väljaannetele või nende osadele.
  - 4.2.7. Depositiitorile viivitamata teatama arhiivifaili valduse kaotusest, Arhiivi vastu väljaande väljaandmise nõude esitamisest või hoiuleantud väljaandele kolmanda isiku õigusest teadasaamisest.
  - 4.2.8. tagastama arhiivfailid Depositiitorile juhul kui kohtuotsuse või kolmandate isikute kokkuleppest selgub, et Depositiitoril ei olnud õigust väljaande või selles sisalduvate teoste või muu materjali kasutamiseks õiguslikku alust.

**5. Lepingu sõlmimine.**

- 5.1. Leping sõlmitakse kirjalikku taasesitamist võimaldavas vormis. Deposiitori soovil võidakse leping sõlmida lihtkirjalikus vormis.
- 5.2. Leping sõlmitakse tähtajatult.
- 5.3. Lepingu muutmine ja lõpetamine toimub poolte kokkuleppel ning samas vormis, millega sõlmiti leping.
- 5.4. Igakordne originaalfailide edastamine digitaalse arhiivi DIGAR kaudu loetakse lepingu lisaks ning sellele kohaldatakse käesoleva lepingu tingimusi.
- 5.5. Lepingut saab sõlmida Deposiitori seadusjärgne esindaja või põhikirjaga selleks volituse saanud esindaja.
- 5.6. Deposiitor määrab omapoolse kontaktisiku suhetes arhiiviga. Kontaktisiku muutusest teatab Deposiitor.
- 5.7. Lepingu sõlmimiseks loetakse hetke, millal Deposiitori saadetud kirjalikku taasesitamist võimaldas vormis leping jõuab Arhiivipidajani. Lihtkirjaliku lepingu sõlmimise hetkeks loetakse lepingu allkirjastamist Arhiivipidaja poolt.

**6. Poolte andmed:****6.1. Rahvusraamatukogu (Arhiivipidaja):**

Nimi: Tiiu Valm  
Amet: peadirektor  
Struktuuriüksus: juhtkond  
Asutus: Eesti Rahvusraamatukogu  
Aadress: Tõnismägi 2, 15189 Tallinn  
Telefon: 630 7611  
e-mail: nlib@nlib.ee  
kuupäev:

**6.2. Deposiitor:**

Nimi:  
Amet:  
Struktuuriüksus:  
Asutus:  
Aadress:  
Telefon:  
e-mail:  
kuupäev:

NÕUSTUN
---------

## **5 Käsiraamat väljaandjale** (Handbook for publishers)

### **5.1 Mida pakub DIGitaalne ARhiiv DIGAR.**

(Introduction to Services provided by DIGital ARchive DIGAR)

**DIGAR on Eesti Rahvusraamatukogus loodud DIGitaalne ARhiiv, mis kannab hoolt Eestis välja antud trükiste elektrooniliste koopiade säilimise ja taaskasutuse eest.**

Tänapäeval luuakse nii trükis kui ka võrgus avaldatavad väljaanded algselt elektroonilisel kujul. Paraku on järjest kiireneva infotehnoloogilise arengu juures digitaalse informatsiooni eluiga lühike ja säilitamine ressursimahukas. Kirjastajatel puuduvad sageli võimalused trükiste eeltööna loodud digitaalsete failide nõuetekohaseks haldamiseks ja hoidmiseks. Seetõttu on suur osa neist pärast trükiväljaande ilmumist kas kustutatud, või tehnoloogia arenedes hävinud. Ometi omavad väljaande eeltööna loodud failid nende hõlpsama levitatavuse ja korduvkasutuse võimaluste poolest kohati suuremat tähtsust kui trükiväljaanne ise.

Eesti Rahvusraamatukogu püüab digitaalse arhiiviga DIGAR luua tingimused väljaandjate saadetud trükielelsete failide ja võrguväljaannete turvaliseks säilimiseks. Arhiveeritud väljaannetele luuakse juurdepääs elektronkataloogi ja bibliograafiliste andmebaaside kaudu, arvestades sealjuures väljaandja kohaldatud kasutuspiiranguid. Esimeses etapis keskendume avaliku sektori väljaannetele. Digitaalse arhiivi loomist Eesti Rahvusraamatukogus toetavad Eesti Vabariigi Kultuuriministeerium ja Euroopa Komisjon programmi *e-Content* projekti reUSE raames.

#### **Huvigrupid**

**Väljaandjale** tagab pakutav arhiveerimisteenus failide säilimise ja levitamise. Samuti saab arhiivi lähetatud originaalfaile kasutada kordustrükkide ettevalmistamisel.

**Rahvusraamatukogule** annab arhiiv võimaluse arendada infoteenuseid ning muuta veebipõhised bibliograafilised andmebaasid täistekstilisteks. Sellega tagame lugejaile kiirema teeninduse ning raamatukogu trükiste parema säilimise.

**Avalikkusele** pakub loodav arhiiv senisest kiiremat ja asukohast sõltumatut veebipõhist juurdepääsu arhiveeritud ja vabakasutuses olevatele elektroonilistele väljaannetele, mis on vähese trükiarvu ja hajutatud leviku tõttu sageli raskesti kättesaadavad. DIGAR lihtsustab ka piiratud liikumisvõimalustega inimeste juurdepääsu olulistele tekstidele.

## 5.2 Mida kogume (What to Deposit. Deposit Formats)

Arhiiv lähtub digitaalse kogu komplekteerimisel põhimõttest, et kogutav ja arhiveeritav aine peab olema avaldatud (või kavatsetakse avaldada) trükisena või võrguväljaandena ning selle pikaajaliseks arhiveerimiseks ja levitamiseks on saadud luba teaviku väljaandjalt. DIGAR on valmis koguma, säilitama ja kättesaadavaks tegema avalikus ja kolmandas sektoris kirjastatud väljaannete digitaalseid faile. Erasektori kirjastajate huvi korral arhiveerime ka nende väljaandeid.

Arhiiv võtab vastu trükiväljaannete digitaalseid originaale ning võrguväljaannete koopiaid **PDF (soovituslik), SGML, XML, HTML, ASCII, TIFF, GIF, JPEG, DOC** vormingutes.

Arhiiv tagab nimetatud vormingus väljaannete pikaajalise säilimise ja taaskasutuse. Teistes vormingutes originaalfailid (kujundusfailid jms.) kuuluvad säilitamisele üksnes muutmata kujul ning nende pikaajalise säilimise ja hilisema taaskasutuse eest arhiiv ei vastuta. Küll saab väljaandja neid kasutada kuni tarkvaralahendused seda võimaldavad.

Arhiivis salvestame väljaandja poolt lähetatud faili originaalkujul. Originaalfailid teisendame pikaajalise säilitamise tingimustele vastavateks arhiivfailideks, mille seisundit tehnoloogia arenedes jälgime ja vajadusel uuendame. Veebis avaldame arhiivfailidest teisendatud madalaresolutsioonilised kasutusfailid.

Et vältida digitaalse arhiivi muutumist kirjastajate tasuta failiserveriks, kogub ja säilitab arhiiv üksnes väljaandeid, mis tehakse lugejaile kättesaadavaks vastavalt väljaandja kehtestatud kasutuspiirangutele. Juurdepääs kasutusfailidele leiab aset elektronkataloogi ESTER ja Eesti Rahvusraamatukogu elektrooniliste andmebaaside vahendusel, mis hõlbustab muidu üksnes trükistena ilmunud teavikute levitamist avaliku elektroonilise teenuse kaudu.

Arhiivil on õigus keelduda failidest, mis ei vasta komplekteerimisnõuetele (näiteks pisitrükised jms), teavitades sellest väljaandjat igal konkreetsetel üksikjuhul.



### 5.3 Kuidas liituda (How to Join the Archive)

Autoriõigused ja nendega kaasnevad õigused kehtivad ka digitaalses keskkonnas. Raamatukogu vajab õiguste omaja luba, et saaksime digitaalses vormingus väljaandeid pikaajalise säilivuse ning kättesaadavuse huvides kopeerida, teisendada ja muuta arhiiv- ning kasutusfailideks.

Juhul kui te ei soovi, et failid ja nende sisu oleksid üldsusele kättesaadavad Internetis, siis on teil õigus piirata failide kättesaadavaks tegemist.

**Enne e-lepingu sõlmimist tutvuge väljaandjale koostatud juhenditega. Juhendid selgitavad originaalfailide üleandmise korda ning juhendavad teid failide saatmisel ftp-ga ja arhiivi saadetud failide kirjeldamisel ja kasutuspiirangute määramisel.**

**Lihtsaim viis loa andmiseks on nõustuda DIGARi teenuse kasutuslepinguga (e-leping).**

Kui **nõustute** kõigi lepingus olevate tingimustega:

- täitke elektroonilise lepingu vormi deposiitori andmed.
- saatke e-leping arhiivile klikates lingil **NÕUSTUN**.

Tegemist on lepinguga, mis sõlmitakse kirjalikku taasesitamist võimaldavas vormis.

Lepingut saab sõlmida juriidilise isiku puhul seadusjärgne esindaja või selleks volituse saanud töötaja.

Lepingu sõlmimiseks loetakse hetke, mil deposiitori andmeid sisaldav e-leping jõuab arhiivipidajani, kes saadab teile registreerimisteate ning kasutajanime ja parooli, mis on vajalikud failide saatmiseks, kirjeldamiseks ja kasutuspiirangute kohaldamiseks digitaalses arhiivikeskkonnas.

Kui te **ei nõustu** kõigi lepingu tingimustega, soovite sõlmida lepingut kirjalikus vormis või teil tekkis lepingut lugedes küsimusi, võtke ühendust arhiivipidajaga e-posti aadressil digar@nlib.ee

## 5.4 Originaalfailide üleandmise kord: (Submission procedures)

1. Deposiitor sõlmib DIGAR TEENUSE KASUTUSLEPINGU (e-leping) ja saadab selle Arhiivipidajale (Rahvusraamatukogu)
2. Arhiivipidaja (Rahvusraamatukogu) saadab Deposiitorile registreerimisteate ja kasutajanime ning parooli.
3. Deposiitor saadab väljaande originaalfailid Rahvusraamatukogu ftp-serverisse.
4. Deposiitor kirjeldab ftp-serverisse saadetud failid ja vajadusel kohaldab neile kasutuspiirangud 14 päeva jooksul peale originaalfailide saatmist ftp-serverisse. Kasutuspiirangu kohaldamisest mitte teatamise korral kuulub väljaanne peale arhiveerimist vabale levitamisele Internetis.
5. Rahvusraamatukogu moodustab failidest ja metaandmetest arhiivobjekti, misjärel saadab deposiitorile teate väljaande jõudmisest digitaalsesse arhiivi.

### Submission procedures:

1. Depositor fills the electronic licence-form and sends it to NLE.
2. NLE sends the depositor notification about being registered and provides the ID and password.
3. Depositor uploads original digital files into temporary storage. Each depositor has his own designated directory in ftp-server.
4. Depositor fills metadata fields and applies copyright and access restrictions. Depositor has to apply the restrictions during 14 days after uploading the original files to ftp-server. If depositor hasn't applied any restrictions during these 14 days, the publication will be accessible in Internet without any restrictions.
5. NLE sends notification to depositor about successful submission, after files have been sent to archival storage

## 5.5 Juhendmaterjalid (Guides)

### 5.5.1 Kasutusjuhend failide transportimiseks FTP abil (Guide for Uploading the Files)

Oma failide Rahvusraamatukogu serverisse saatmiseks on teil vaja kasutada FTP-programmi. Seda eeskätt põhjusel, et trükieelsete failide puhul on reeglina tegemist mahult väga suurte failidega ja nende üle veebi saatmine ei pruugi ladusalt õnnestuda.

Juhul kui teie arvutis puudub failide transportimist võimaldav ftp-programm võite selle hõlpsasti alla laadida järgmiselt võrguleheküljelt:

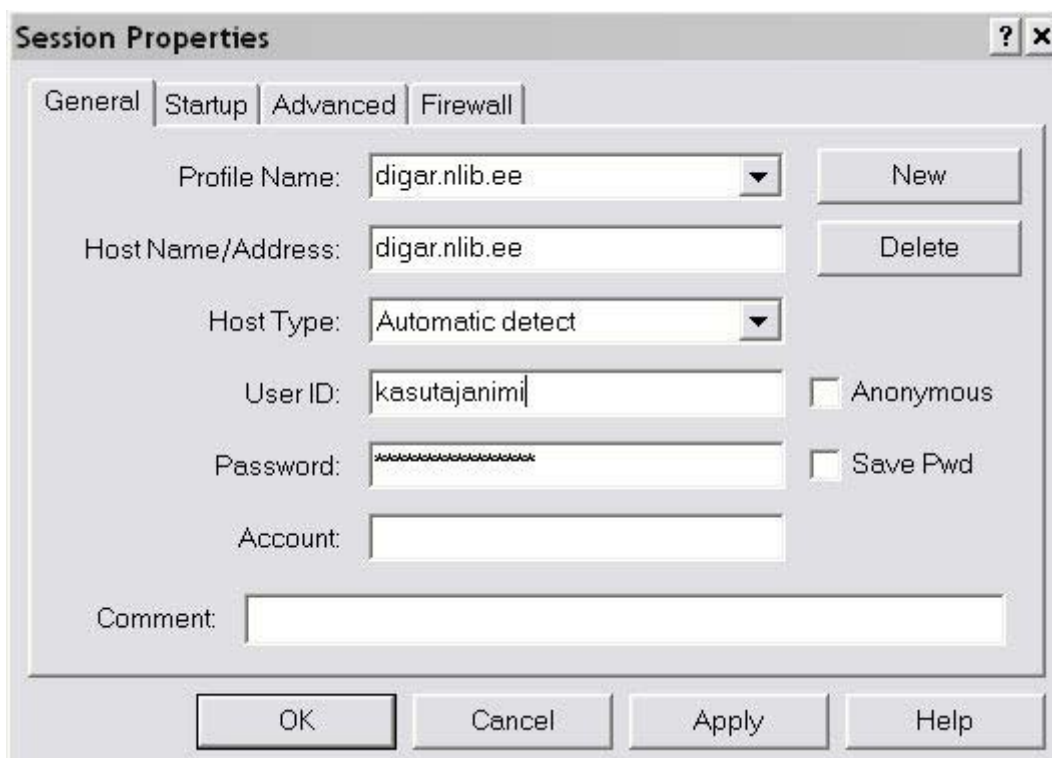
[http://tucows.ibs.ee/networkadministration\\_ftp\\_license.html](http://tucows.ibs.ee/networkadministration_ftp_license.html)

Vabavarana pakutavad programmid on selekteeritavad väljendi Freeware kaudu.

Ühed populaarsemad failide transportimise programmid Windows töökeskkonnas on FTP Commander ja WS\_FTP, mille piirvara versioon LE (Limited Edition) on mõeldud riigiteenistujatele, tudengitele ja kõigile mitteärilistel eesmärkidel tegutsevatele kasutajatele. Viimase programmi näitel oleme illustreerinud ka allpool koostatud ftp kasutamise juhendit.

FTP kasutamise eesmärk on kopeerida enda arvutist failid Rahvusraamatugu serverisse, mille kasutamise õigused on teile antud.

FTP programmi avamisel küsitakse teilt esmalt kasutajanime ja parooli. Need olete te eelnevalt saanud Rahvusraamatukogu arhiivipidajalt nõuetekohase e-lepingu täitmise järgselt.



Host Name: digar.nlib.ee

User ID: - sinna kirjutate kasutajanime, mille saite arhiivilt.

Password: - sinna kirjutate parooli, mille saite arhiivilt.

Ühenduse saamiseks klikake: **OK**

FTP programm võtab seejärel ühenduse ftp-serveriga ja avab paremale poole (Remote Site) teile eraldatud kataloogi Rahvusraamatukogu ftp-serveris. Vasakule poole (Local System) avatakse kataloogid teie kohalikus arvutis.

### Failide oma arvutist kaugarvutisse lähetamiseks toimige järgmiselt:

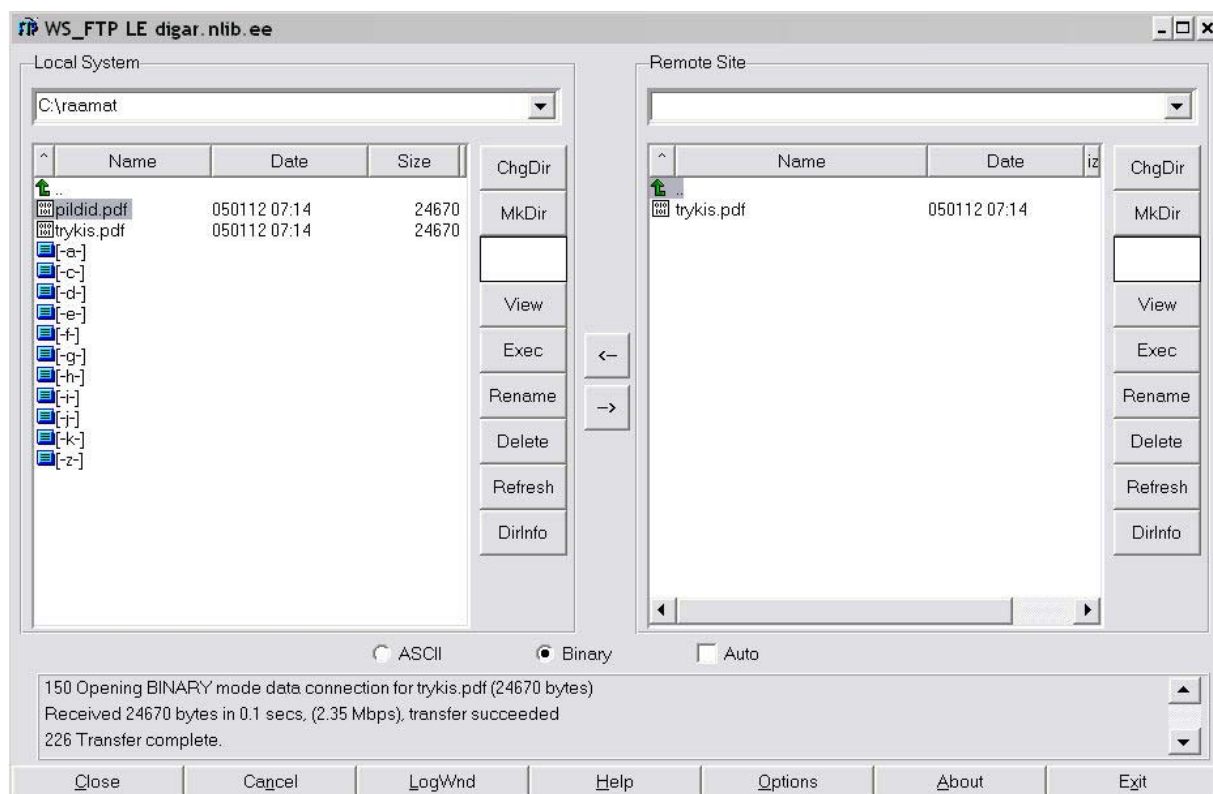
Märgistage kataloog või failid oma arvutis (Local System), mida soovite arhiivi saata. Juhul kui soovite saata mitme erineva väljaande faile üheaegselt, soovitate teil koguda ühte väljaandesse kuuluvad failid eelnevalt kokku ühtsesse kataloogi. See hõlbustab hilisemal kirjeldamisel failide kogumist õige väljaande nimetuse alla.

Kataloogides saab liikuda analoogselt File Manager'ile või Windows Explorer'ile.

Kaks korda üleval vasakus servas rohelisel noolel klikates jõuate kataloogipuus tagasi. Edasi liikumiseks klikake kaks korda kataloogi peal.

→ klikk sellel noolel käivitab vasakus aknas (Local System) märgistatud kataloogi või faili(de) kopeerimise Rahvusraamatukogu ftp-serveri kataloogi, mis teile sisse logimisel avati.

← klikk sellel noolel käivitab paremas aknas (Remote Site) märgistatud kataloogi või faili(de) kopeerimise oma arvutisse tagasi.



Failide kopeerimise järel Rahvusraamatukogu ftp-serverisse veenduge et kõik failid, mida soovisite lähetada, jõudsid parempoolsesse (Remote Site) aknasse kohale.

“Close” – katkestab ühenduse Rahvusraamatukogu ftp-serveriga.

“Exit” – sulgeb ftp programmi teie arvutis.

NB! Järgmiseks minge arhiivi koduleheküljelt ([digar.nlib.ee](http://digar.nlib.ee)) väljaandja logini kaudu arhiivisüsteemi ning kirjeldage teie poolt ftp-ga lähetatud väljaanne ning sellele kohaldatavad kasutuspiirangud.

### 5.5.2 Juhend failide kirjeldamiseks ning kasutuspiirangute määramiseks

(Guide for Filling Descriptive Metadata Form and Applying Access Restrictions)

Väljaandja tagab, et arhiivi saadetud originaalfailidega oleksid kaasas sisu ja õigusi kirjeldavad metaandmed. See teave on oluline, et bibliograafiliste ning tehniliste metaandmete kõrval säiliks andmed ka õiguste kohta.

Faile saate kirjeldada arhiivi kodulehel ([digar.nlib.ee](http://digar.nlib.ee), **VÄLJAANDJA LOGIN**), kasutades e-lepingu sõlmimisel eraldatud kasutajanime ja parooli.



#### FTP-sse laetud failid:

	<b>Kat1</b>	
	failinimi.pdf	Lisa
	failinimi.pdf	Lisa
	failinimi.pdf	Lisa
	<b>Kat2</b>	
	failinimi.pdf	Lisa
	failinimi.pdf	Lisa
	failinimi.pdf	Lisa
	failinimi.pdf	Lisa

#### Väljaande kirjeldamine:

Väljaande pealkiri:	<input type="text"/>	
Kaasväljaandja:	<input type="text"/>	
Autor(id):	<input type="text"/>	<input data-bbox="1236 1310 1284 1355" type="button" value="+"/>
Aasta:	<input type="text"/>	
Copyright:	<input type="text"/>	
<b>Väljaande kasutuspiirang:</b>		
Avalik alates:	<input type="text" value="19.02.2005"/>	<input type="button" value="Vali kalendrist"/>
<input type="button" value="Saada arhiivi"/>		
NB! Enne saatmist lisa failid.		

#### Väljaandega seotud failid:

Failinimi		Märkus faili kohta	Kasutuspiirangud
1.faili.pdf_	Eemalda	<input type="text"/>	Kättesaadav ▾
2.faili.pdf_	Eemalda	<input type="text"/>	Kättesaadav ▾

Akna vasakus servas avanevad FTPsse laetud failid.

**Kirjeldamiseks:**

1. sisestage **väljaande pealkiri, autor(id), ilmunisaasta ning õiguste omanik**. Juhul kui soovite kehtestada väljaandele vabakasutuspiirangut Internetis, valige **kasutuspiirangu** kalendriaknast kuupäev, milleni soovite piirata juurdepääsu Internetis väljaande digitaalsele kasutuskooptiale.

**NB! Kehtestatud kasutuspiirangud ei laiene Rahvusraamatukogu sisevõrgus töötavatele lugejaarvutitele.**

2. Lisage kirjeldatava väljaande failid ükshaaval ja õiges järjestuses FTPsse laetud failide nimistust, klikates lingile **Lisa**. Failide järjestust võib muuta ka pärast lisamist, liigutades neid failirea lõpus olevate noolte abil üles/alla.

Kõik lisatud failid paigutuvad tabelisse **Väljaandega seotud failid**. Olles ekslikult lisanud teise väljaandesse kuuluva faili, eemaldage see klikates lingile **Eemalda** ja see suundub tagasi **FTPsse laetud failide** nimistusse.

3. Kasutuspiiranguid, märkusi ja selgitusi saab edastada ka **iga üksikfaili** kohta eraldi. See võib osutuda vajalikuks, kui väljaande artikkelil või peatükkidel on erinevad autorid ja mõni neist on keelanud oma teoste vaba levitamise Internetis.
4. Olles kirjeldanud digitaalse väljaande ja lisanud selle juurde kuuluvad failid õiges järjestuses, kontrollige sisestatud informatsiooni tõesust ja saatke kirjeldus koos failidega arhiivi (link: **Saada arhiivi**)

Teie failid jõuavad arhiivihaldurini, kes kontrollib saabunud materjalide sisulist terviklikkust, failide tehnilist seisundit, andmete õigsust ning moodustab failidest ja metaandmetest arhiivobjekti.

Arhiivobjekti moodustumise hetkel võtab arhiiv enda hooleks väljaande pikaajalise säilitamise ja teile saadetakse e-postiga vastav teade.

Samuti võetakse teiega e-posti teel ühendust, kui arhiivi saabunud materjal polnud sisult terviklik, tehniliselt vastuvõetav või korrektne.