Copyright © 2011 ESA and other members
of the SCIDIP-ES Consortium, www.scidip-es.eu

SCIDIP-ES
SCIENCE DATA INFRASTRUCTURE FOR PRESERVATION - EARTH SCIENCE

# D21.3 Packaging Toolkit Installation, Deployment and User Manual

| | | |
|---|---|---|
| **Work package** | WP21 Services/Toolkits Development and Adaptation | |
| | | |
| **Task** | T21.10 Packaging Toolkit Development | |
| **Author (s)** | Shirley Crompton, Brian Ritchie | STFC |
| **Author (s)** | Andrea Colapicchioni | ACS |
| **Author (s)** | Holger Brocks | ICT |
| **Author (s)** | Yannis Marketakis | FORTH |
| **Author (s)** | Felix Engel | FTK |
| **Author (s)** | Jinsongdi Yu | JU |
| **Author (s)** | Luigi Briguglio | ENG |
| **Authorized by** | Name Surname | Company |
| **Reviewer** | Name Surname | Company |
| **Doc Id** | | |
| **Dissemination Level** | ~~CONFIDENTIAL~~/PUBLIC | |
| **Issue** | 1.0 | |
| **Date** | 13/02/2014 | |

Copyright © 2011 ESA and other members
of the SCIDIP-ES Consortium, www.scidip-es.eu

**Abstract**:

This document represents the Deployment and User Manual for the Packaging Toolkit v 0.0.2 developed in the frame of SCIDIP-ES project. This document contains all useful information on how to install, configure and use the Packaging component.

## Document Log

| Date | Author | Changes | Version | Status |
|---|---|---|---|---|
| | | | | |
| | | | | |
| **12/02/2014** | Felix Engel, Holger Brocks | Created from D21.4. Update sections 2.4 and 2.6. Added section 2.2 and chapter 3 | 0.1 | Draft |

## TABLE OF CONTENTS

SCIDIP-ES SCIENCE DATA INFRASTRUCTURE FOR PRESERVATION - EARTH SCIENCE ..... 1

1 INTRODUCTION ..... 7

1.1 PURPOSE AND SCOPE ..... 7
1.2 WHO SHOULD READ THIS DOCUMENT ..... 7
1.3 SYSTEM CONTEXT ..... 7
1.4 RELEASE NOTES ..... 7

2 INSTALLATION GUIDE ..... 7

2.1 OVERVIEW ..... 8
2.2 LICENSE INFORMATION AND TERMS OF USE ..... 8
2.3 PREREQUISITES ..... 8
2.3.1 SOFTWARE PREREQUISITES ..... 8
2.3.2 HARDWARE PREREQUISITES ..... 8
2.4 OSS/COTS INSTALLATION ..... 8
2.5 DOWNLOAD INFORMATION ..... 8
2.6 PACKAGING INSTALLATION ..... 8
2.7 CONFIGURATION ..... 9
2.8 UNINSTALLATION ..... 9

3 SOFTWARE DESIGN ..... 9

4 USING SCIDIP-ES PACKAGING ..... 11

4.1 GETTING STARTED ..... 11
4.2 OPERATING ..... 11
4.2.1 CREATING AN AIP ..... 12
4.2.2 CREATE AN AIU ..... 15
4.2.3 CREATE AN AIC ..... 15
4.2.4 MODIFY EXISTING IP ..... 16
4.2.5 BROWSE CREATED OR MODIFIED INFORMATION PACKAGES ..... 17

5 REFERENCE MANUAL ..... 18

5.1 KEYBOARD SHORTCUTS ..... 18
5.2 COMMAND-LINE COMMANDS ..... 18
5.3 PUBLIC APIS ..... 18

6 TROUBLESHOOTING COMMON ISSUES ..... 18

ANNEX A. FIGURES AND TABLES ..... 18

A.1. LIST OF FIGURES ..... 18
A.2. LIST OF TABLES ..... 19

ANNEX B. TERMINOLOGY ..... 20

# 1 Introduction

## 1.1 Purpose and Scope

This document provides an overview of the M30 release of the Packaging Toolkit focusing on its design, installation, maintenance and usage.

## 1.2 Who should read this document

Users who may want to deploy and/or use the Packaging Toolkit.

## 1.3 System Context

Packaging plays a major role during ingest and access of digital object in an OAIS compliant archive. Packaging assembles and represents all constituents required to build an adequate information package (IP). Being part of ingestion process packaging has to interact at least with Persistent Identifier Service, Registry and the Storage Service. According to the OAIS specification an information package could have different manifestations. An archival information package (AIP) consist of the content information (CI) and preservation description information (PDI), an archival information unit (AIU) is an AIP that is not restricted to describe one CI, but multiple. However, an archival information collection (AIC) is an aggregation of multiple AIPs. All information package constituents could be part of the package itself, but could possibly also be stored in a distributed environment. The creation and modification of AIPs, AIUs and AICs in a local or distributed context is part of the SCIDP-ES Packaging implementation.

## 1.4 Release Notes

Packaging Version 0.0.2 is able to pack digital objects using OAI-ORE[1] and XFDU/SAFE[2] as packaging formats. All different information package types (AIP, AIU, AIC) could be serialized and de-serialized. An IP serialization could be transformed in one of the other supported serialization formats. The packaging implementation integrates the SCIDP-ES common framework component and uses it for the communication with the various representation information (RepInfo) registry installations at project partner's sites. The internationalization of the current graphical user interface is supported in German and English. The implementation uses the SCIDIP-ES information model as basis for the implementation.

# 2 Installation Guide

Packaging is implemented as a JavaEE 1.6 service to be deployed in tomcat application server version 7. Packaging is a Maven project with a *.war target, that is used to be deployed on any application server. Since GUI and packaging API are assembled in the same project, it is only required to deploy one *.war file. Since the encoding of the source file is set to UTF-8 the war file should be platform independent.

[1] Open Archival Initiative – Object Reuse and Exchange - http://www.openarchives.org/ore/
[2] XFDU/SAFE product - http://earth.esa.int/SAFE/models.html

## 2.1 Overview

## 2.2 License Information and Terms of Use

The SCIDIP-ES Packaging Toolkit is licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance with the License. A copy of the License could be obtained at: http://www.apache.org/licenses/LICENSE-2.0 Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Note: the RichFaces[3] project is used to implement the graphical user interface of the Packaging software. The RichFaces project is licensed under the GNU LESSER GENERAL PUBLIC LICENSE[4].

## 2.3 Prerequisites

- Tomcat version 7, CATALINA_HOME environment variable has to be set.
- Http server (e.g. Apache http)

### 2.3.1 Software prerequisites

- Java EE Development Kit 6 (for installing and running Tomcat)
- Increase Java Opts to: -XX:MaxPermSize=256m -Xms50m -Xmx512m

### 2.3.2 Hardware prerequisites

None

## 2.4 OSS/COTS Installation

*None*

## 2.5 Download information

The recent stable source code could be accessed from SVN at *Sourceforge*. The URL to the svn trunk is: svn://svn.code.sf.net/p/digitalpreserve/code/SCIDIP-ES/software/services/Packaging/trunk

The software could as well be accessed through the SCIDIP-ES maven nexus repository at: http://nexus.scidip-es.eu/content/repositories/releases/eu/scidipes/services/PackagingSCIDIP-ES/0.0.2/

## 2.6 Packaging Installation

Copy packaging war file to the tomcat webapps directory

[3] RichFaces - http://www.jboss.org/richfaces
[4] LGPL - https://www.gnu.org/licenses/lgpl.html

## 2.7 Configuration

- Configure project paths in /PackagingSCIDIP-ES/src/webapp/ptconf.xml
  - *<catalinaprojectdir>*
    - E.g: *work\Catalina\localhost\PackagingSCIDIP-ES*
    - Specifiy the path on the server from CATALINA_HOME to Packaging project top level directory
    - Must end with a slash
  - *<resultpage>*
    - E.g.: *result.xhtml*
    - Specify the www site used to provide AIP to download
  - *<resultDir>*
    - E.g.: */var/www/html/*
    - Specify the path to the http server directory used to provide AIP's for download

Some further configurations are required in order to properly apply the third party software that is used for SAFE and XFDU serialization.

- Set the *SAFE_HOME* environment variable to: SAFE_HOME="*TOMCAT_install_dir"*/webapps/PackagingSCIDIP-ES/WEB-INF/lib/
- Add the path to the XFDU/SAFE libraries in Tomcat configuration file ($CATALINA BASE/bin/catalina.sh)
  - For Linux operating system
    - # Add xfdu-core-1-1-patch-5-1.1.5.jar, drb-2-3-release.jar, log4j-1.2.15.jar to classpath
    - CLASSPATH=$CLASSPATH:$CATALINA BASE/webapps/PackagingSCIDIP-ES/WEB-INF/lib/xfdu-core-1-1-patch-5-1.1.5.jar
    - CLASSPATH=$CLASSPATH:$CATALINA BASE/webapps/PackagingSCIDIP-ES/WEB-INF/lib/drb-2-3-release.jar
    - CLASSPATH=$CLASSPATH:$CATALINA BASE/webapps/PackagingSCIDIP-ES/WEB-INF/lib/log4j-1.2.15.jar
  - For Windows operating system
    - set "CLASSPATH=%CLASSPATH%:%CATALINA BASE%/webapps/PackagingSCIDIP-ES/WEB-INF/lib/xfdu-core-1-1-patch-5-1.1.5.jar"
    - set "CLASSPATH=%CLASSPATH%:%CATALINA BASE%/webapps/PackagingSCIDIP-ES/WEB-INF/lib/drb-2-3-release.jar"
    - set "CLASSPATH=%CLASSPATH%:%CATALINA BASE%/webapps/PackagingSCIDIP-ES/WEB-INF/lib/log4j-1.2.15.jar"

## 2.8 Uninstallation

Undeploy war file from tomcat, remove the third party tools and their classpaths as defined in the last section.

# 3 Software Design

The Packaging software is implemented as a *Java Server Faces* (JSF) project. JSF is used for the graphical user interface implementation. Here the user interface interacts with an underlying Java API that supports various operations for AIP creation and modification (cf. Figure 1).

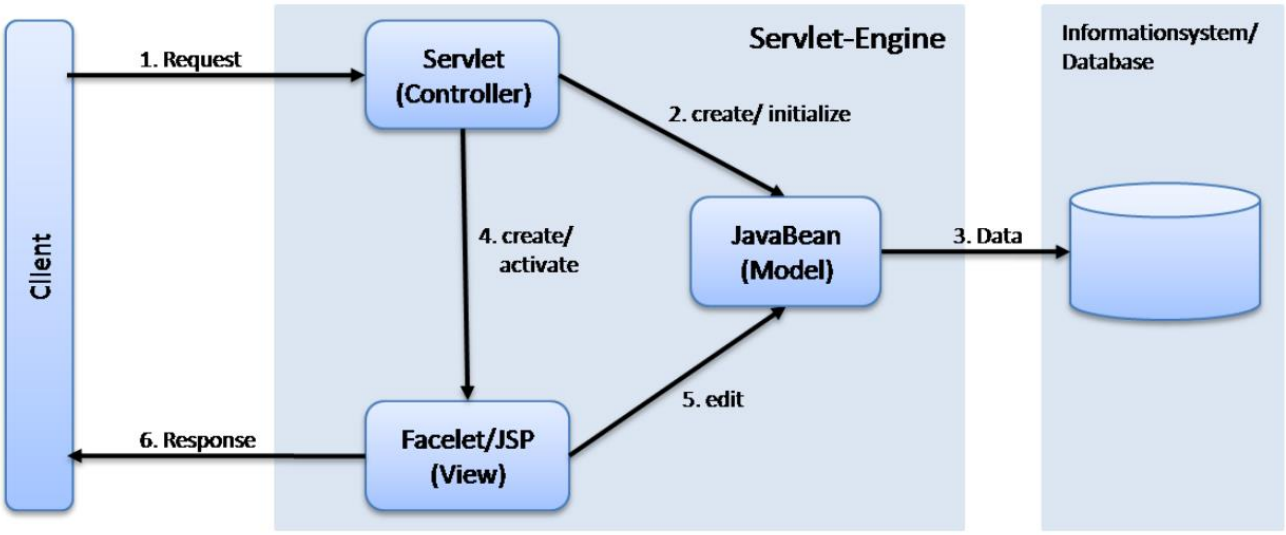


Figure 1: JSF model

The underlying Packaging API (cf. Figure 2) is a scalable, format-independent library for managing OAIS information packages. It comprises factory methods for creating information packages and its internal structure is based on Java Hashmaps.

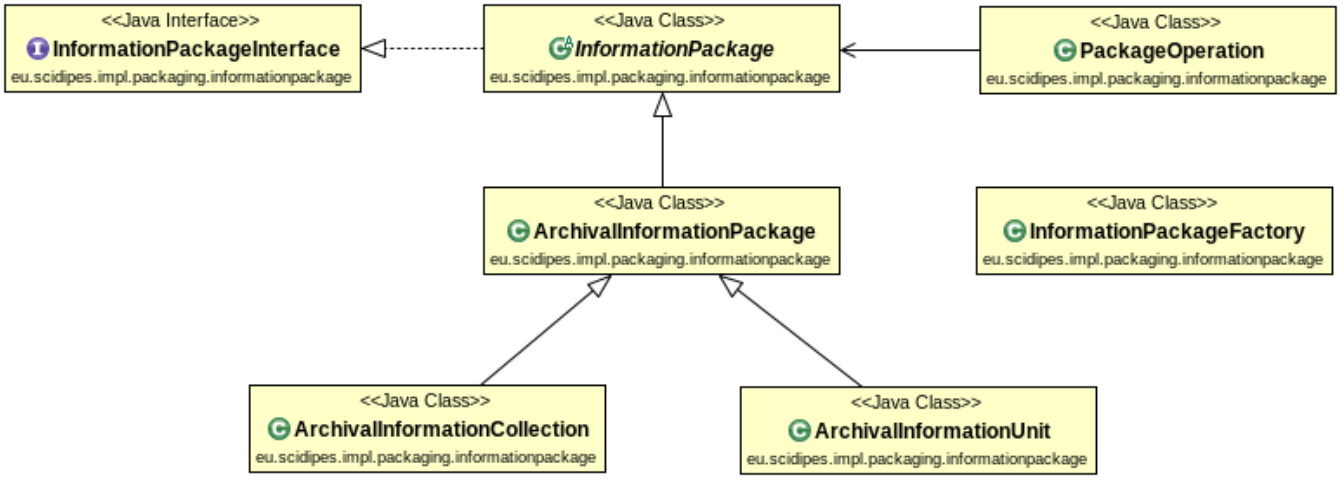


Figure 2: Packaging API

The general flow of AIP creation or modification that is triggered through the user interface is depicted in the UML Sequence diagram in Figure 3. New information packages are created through corresponding factory methods, the constituents of the information package can then be manipulated by add/modify/delete/get methods. RepInfo is added by interacting with the RepInfo Registry. The information package is then serialized (OAI-ORE, SAFE, XFDU) and handed over to the Storage Provider.

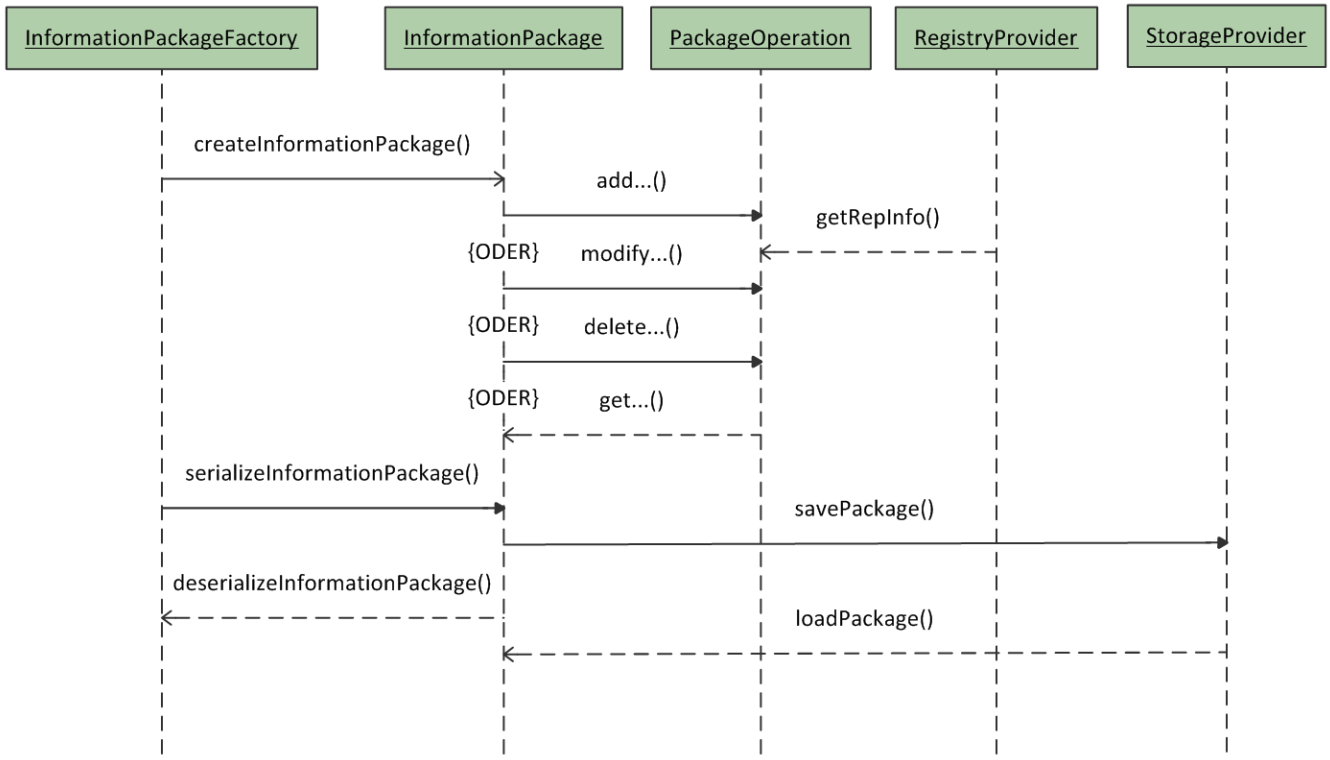


Figure 3: Packaging API for creation and modification of AIPs

# 4 Using SCIDIP-ES Packaging

## *4.1 Getting Started*

Packaging is a web based application, accessible via any current browser. This implementation has been tested with Chrome.

## *4.2 Operating*

The creation and modification of AIP, AIUs and AICs is currently supported by the packaging implementation in a local and a distributed application context.

Figure 4 shows the top menu of the Packaging Toolkit and its main operations. Example usage of these operations is given in the following sub-sections.

Figure 4: Top menu of the application

### 4.2.1 Creating an AIP

In a first step a title for the AIP and the digital object to be preserved has to be specified. After activating the add button the browsed digital object will be uploaded. The AIP references could be added in the following (Figure 6).

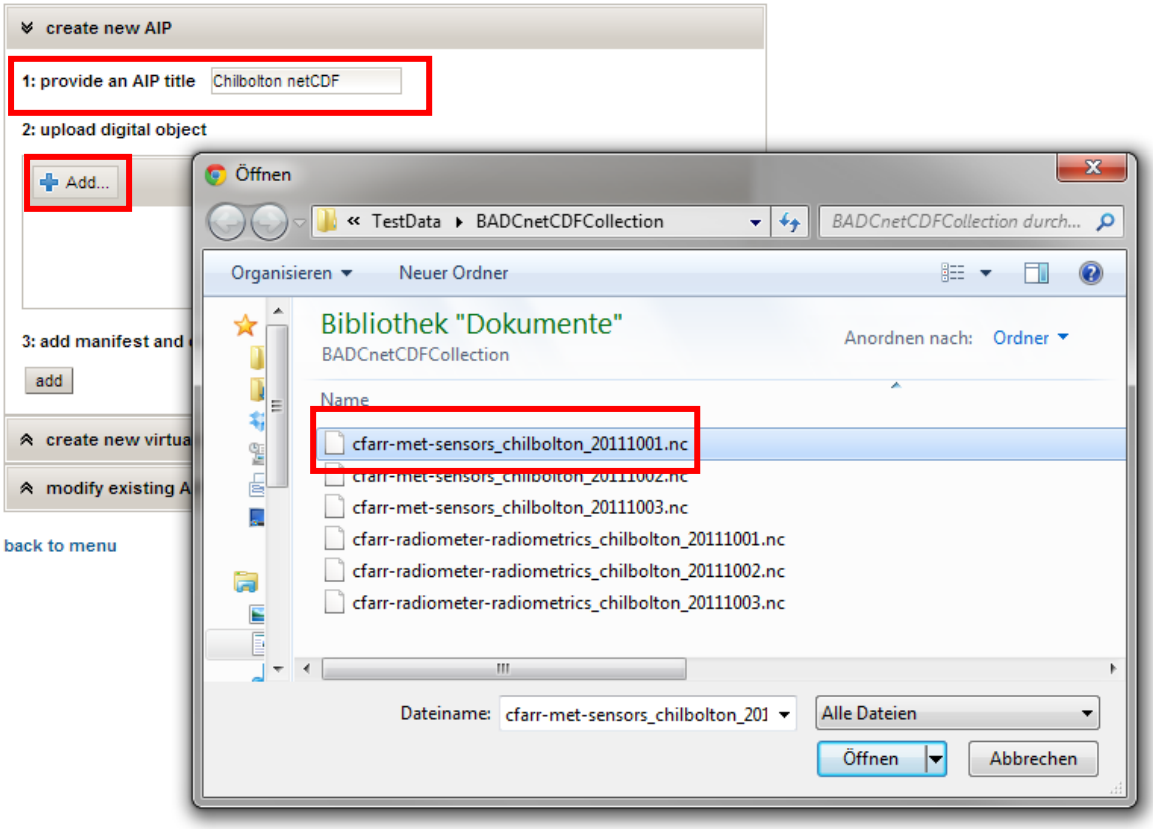


Figure 5: Specify title and digital object for upload

Within this window the AIP constituents could be specified, modified or deleted (Figure 6).

The column on the left hand side is used to select one of the IP constituent; the column in the middle presents the value of an IP constituent and provides a button to add representation information from the project partner registries. The right hand column is used in order to add, modify or delete the value of an IP constituent. Furthermore, the column on the right hand side is used in order to specify the serialization format and to trigger the actual packaging process.

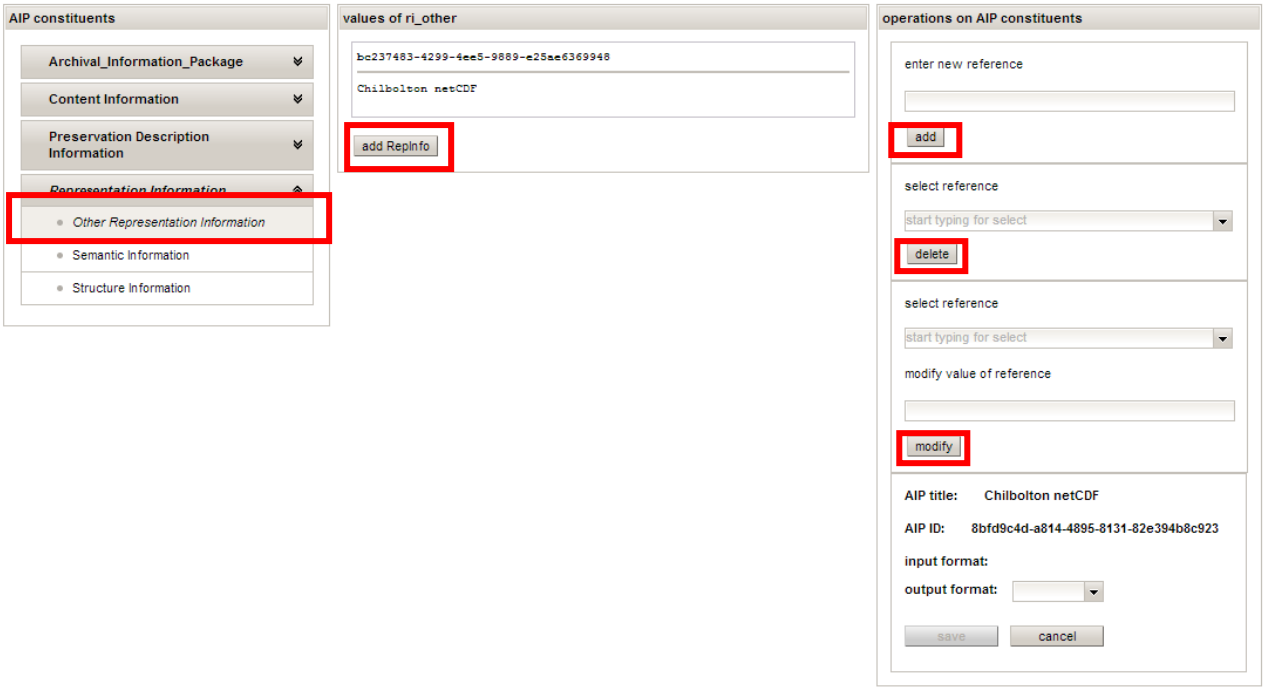


**Figure 6: Specification of IP constituents**

To add representation information a further window is opened that enables the selection of one of the project partner's registries, with ability to specification of provided categories. All representation information objects that are part of the selected registry and category are displayed and could be selected (Figure 7).

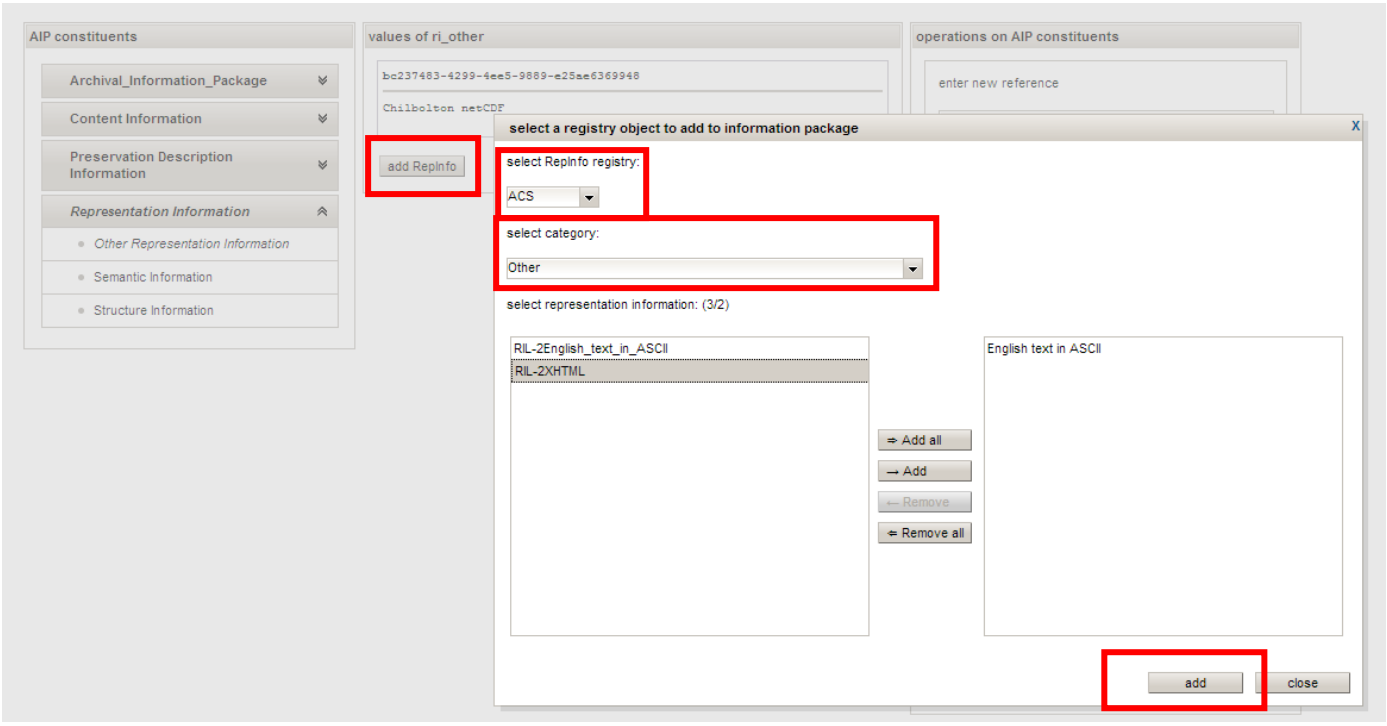


**Figure 7 : Add representation information to an information package**

The specification of information package constituents is followed by the selection of an output format. The activation of the save button triggers the serialization process (Figure 8).

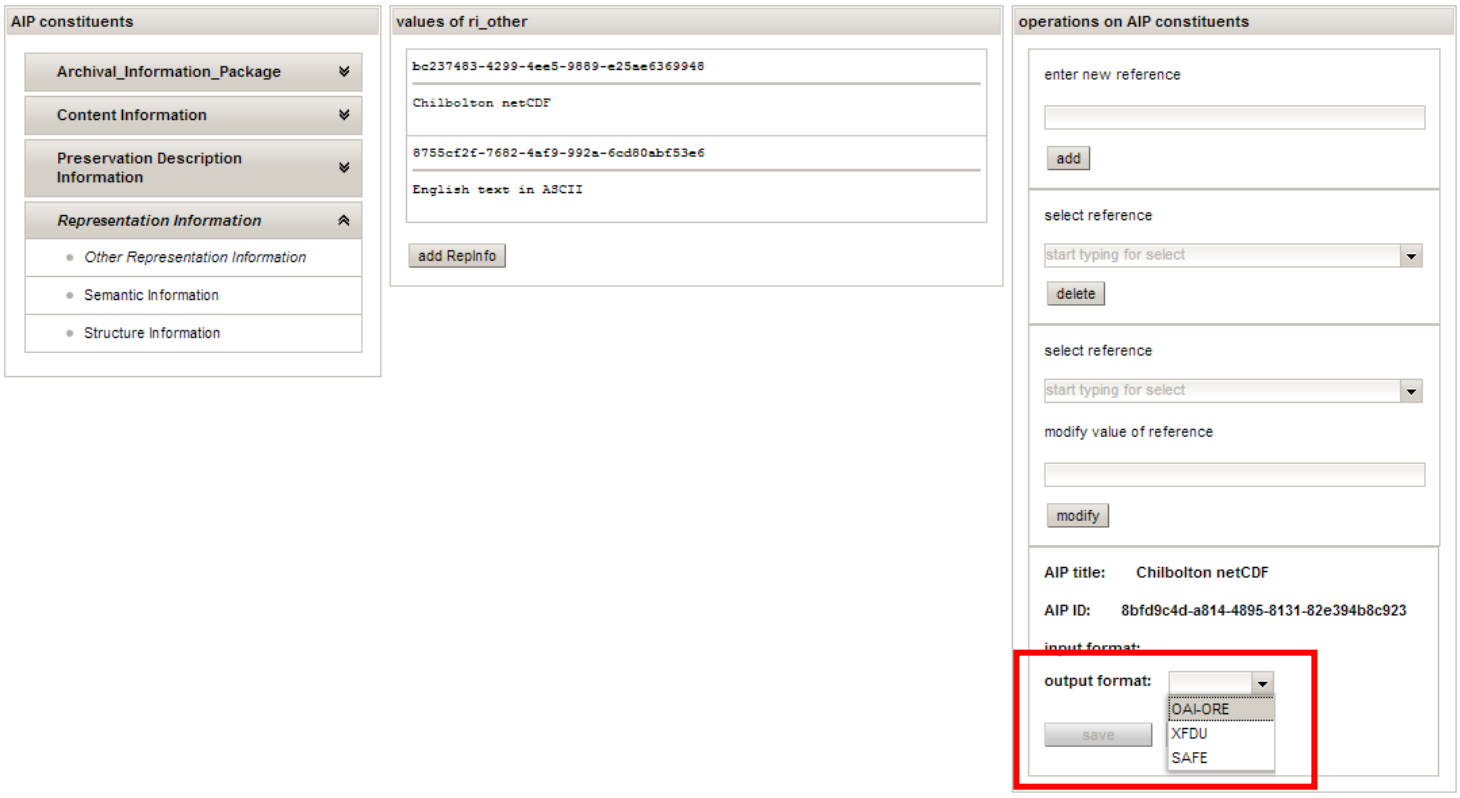


**Figure 8: Creating aSAFE information package**

### 4.2.2 Create an AIU

An AIU is an AIP that encompasses more than one digital object. The only difference to the creation of an AIP is that several objects could be selected for packaging (Figure 9).

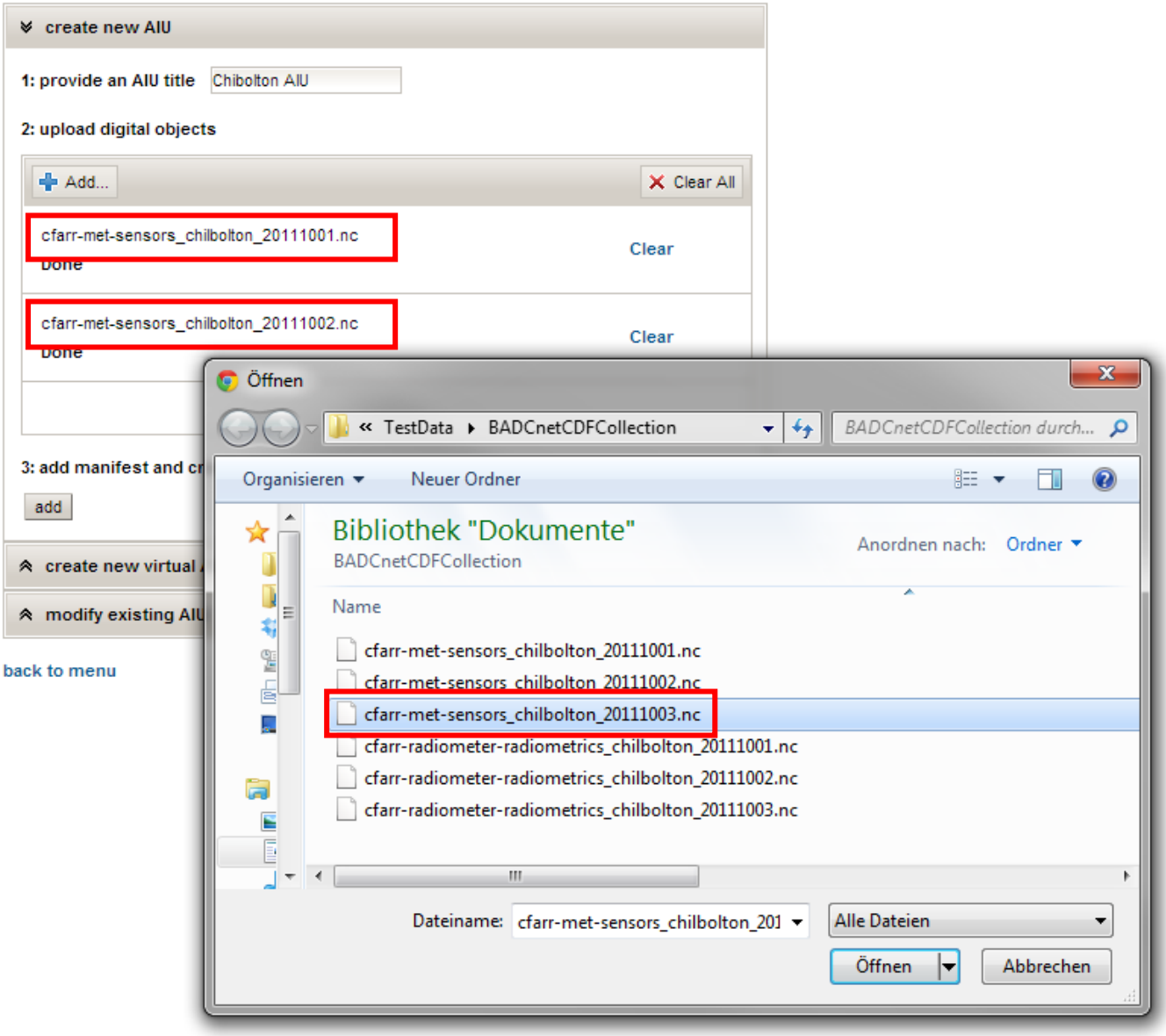


Figure 9: Creating an AIU

### 4.2.3 Create an AIC

An AIC is an information package that aggregates several AIPs which share some common data. The expected input is AIPs or AIUs (Figure 10).

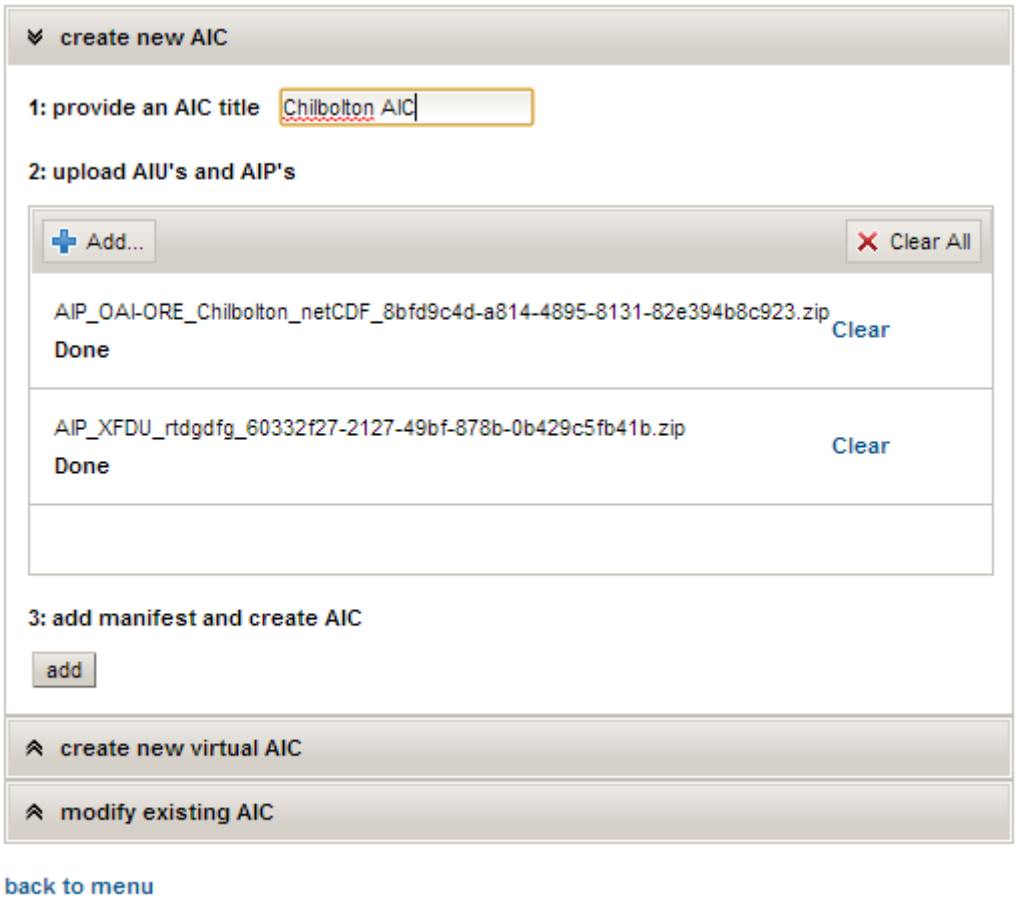


Figure 10: AIC creation

### 4.2.4 Modify existing IP

The first step to modify an existing AIP is to select and upload it to the SCIDIP-Es server (Figure 11).

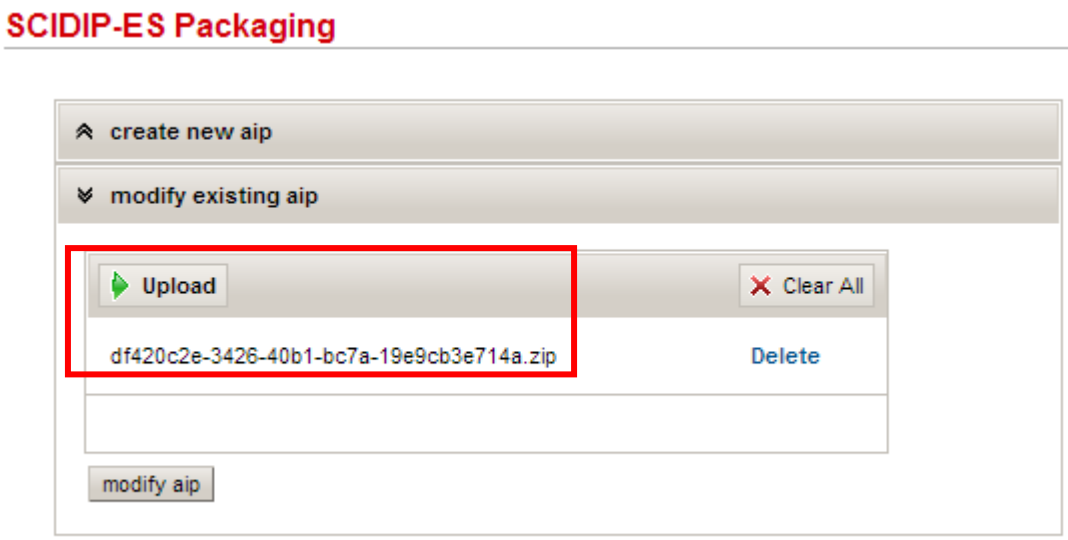


Figure 11: Upload an existing AIP

After uploading the AIP its constituents could be modified by selecting the button 'modify aip'.

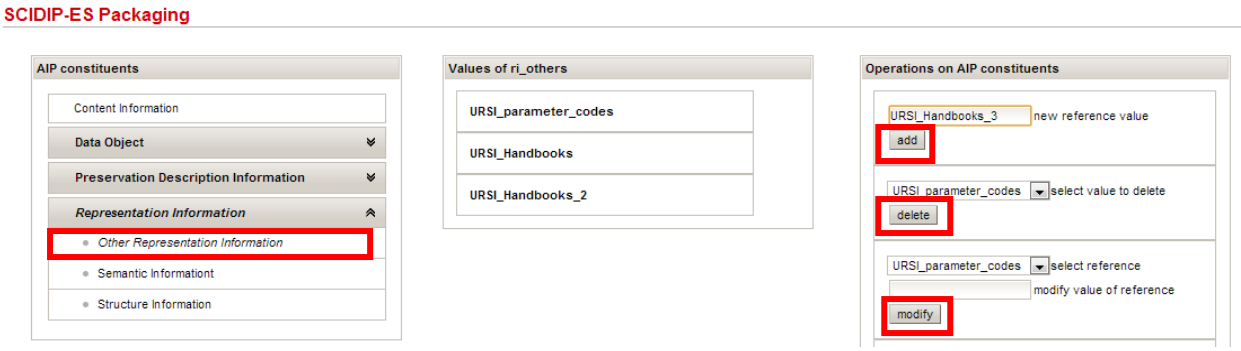


Figure 12: Modification of AIP manifest file

After the upload process the information package manifest file could be browsed and modified (Figure 12). Modification includes the following operations:

- adding new parameters
- delete existing parameters, and
- modify existing parameters.

Having completed AIP modification the AIP could be saved back again in one of the provided formats through activation of the save button.

### 4.2.5 Browse created or modified information packages

The information package browser offers the ability to browse all created and modified information packages (Figure 13). The browser provides functionalities for the download of information packages

from the server, modification of information packages and the removal of an information package from the server.

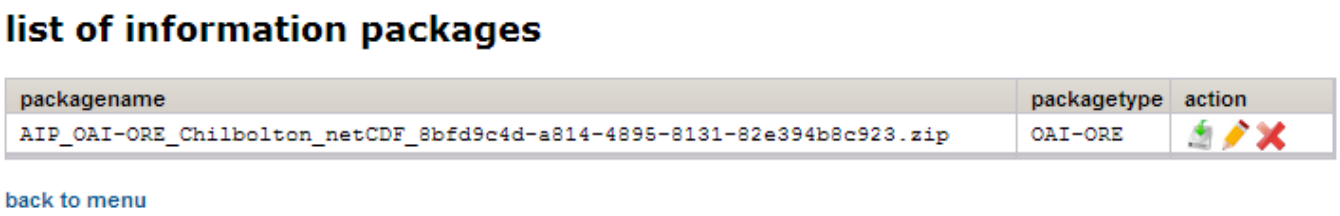


Figure 13: Information package browser

# 5 Reference Manual

None

## *5.1 Keyboard shortcuts*

None

## *5.2 Command-line commands*

None

## *5.3 Public APIs*

None

# 6 Troubleshooting Common Issues

*NA*

# Annex A. Figures and Tables

## A.1. List of Figures

Figure 1: JSF model ..... 10
Figure 2: Packaging API ..... 10
Figure 3: Packaging API for creation and modification of AIPs ..... 11
Figure 4: Top menu of the application ..... 12
Figure 5: Specify title and digital object for upload ..... 12

Figure 6: Specification of IP constituents ..... 13
Figure 7 : Add representation information to an information package ..... 14
Figure 8: Creating aSAFE information package ..... 14
Figure 9: Creating an AIU ..... 15
Figure 10: AIC creation ..... 16
Figure 11: Upload an existing AIP ..... 17
Figure 12: Modification of AIP manifest file ..... 17
Figure 13: Information package browser ..... 18

## A.2. List of Tables

**No table of figures entries found.**

## Annex B. Terminology

| ACRONYM | DESCRIPTION |
|---|---|
| **AIP** | Archival Information Package |
| **ARK** | Archival Resource Key |
| **CDMI** | Cloud Management Interface |
| **DOI** | Digital Object Identifier |
| **ES** | Earth Science |
| **GIS** | Gap Identification Service |
| **KB** | Knowledge Base |
| **OS** | Orchestration Service |
| **OWL** | Web Ontology Language |
| **PI** | Persistent Identifier |
| **PNM** | Preservation Network Model |
| **PURL** | Persistent Uniform Resource Locator |
| **RDF** | Resource Description Framework |
| **RepInfo** | Representation Information |
| **SNIA** | Storage Networking Industry Association |
| **SWKM** | Semantic Web Knowledge Middleware |
| **VM** | Virtual Machine |
| **WP** | Work Package |
| **XAM** | eXtensible Access Method |
| **XML** | eXtensible Mark-up Language |