



SIP Interface Specification

Document information

| | |
|-------------------------|-----------------------------------|
| Workproducts contained: | APP 030: Interface Specifications |
| Author: | H. Verhoeven |
| Version: | 2.5 |
| Date: | 12-01-2006 |
| Classification: | Unclassified |



Revision history

| Revision Number | Revision date | Author | Reviewer | Summary of changes |
|-----------------|---------------|----------------------|---|---|
| 0.1 | 04-10-2001 | Han van Straaten | | Initial version |
| 0.2 | 26-10-2001 | Lucky Kartasasmita | | SIP specification & SIP filename modified |
| 0.3 | 30-11-2001 | Han van Straaten | Hans Verhoeven & Frank Taylor Parkins | Reserved extension name added |
| 1.0 | 11-07-2002 | Frank Taylor Parkins | | Processed review comment |
| 1.1 | 03-10-2003 | Frank Taylor Parkins | Hans Verhoeven | Converted SIP specification to SIP interface specification |
| 1.2 | 21-11-2003 | Frank Taylor Parkins | Hans Verhoeven | Processed review comment |
| 1.3 | 10-09-2004 | Hans Verhoeven | Frank Taylor Parkins | Additions and clarifications as a result of review comments from Kopal (DDB, SUB) project members. |
| 1.4 | 21-04-2005 | Maarten de Wit | Hans Verhoeven & Tanja van Tuijn | Internal review comments processed. |
| 2.0 | 31-05-2005 | Maarten de Wit | | Included support for KOPAL Universal Object Format and multi-organization |
| 2.1 | 14-06-2005 | Maarten de Wit | H. Verhoeven, F. Taylor Parkins & H. van Straaten | Internal review comments processed |
| 2.2 | 28-06-2005 | Maarten de Wit | H. Verhoeven | Further comments processed |
| 2.3 | 11-07-2005 | Hans Verhoeven | | <ul style="list-style-type: none">Updated TM18 and TM22 requirements with the decision to have the process history referenced from the ADMID attribute (REQUPD-01).Removed gzip as archive format as it is only a compression format (REQUPD-02).Updated example xml in DIAS-METS format to have URN IDs for file types (in ImerFile:format). |
| 2.4 | 29-08-2005 | Hans Verhoeven | | Reworked Kopal review comments. |
| 2.5 | 12-01-2006 | Frank Taylor Parkins | | Rework after PTMs 84A and 94A |

All information contained in this document is subject to change without notice. The products described in this document are NOT intended for use in applications such as implantation, life support, or other hazardous uses where malfunction could result in death, bodily injury or catastrophic property damage. The information contained in this document does not affect or change IBM product specifications or warranties. Nothing in this document shall operate as an express or implied license or indemnity under the intellectual property rights of IBM or third parties. All information contained in this document was obtained in specific environments, and is presented as an illustration. The results obtained in other operating environments may vary.

THE INFORMATION CONTAINED IN THIS DOCUMENT IS PROVIDED ON AN "AS IS" BASIS. In no event will IBM be liable for damages arising directly or indirectly from any use of the information contained in this document.

Date: 12-01-2006

Title: SIP Interface Specification
Copyright © by IBM Nederland N.V. 2000, 2006



Table of contents

1 Introduction 4

 1.1 Purpose 4

 1.2 Scope and Approach 4

 1.3 Document overview and References 4

2 SIP data package 5

 2.1 The digital asset 5

 2.2 The technical metadata 5

 2.3 Appearances 5

3 SIP Interface 6

 3.1 Request File types 6

 3.1.1 Purpose 6

 3.1.2 Request 6

 3.1.3 Response 6

 3.2 Request Storage Rules 7

 3.2.1 Purpose 7

 3.2.2 Request 7

 3.2.3 Response 7

 3.3 Request Library Functions (DIAS-KB specific) 7

 3.3.1 Purpose 7

 3.3.2 Request 7

 3.3.3 Response 8

 3.4 Request SIP transport parameters 8

 3.4.1 Purpose 8

 3.4.2 Request 8

 3.4.3 Response 8

 3.5 Submit SIP data package 8

 3.5.1 Purpose 8

 3.5.2 Submission 9

 3.5.3 Response 9

4 SIP data package in detail 10

 4.1 DIAS-KB format 10

 4.1.1 SIP interface 10

 4.1.2 Technical metadata in DIAS-KB format 11

 4.1.3 The special files 11

 4.1.4 Examples 11

 4.2 DIAS-METS format 13

 4.2.1 SIP Interface 13

 4.2.2 Technical Metadata in DIAS-METS format 14

 4.2.3 Examples 20

5 APPENDIX A: Extended Backus-Naur Form 22



1 Introduction

1.1 Purpose

This document describes the Submission Information Package (SIP) interface of the DIAS-Core.

This document is intended for customers of DIAS-core who want to setup an interface to store information in DIAS-Core as well as the development group of DIAS-core who are responsible for the interface development.

See the DIAS-Core Capabilities Specification document [ref 1] for more information about DIAS-Core.

1.2 Scope and Approach

The Digital Information Archiving System (DIAS) is based on the Open Archival Information System ([OAIS]) ISO Reference Model that has been defined by the Consultative Committee of Space Data Systems (CCSDS) , see ref [2]. Interfacing with DIAS means exchanging digital assets and / or metadata by data packages. To add a digital asset to the system a Submission Information Package (SIP) data package is used. The Dissemination Information Package (DIP) is used to retrieve metadata and the digital asset out of the DIAS System. The SIP and DIP data packages are transferred over the SIP and DIP interfaces.

The capabilities of the SIP Interface are labelled with unique numbered items called Interface Items to facilitate tracing and tracking of these items. The Interface Items conform to the following format IFI_SIPnnnX for DIAS-KB and UOF.xx.xxx for DIAS-METS.

1.3 Document overview and References

Chapter 2 introduces the SIP data package.

Chapter 3 describes the SIP interface.

Chapter 4 details the SIP data package.

The following documents are referenced:

| | |
|---------|--|
| DIAS | [1] DIAS-Core Capabilities Specification |
| General | [2] Reference Model for an Open Archival Information System (OAIS), Jan 2002, (http://www.ccsds.org/documents/650x0b1.pdf). [3] METS: The technical metadata in DIAS-METS format is based on version 1.4 of the Metadata Encoding and Transmission Standard of the Library of Congress (http://www.loc.gov/standards/mets). [4] LMER: The lmer version 1.2 standard is described in http://www.ddb.de/standards/lmer/index.htm |

| | |
|------------------|--|
| Date: 12-01-2006 | Title: SIP Interface Specification Copyright © by IBM Nederland N.V. 2000, 2006 |
|------------------|--|



2 SIP data package

The SIP data package is a self describing package of data. Meaning the SIP data package is not just a file with some data. Instead it is a bundle of technical metadata and a digital asset.

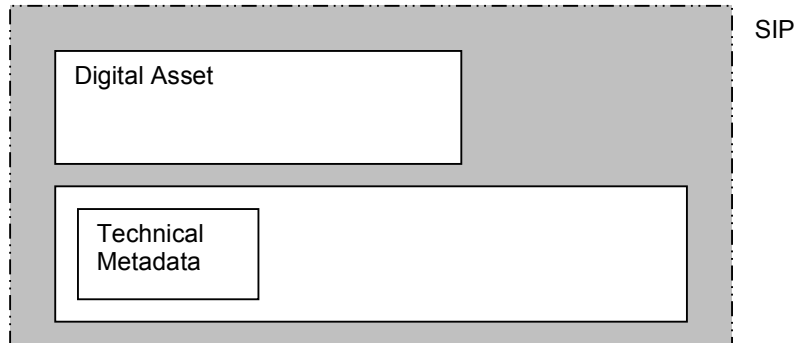


Figure 1: SIP data package

The format of the SIP data package provides a flexible mechanism for constructing a single file that is a container for a digital asset and technical metadata of the digital asset.

2.1 The digital asset

The digital asset, or shortly called the asset, can be as simple as a single file or as complex as a collection of directory trees and files. This is the main content.

The files of the asset can be of different sizes and shall be of one of the file types supported by DIAS-Core. The set of supported file types of the DIAS-Core must be obtained via the SIP interface (See IFI_SIP001A). Prior to constructing SIP data packages this set can be retrieved. The set of supported file types depends on preservation efforts for making and keeping available digital information as well as new capabilities for transformation of content type.

2.2 The technical metadata

The technical metadata consists of standard and custom specific elements describing the asset and the associated directories and files in the asset. See chapter 4, for more information.

2.3 Appearances

The SIP data package can appear in several archive formats.

The SIP file must be offered to the DIAS-Core through FTP (or SFTP). More on this topic in detail in chapter 4, 'SIP data package in detail'.



3 SIP Interface

This interface describes the procedure for Producers to supply DIAS-Core with one or more SIP data packages. The SIP Interface is shown in the following figure as ISIP (Interface-SIP):

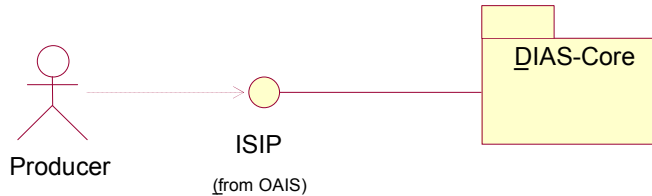


Figure 2: Component diagram SIP interface context

The Producer is responsible for generating a SIP data package complying with the SIP specification and transportation to the DIAS-Core. If a SIP data package does not conform to the specification it will be rejected by DIAS. A rejected SIP data package will be moved into the SIP error area on the DIAS system. The SIP error area must be inspected by the DIAS Administrator to check for rejected SIP's and notify the Producer. After correcting a failed SIP data package it can be submitted to DIAS again. When a SIP data package is accepted by DIAS-Core it will be transformed into technical metadata and an Archival Information Package (AIP). The SIP data package will be deleted upon successful archival.

The following interactions are supported via the SIP interface:

- Request File types and properties;
- Request Storage Rules;
- Request Library Functions;
- Request SIP transport parameters;
- Submit SIP data package.

UOF.MOS.F1 DIAS-Core will be able to handle multiple organizations in a single instance of the DIAS system. DIAS-Core provides a virtual repository for each organization so that assets from the various organizations are separated. Access controls are applied to ensure that users can only access assets and related information from their own organization.

3.1 Request File types

3.1.1 Purpose

In order to get the right values to describe the file type in the SIP data package the Producer can retrieve a list of file type id and file properties. Note that the use of unsupported (or unknown) file types is permitted. However, these files may not open correctly and are not subject to preservation processing.

3.1.2 Request

Request is formed as URL.

IFI_SIP001A <http://<dias>/AccessManager/AccessManager?cmd=filetypes&destId=amgr&sourceId=<src>>

| | |
|--------|---|
| <dias> | hostname of the dias system. |
| <src> | id of the source, the producer (free format: e.g. NA) |

3.1.3 Response

The mapping is returned in xml which conforms to the following DTD;

| | |
|------------------|--|
| Date: 12-01-2006 | Title: SIP Interface Specification Copyright © by IBM Nederland N.V. 2000, 2006 |
|------------------|--|



```

IFI_SIP002A  <!ELEMENT FILETYPES (Item)+          >
              <!ELEMENT Item (#PCDATA)      >
              <!ATTLIST Item
                mimeType          CDATA          #REQUIRED
                fileType         CDATA          #REQUIRED
                fileTypeVersion  CDATA          #REQUIRED
                fileTypeStatus   CDATA          #REQUIRED
                fileTypeID       CDATA          #REQUIRED
                fileExtension     CDATA          #REQUIRED
              >

```

3.2 Request Storage Rules

Storage Rules consist of rule based scenario exists which determines the kind of medium to use to store the AIP.

3.2.1 Purpose

In order to get the valid storage rules that can be used in the SIP data package the SIP producer can retrieve a list of the storage rules. Note that the use of unsupported storage rules results in a rejection of the SIP data package.

3.2.2 Request

Request is formed as an URL.

```
IFI_SIP010A  http://<diashost>/AccessManager/AccessManager?cmd=storagerules&destId=amgr&sourceId=<src>
```

<diashost> hostname of the dias system.
 <src> id of the source, the producer (free format: e.g. NA).

3.2.3 Response

As response a list in XML is returned which conforms to the following DTD;

```

IFI_SIP011A  <!ELEMENT STORAGERULES (Item)+      >
              <!ELEMENT Item (#PCDATA)      >
              <!ATTLIST Item
                StorageRule      CDATA          #REQUIRED
              >

```

3.3 Request Library Functions (DIAS-KB specific)

Library Functions are logical names for storage places. Under those names a rule based scenario exists which determines the kind of medium to use to store the AIP.

3.3.1 Purpose

In order to get the valid library function names that can be used in the SIP data package the SIP producer can retrieve a list of the library functions. Note that the use of unsupported (or unknown) library functions results in a rejection of the SIP data package.

3.3.2 Request

Request is formed as an URL.

```
IFI_SIP010A  http://<diashost>/AccessManager/AccessManager?cmd=libfunctnames&destId=amgr&sourceId=<src>
```

<diashost> hostname of the dias system.
 <src> id of the source, the producer (free format: e.g. NA).

| | |
|------------------|--|
| Date: 12-01-2006 | Title: SIP Interface Specification Copyright © by IBM Nederland N.V. 2000, 2006 |
|------------------|--|



3.3.3 Response

As response a list in XML is returned which conforms to the following DTD;

```

IFI_SIP011A  <!ELEMENT LIBRARYFUNCTIONS (Item)+          >
              <!ELEMENT Item (#PCDATA)           >
              <!ATTLIST Item
                libraryFunctionName  CDATA          #REQUIRED
              >

```

3.4 Request SIP transport parameters

The default way to transport the SIP data package to DIAS-Core is using the FTP protocol (optionally SFTP is supported).

3.4.1 Purpose

Request FTP transport parameters from DIAS-Core.

3.4.2 Request

Request is formed as an URL.

```

IFI_SIP020A  http://<diashost>/AccessManager/AccessManager?cmd=FTPInfo&destId=amgr&sourceId=
<src>&CMUser=<user>&CMPasswd=<password>

```

- <diashost> hostname of the dias system.
- <src> id of the source, the producer (free format: e.g. NA).
- <user> user as defined at DIAS-Core.
- <password> password for above user.

Optionally it is possible to provide this interface via secure-http (https). This might be useful in cases where it is undesired to have unencrypted passwords being sent over the interface.

3.4.3 Response

Returned are the parameters to use for transportation of the SIP data package.

```

IFI_SIP021A  <!ELEMENT FTPINFO (FTPInfoItem)+          >
              <!ELEMENT FTPInfoItem (#PCDATA)       >
              <!ATTLIST FTPInfoItem
                FTP_User              CDATA          #REQUIRED
                FTP_Password         CDATA          #REQUIRED
                FTP_Server            CDATA          #REQUIRED
                SIP.Preloadarea       CDATA          #REQUIRED
              >

```

The parameters are returned as the values of the named attributes.

UOF.MOS.F2 The SIP.Preloadarea will be different for each organization using the same DIAS-Core infrastructure.

3.5 Submit SIP data package

3.5.1 Purpose

Transportation of the SIP data package from Producer to DIAS-Core.

| | |
|------------------|--|
| Date: 12-01-2006 | Title: SIP Interface Specification Copyright © by IBM Nederland N.V. 2000, 2006 |
|------------------|--|



3.5.2 Submission

IFI_SIP030A Irrespective of the transfer protocol the following procedure must be followed:

- Use a unique temporary filename during the transfer to prevent the DIAS scheduler from issuing a load request on an incomplete file (there should be a procedural agreement amongst the various SIP submitters and taking IFI_SIP100A or UOF.sip.F4 for the filename into account).
- Transfer the SIP data package.
- Upon finishing the transfer rename the temporary file to a file with a unique name and extension.
- Only files with supported format will be scheduled for loading in DIAS. The scheduler examines files in order of timestamp.

NOTE:

- If the 'ftp' protocol is used only a rename command within the directory is supported.
- Do not use the word IBM as extension; it is used by the application to indicate a queued SIP.
- DIAS-KB: Submitting the same SIP data package again will normally result in an error since the Asset with the specified NBN will already exist in DIAS.
- DIAS-METS: Submitting the same SIP data package again can result in a duplicate asset in case the object identifier has not been specified (only the PersistentIdentifier parameter in the mets.xml file is mandatory). There can be only one new Asset (an asset without migration history) associated with an externalAssetID. Submitting a new asset more than once will therefore result in an error. Submitting a migrated asset more than once can lead to multiple migrated versions (depends on whether auto-generated version numbering must be performed).

3.5.3 Response

IFI_SIP031A A return code is returned by the transfer-command (e.g. ftp) indicating the success of the transmission.



4 SIP data package in detail

The SIP data package can be in one of the following formats:

- DIAS-KB: the format as created for the Koninklijke Bibliotheek;
- DIAS-METS: the format as created for KOPAL.

4.1 DIAS-KB format

4.1.1 SIP interface

IFI_SIP100A The name of the SIP-file is subject to a naming convention and is constructed as follows:

<IngestUserID>-<NBN>-<asset type>-<date>-<time>.zip.

Due to the limitation in the filename's length in the operating system used, the total filename's length should not be longer than 255 characters.

| Field | Description | Format |
|--------------|---|-----------------------------------|
| IngestUserID | UserID as declared by the system | String, length 8 characters max. |
| NBN | (first part of) National Bibliographic Number | String, length 10 characters max. |
| Asset type | Type of asset (publication) as defined by the system Current supported values: <ul style="list-style-type: none"> • OriginalEpublication, • ConvertedEpublication • InstalledEpublication | String, length is type dependent |
| Date | Date of creation | DD-MM-YYYY |
| Time | Time of creation | hh-mm-ss-SSS |

example:

user123-test0001-OriginalEpublication-24-10-2001-14-56-30-673.zip

IFI_SIP101A Only one asset can be present in a SIP which can be of one publication type.

IFI_SIP102A The files of the asset reside in a sub-directory with the name of the asset type.

IFI_SIP103A At the root level of the SIP resides a 'toc' file holding the description of the SIP. The file has the name SIP_toc.xml. The description of the asset is the layout of the directories and files, with their file type and size, plus additional Technical Metadata.

IFI_SIP104A The 'toc' file (SIP_toc.xml) must include a reference to SIP.dtd containing the formal specification. The reference shall be as follows:

<!DOCTYPE SIP SYSTEM "<http://<dias>/dtd/SIP.dtd>".>

The file adheres to this DTD. For each file the size in bytes and the file type shall be specified. The file type shall be one as defined in the system (system referred to by the URL).

IFI_SIP105A The SIP file sets shall be archived complying with the ZIP format plus:

- Optionally "No Compression" as many image file formats are already compressed;
- Relative path names (No drive letters);
- CRC32 checksum;
- Each directory level as a separate entry;
- The SIP_toc.xml file shall be at the root level of the ZIP file and preferably it should be the first file in the archive (to speed up ingest);

| | |
|------------------|--|
| Date: 12-01-2006 | Title: SIP Interface Specification Copyright © by IBM Nederland N.V. 2000, 2006 |
|------------------|--|



- The size of any entry in the ZIP file should not be greater than 2.147.483.647 bytes (2GB minus 1 byte) (this is a limitation of the ZIP-format).

The ZIP software must be compliant with PKZIP version 2.50 or higher but lower than version 5.0.

4.1.2 Technical metadata in DIAS-KB format

The technical metadata, grouped in a MetadataBlock, consists of a standard set of metadata elements together with a variable set of metadata elements describing the associated directories and files in the asset.

The standard set of metadata consists of the following elements:

- NBN NBN of this asset
- originalNBN NBN of the original publication, shall be same as NBN when this is an original publication
- refPlatformNBN NBN of a platform the publication is dependent on, can be empty
- supplier name of the publisher
- starterFileName name of file to open as default at retrieval (e.g. index.html)
- dateOfCreation date the publication in the format: "DD-MM-YYYY"
- ingestUserID User responsible for ingesting
- sourceType Type of source
- setupFileName Filename to start during a setup (only for assets that have to be installed)
- sourceDescription Free descriptive text

4.1.3 The special files

There is a method for including categories of special files in which directory trees and files can be stored that can contain additional metadata files (like scanned images of CD-booklet) to describe the digital asset. The special files must be of supported DIAS file types in case these files have to be subject of long term preservation. Unknown or unsupported files can be stored. However, these files may not open correctly and are not subject to preservation processing. The special files method consists of applying an additional directory structure for the special files.

4.1.4 Examples

Example of SIP content:

```
SIP_toc.xml
OriginalEpublication/
OriginalEpublication/books/
OriginalEpublication/books/FTP-RFC959.zip
Booklet/
Booklet/Cover.jpg
```

Example of related SIP_toc.xml file:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE SIP SYSTEM "http://banks/dtd/KB.DNEP.SIP.dtd">
<SIP>
  <Asset>
    <Epublication>
      <OriginalEpublication>
        <DirTree rootName="OriginalEpublication">
          <Directory dirName="books">
            <File name="FTP-RFC959.zip" size="34407" type="20"/>
          </Directory>
        </DirTree>
        <DirTree rootName="Booklet">
```

| | |
|------------------|--|
| Date: 12-01-2006 | Title: SIP Interface Specification Copyright © by IBM Nederland N.V. 2000, 2006 |
|------------------|--|



```
<File name="Cover.jpg" size="1202" type="4"/>
</DirTree>
</OriginalEpublication>
</Epublication>
</Asset>
<MetadataBlock>
  <Metadata  supplier="nameOfPublisher"
            starterFileName="books/FTP-RFC959.zip"
            refPlatformNBN=""
            originalNBN="nbn001"
            dateOfCreation="24-07-2003"
            NBN="nbn001"
            ingestUserID="AMS3"
            sourceType="CD-ROM"
            setupFileName=""
            sourceDescription="zomaar iets met een omslag"
            libraryFunctionName="Depot"/>
</MetadataBlock>
</SIP>
```



4.2 DIAS-METS format

4.2.1 SIP Interface

UOF.sip.F1 The SIP Data Package must be in of the following formats:

| SIP Data Package Format | Description | File Extension |
|-------------------------|-------------------------------------|----------------|
| ZIP | ZIP format | .zip |
| GNU-TAR-ZIPPED | GNU-TAR-ZIPPED format | .tar.gz |
| TAR or GNU-TAR | Standard Unix TAR or GNU TAR format | .tar |

Table 1: SIP Data Package Formats

- UOF.sip.F2 Each ZIP format used for the SIP Data Package must be compliant with PKZIP version 2.50 or higher but lower than 5.0.
- UOF.sip.F3 The ZIP format used for SIP Data Packages must only use the standard ZIP-compression features.
- UOF.sip.F4 Each SIP Data Package must have a unique file name, must have a file extension that conforms to the package format and must comply with Unix system standards.
- UOF.sip.F6 Each SIP Data Package must contain a maximum of one Asset.
- UOF.sip.F7 The SIP Data Packages must contain at the root level an XML-document that contains metadata. This file must have the name 'mets.xml' and must be in METS V1.4 format.
- UOF.sip.F11 The technical metadata file (mets.xml) must be in UTF-8 encoding.
- UOF.sip.F8 The ZIP archive format used for the SIP Data Package should comply with the following:
 - Optionally "No Compression" as many image file formats are already compressed;
 - Relative path names (No drive letters);
 - CRC32 checksum;
 - Each directory level as a separate entry;
 - The mets.xml file shall be at the root level of the ZIP file and preferably it should be the first file in the archive (to speed up ingest);
 - The size of any file entry in the ZIP file should not be greater than 2.147.483.647 bytes (2GB minus 1 byte) (this is a limitation of the ZIP format).
- UOF.sip.F9 Each TAR format used for the SIP Data Package must be compliant with Gnu-TAR versions up to 1.15.1. GNU-TAR 1.15.1 does support a variety of tar formats (e.g. gnu, oldgnu, v7, ustar, star, pax, posix).
- UOF.sip.F10 Each TAR archive format used for the SIP Data Package should comply with the following:
 - Relative path names (No drive letters);
 - No multi-volume support;
 - The mets.xml file shall be at the root level of the TAR file and preferably it should be the first file in the archive (to speed up ingest);
 - TAR archives can be very large and can contain very large files. The TAR format uses 12 character positions per TAR entry to store the size of a single file. Those 12 characters are used as 11 octal digits until 8GB ($2^3)^{11}-1$ and are used as base-256 number above 8GB. Conceptually this would mean that a TAR entry can contain a file with a size of $2^{66}-1$ bytes. However, that limit can not be reached since the maximum conceptual size of a whole TAR archive is limited to $2^{63}-1$ bytes. The TAR archives should be smaller than the DIAS Preload file system size. The Preload file system size

| | |
|------------------|--|
| Date: 12-01-2006 | Title: SIP Interface Specification Copyright © by IBM Nederland N.V. 2000, 2006 |
|------------------|--|



is limited by the file system type (e.g. a JFS2 file type on AIX 5L v5.2 with 64-bit kernel can be as large as 16TB or 1TB for the 32-bit kernel. The architectural limit of a JSF2 file system is 4PB. A terabyte (TB) corresponds with 1,024GB and a petabyte (PB) corresponds with 1,048,576GB).

4.2.2 Technical Metadata in DIAS-METS format

The technical metadata in DIAS-METS format conforms to the requirements in the following sections. The technical metadata is contained in an XML-document that is stored in the root of the SIP/DIP Data Packages and has a name "mets.xml".

The following sections only show the data that is of importance in the DIAS-Core processing.

UOF.sipdip.TM23 The technical metadata XML-document must be constructed according to the following XML-Schema's:

- METS v1.4;
- ImerObject v1.2;
- ImerFile v1.2;
- ImerProcess v1.2;
- DublinCore Simple v2002-12-12;
- And schema's of all wrapped XML-extension data.

UOF.sipdip.TM1 The technical metadata in DIAS-METS format is based on version 1.4 of the Metadata Encoding and Transmission Standard of the Library of Congress (<http://www.loc.gov/standards/mets>). The XML-document containing the metadata (mets.xml) must be structured according to the following specification described in Extended Backus-Naur Form (EBNF) (see appendix A for a description of the EBNF notation):

```

DIAS-METS ::=      MetsV1.4
                ;

MetsV1.4( [ID], [OBJID], [LABEL], [TYPE], [PROFILE] )
 ::=      metsHdr
        +      0 { dmdSec } maxDmdSec
        +      1 { amdSec } maxAmdSec
        +      0 { fileSec } maxFileSec
        +      structMap
        ;

metsHdr( [ID], CREATEDATE, [LASTMODDATE], [RECORDSTATUS] )
 ::=      agent
        ;

agent( [ID], ROLE, [OTHERROLE], TYPE, [OTHERTYPE] )
 ::=      name
        ;

name      ::=      xml-characters;

dmdSec( ID, [GROUPID], [ADMID], [CREATED], [STATUS] )
 ::=      mdWrap
        ;

mdWrap( [ID], MIMETYPE, LABEL,
        group METADATA ( MDTYPE, [OTHERMDTYPE] ))
 ::=      xmlData
        ;

```



```

xmlData( ) ::= // XML-data from known other XML-Schema's
;

amdSec( [ID] ) ::=      1 { techMD } maxTechMD
+                      0 { digiprovMD } maxDigiProvMD
;

techMD( ID, [GROUPID], [ADMID], [CREATED], [STATUS] )
::=      mdWrap
;

digiprovMD( ID, [GROUPID], [ADMID], [CREATED], [STATUS] )
::=      mdWrap
;

fileSec( [ID] ) ::=      1 { fileGrp } maxFileGrp
;

fileGrp( ID, [VERSDATE], [ADMID], [USE] )
::=      1 { file } maxFile
;

file( ID, MIMETYPE, [SEQ], SIZE, CREATED, CHECKSUM, CHECKSUMTYPE,
[OWNERID], [ADMID], [DMDID], [GROUPID], [USE] )
::=      1 { FLocat } maxFLocat
;

FLocat( [ID], [USE], group LOCATION ( LOCTYPE, [OTHERLOCTYPE]), group xlink:simpleLink )
::=
;

structMap( [ID], [TYPE], [LABEL] )
::=      div
;

div( [ID], [ORDER], [ORDERLABEL], [LABEL], [DMDID], [ADMID], TYPE, [CONTENTIDS] )
::=      0 { mptr } maxMptr
|        0 { fptr } maxFptr
;

mptr( [ID], group LOCATION( LOCTYPE, [OTHERLOCTYPE], [CONTENTIDS] ),
group xlink:simpleLink ( ) )
::=      ;

fptr( [ID], FILEID, [CONTENTIDS] )
::=
;

```

UOF.sipdip.TM3

The DIAS-METS XML-document must contain a single mets.metsHdr element. The mets.metsHdr must contain following attributes and elements:

- metsHdr.CREATEDATE, date/time of creation of the mets-document;
- metsHdr.agent.ROLE, the role of the agent;
- metsHdr.agent.TYPE, the type of agent;
- metsHdr.agent.name, the name of the agent.

| | |
|------------------|--|
| Date: 12-01-2006 | Title: SIP Interface Specification Copyright © by IBM Nederland N.V. 2000, 2006 |
|------------------|--|



UOF.sipdip.TM4 The DIAS-METS XML-document must contain mdWrap-elements with the xmlData-element for metadata in sections mets.dmdSec, mets.amdSec.techMD and mets.amdSec.digiprovMD.

UOF.sipdip.TM5 The DIAS-METS XML-document must contain the following mandatory sections:

- one mets.metsHdr;
- one or more mets.amdSec.techMD-elements;
- zero or more mets.amdSec.digiprovMD-elements;
- one mets.fileSec.fileGrp with ID attribute equal to "ASSET";
- one mets.structMap with TYPE attribute equal to "ASSET".

UOF.sipdip.TM6 One mets.amdSec.techMD-element must contain a mdWrap.xmlData-element that contains metadata about the Asset in ImerObject format. See ref [4].
<http://www.ddb.de/standards/Imer/index.htm> for more information about the Imer v1.2 standard.
 The ImerObject format is described in Extended Backus-Naur Form by the following structure:

```
ImerObjectV1.2 ::=
+ [objectIdentifier]
+ [name] // not used in DIAS-Core
+ persistentIdentifier // used to reference the asset
+ [transferURL] // not used in DIAS-Core
+ [transferFormat] // not used in DIAS-Core
+ [transferMimeType] // not used in DIAS-Core
+ [transferChecksum] // not used in DIAS-Core
+ 0 { groupIdentifier } maxGroupIdentifier
+ [objectVersion]
+ [masterCreationDate] // not used in DIAS-Core
+ [metadataCreationDate] // not used in DIAS-Core
+ [metadataRecordCreator]
+ [startFile]
+ [numberOfFiles]
+ [status]
+ [comments]
;
```

Remark: the persistentIdentifier is defined as the international unique identification of the object.

UOF.sipdip.TM7 The mets.amdSec.techMD-element must contain a mdWrap.xmlData-element for each file with metadata in ImerFile format. The ImerFile format is described in Extended Backus-Naur Form by the following structure:

```
ImerFileV1.2 ::=
+ [fileIdentifier]
+ [path] // not used in DIAS-Core
+ [name] // not used in DIAS-Core
+ [size] // not used in DIAS-Core
+ [fileDateTime] // not used in DIAS-Core
+ [fileChecksum] // not used in DIAS-Core
+ [mimeType] // not used in DIAS-Core
+ format
+ [formatInfos]
+ [creatorApplication]
+ [viewerApplication]
+ 0 { linkedTo } maxLinkedTo
+ [comments]
+ [category] // not used in DIAS-Core
+ {xmlData}
;
```



- UOF.sipdip.TM8 Each mets.amdSec.techMD.mdWrap.xmlData.ImerFile.xmlData-element must contain metadata in an XML-extension-schema, of which the XML-Schema must be locally available on the DIAS system.

- UOF.sipdip.TM9 Each mets.dmdSec.mdWrap.xmlData-element must contain metadata in an XML-extension-schema, of which the XML-Schema must be locally available on the DIAS system.

- UOF.sipdip.TM10 The DIAS-METS XML-document may contain one mets.amdSec.digiproVMD-element, that is referenced from the fileGrp (with ID equal to "ASSET"). The presence of this element indicates a migrated Asset. The digiproVMD-element must contain a mdWrap.xmlData-element containing metadata about the migration process applied on the Asset in ImerProcess format. The *ImerProcess.oldMetadataRecordCreator* and *ImerProcess.oldObjectIdentifier* specify the Asset that was used as the basis for the migration process. The ImerProcess format is described in Extended Backus-Naur Form by the following structure:

```

ImerProcessV1.2 ::=
+   oldMetadataRecordCreator
+   oldObjectIdentifier
+   oldObjectVersion
+   purpose
+   processCreator
+   permission
+   permissionDate           // not used in DIAS-Core
+   steps
+   { step( NUMBER ) }
+   result
+   completionDate          // not used in DIAS-Core
+   comments
;

```

- UOF.sipdip.TM21 The DIAS-METS XML-document may contain a mets.amdSec.digiproVMD-element for each file. Each digiproVMD-element must be referenced from a file-element in the fileGrp (with ID equal to "ASSET"). The digiproVMD-element must contain a mdWrap.xmlData element containing metadata about the migration process applied on the file in ImerProcess format. The ImerProcess format as used for file migrations is described in Extended Backus-Naur Form by the following structure:

```

ImerProcessV1.2 ::=
+   purpose
+   processCreator
+   steps
+   { step( NUMBER ) }
+   result
+   completionDate          // not used in DIAS-Core
;

```

- UOF.sipdip.TM11 The mets.structMap.div-element must contain a mets.structMap.div.fptr-element for each file that belongs to the Asset. The TYPE attribute of this div-element must be "ASSET".

- UOF.sipdip.TM12 Each fptr-element in a mets.structMap.div-element must have a FILEID attribute referencing a file that has been defined in the mets.fileSec.fileGrp.file-element.

- UOF.sipdip.TM13 Each mets.fileSec.file element must have the following attributes:
 - ID, unique identification within the XML-document;
 - MIMETYPE, the file's Mimetype;
 - CREATED, the creation date/time of the file;
 - SIZE, the file size in bytes;
 - CHECKSUM, the checksum of the file;

| | |
|------------------|--|
| Date: 12-01-2006 | Title: SIP Interface Specification Copyright © by IBM Nederland N.V. 2000, 2006 |
|------------------|--|



- CHECKSUMTYPE, the algorithm used to calculate the checksum.

| | |
|------------------|---|
| UOF.sipdip.TM14 | Each mets.fileSec.file-element must define each physical file with a FLocat-element. The FContent element is not allowed. The FLocat-element must have a URI to the file that is defined in the xlink:href attribute with a relative path. The URI should use the "file://" protocol. The LOCTYPE attribute of each file element must be set to "URL". |
| UOF.sipdip.TM15 | All date, time and date/time values must be specified in accordance with ISO 8601:1988 (E) International Standard. See http://en.wikipedia.org/wiki/ISO_8601 . |
| UOF.sipdip.TM16 | The following checksum types are allowed in the technical metadata for the file-elements: <ul style="list-style-type: none">• SHA-1;• MD5. |
| UOF.sipdip.TM17 | Technical metadata in mets.amdSec.techMD must be referenced from the mets.fileSec in one of the following ways: <ul style="list-style-type: none">• Via the ADMID attribute of the mets.fileSec.fileGrp-element, where the reference is to a ImerObject-element;• Via the ADMID attribute of the mets.fileSec.fileGrp.file-element, where the reference is to a ImerFile-element. |
| UOF.sipdip.TM18 | Technical metadata in mets.amdSec referenced from the fileGrp and file-elements must not result in duplicate information. The following rules apply: <ul style="list-style-type: none">• There must be a maximum of one ImerObject-element for the complete Asset;• There must be a maximum of one ImerProcess-element per file-element;• There must be a maximum of one ImerFile-element per file-element. |
| UOF.sipdip.TM19 | Bibliographical Metadata in mets.dmdSec shall be referenced in one of the following ways: <ul style="list-style-type: none">• Via the DMDID attribute of the mets.structMap.div-element with TYPE equal to "ASSET" where the reference is for the complete Asset; mets.structMap.div-elements with an other TYPE are allowed but not processed.• Via the DMDID attribute of the mets.fileSec.fileGrp.file-element, where the reference is only for the file. Not referenced Bibliographical Metadata shall not be stored in Data Management. |
| UOF.sipdip.TM22 | Digital Provenance metadata in mets.amdSec.digiprovMD must be referenced from the mets.fileSec in one of the following ways: <ul style="list-style-type: none">• Via the ADMID attribute of the mets.fileSec.fileGrp-element, where the reference is to a ImerProcess-element (for migration of the Asset);• Via the ADMID attribute of the mets.fileSec.fileGrp.file-element, where the reference is to a ImerProcess-element (for migration of a file). |
| UOF.sipdip.TM22A | The order of the references in the ADMID attribute of the mets.fileSec.fileGrp-element is important. The following rules must be adhered to: <ul style="list-style-type: none">• If the Asset is a new Asset (no migration process) then the reference is to Technical metadata in mets.amdSec.techMD containing the ImerObject-element about the Asset;• If the Asset is a migrated Asset then:<ul style="list-style-type: none">○ The first reference is to Digital Provenance metadata in mets.amdSec.digiprovMD containing ImerProcess-element of the last migration process applied on the Asset;○ Subsequent references (except the last reference) are to Digital Provenance metadata in mets.amdSec.digiprovMD containing the ImerProcess-element of a migration process applied on the Asset. Each migration process is stored in a separate mets.amdSec.digiprovMD-element. The order of the references in the ADMID attribute must be in chronologically descendent order; |



- The last reference is to Technical metadata in mets.amdSec.techMD containing the lmerObject-element about the Asset.

UOF.sipdip.TM25

The parameters of repeating groups in the DIAS-METS technical metadata and the lmer-extension schemas must comply with the values in the following table.

| Parameter | Limit Level |
|--------------------|-------------|
| maxDmdSec | 5 |
| maxAmdSec | 5000 |
| maxFileSec | 1 |
| maxTechMD | 5001 |
| maxDigiprovMD | 5001 |
| maxFileGrp | 1 |
| maxFile | 5000 |
| maxFLocat | 1 |
| maxMptr | 250 |
| maxFptr | 5000 |
| maxGroupIdentifier | 100 |
| maxLinkedTo | 5000 |

Table 2: Limits of Parameters



4.2.3 Examples

Example of SIP content:

mets.xml
31_2004_Article_7001_Abstract.html
BodyRef\
BodyRef\PDF\
BodyRef\PDF\31_2004_Article_7001.pdf

Example of related mets.xml file:

```
<?xml version="1.0" encoding="UTF-8" ?>
<mets xmlns="http://www.loc.gov/METS/" xmlns:lmer="http://www.ddb.de/LMER"
  xmlns:lmerObject="http://www.ddb.de/LMERObject" xmlns:lmerProcess="http://www.ddb.de/LMERprocess"
  xmlns:lmerFile="http://www.ddb.de/LMERfile" xmlns:mets="http://www.loc.gov/METS/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xlink="http://www.w3.org/1999/xlink"
  xsi:schemaLocation="http://www.loc.gov/METS/ http://www.loc.gov/standards/mets/version14/mets.xsd
  http://www.ddb.de/LMER lmer.xsd http://www.ddb.de/LMERObject http://www.ddb.de/standards/lmer/lmer-
  object.xsd http://www.ddb.de/LMERprocess http://www.ddb.de/standards/lmer/lmer-process.xsd
  http://www.ddb.de/LMERfile http://www.ddb.de/standards/lmer/lmer-file.xsd">
<metsHdr CREATEDATE="2005-05-06T12:00:00">
  <agent ROLE="ARCHIVIST" TYPE="ORGANIZATION">
    <name>DDB</name>
  </agent>
</metsHdr>
<amdSec>
  <techMD ID="AMD0001">
    <mdWrap LABEL="LMERObject" MDTYPE="OTHER" MIMETYPE="text/xml">
      <xmlData>
        <lmerObject:name>INVARIANT THEORY FOR NON-ASSOCIATIVE REAL TWO-DIMENSIONAL ALGEBRAS AND ITS
        APPLICATIONS</lmerObject:name>
        <lmerObject:persistentIdentifier>urn:nbn:de:spr:001-000000001</lmerObject:persistentIdentifier>
        <lmerObject:objectVersion>1</lmerObject:objectVersion>
        <lmerObject:masterCreationDate>2003-10-09T18:30:00</lmerObject:masterCreationDate>
        <lmerObject:metadataCreationDate>2005-05-06T15:20:00</lmerObject:metadataCreationDate>
        <lmerObject:metadataRecordCreator>IBM</lmerObject:metadataRecordCreator>
        <lmerObject:status>harvested</lmerObject:status>
        <lmerObject:numberOfFiles>2</lmerObject:numberOfFiles>
        <lmerObject:comments>Beispielobjekt</lmerObject:comments>
      </xmlData>
    </mdWrap>
  </techMD>
  <techMD ID="AMD0002">
    <mdWrap LABEL="LMERfile" MDTYPE="OTHER" MIMETYPE="text/xml">
      <xmlData>
        <lmerFile:format>urn:diasid:fty:loc01:020050701120000000000</lmerFile:format>
        <lmerFile:formatInfos>Linearized PDF</lmerFile:formatInfos>
        <lmerFile:comments>Beispieldatei</lmerFile:comments>
        <lmerFile:creatorApplication>Acrobat Distiller 5.0 (Windows)</lmerFile:creatorApplication>
        <lmerFile:viewerApplication>Adobe Reader</lmerFile:viewerApplication>
        <lmerFile:category>multimedia</lmerFile:category>
        <lmerFile:xmlData /
          remark: some tags removed
        </lmerFile:xmlData>
      </xmlData>
    </mdWrap>
  </techMD>
  <techMD ID="AMD0003">
    <mdWrap LABEL="LMERfile" MDTYPE="OTHER" MIMETYPE="text/xml">
      <xmlData>
        <lmerFile:format> urn:diasid:fty:loc01:0200507011200000000010</lmerFile:format>
        <lmerFile:viewerApplication>Firefox</lmerFile:viewerApplication>
        <lmerFile:comments>Beispieldatei</lmerFile:comments>
        <lmerFile:category>text</lmerFile:category>
        <lmerFile:xmlData
```



```
    remark: some tags removed
  </lmerFile:xmlData>
</xmlData>
</mdWrap>
</techMD>
</amdSec>
<fileSec>
  <fileGrp ID="ASSET" ADMID="AMD0001">
    <file ID="FILE0001" ADMID="AMD0002" MIMETYPE="application/pdf" CREATED="2004-05-13T14:59:55" SIZE="236598"
CHECKSUM="d9510684fa6b6f9d3dcfefb2629635a7" CHECKSUMTYPE="MD5">
      <Flocat LOCTYPE="URL" xlink:href="file:///BodyRef/PDF/31_2004_Article_7001.pdf" />
    </file>
    <file ID="FILE0002" ADMID="AMD0003" MIMETYPE="text/html" CREATED="2004-05-13T12:34:24" SIZE="3682"
CHECKSUM="e8c2e038ec52fec67ddc4ca29322202f" CHECKSUMTYPE="MD5">
      <Flocat LOCTYPE="URL" xlink:href="file:///31_2004_Article_7001_Abstract.html" />
    </file>
  </fileGrp>
</fileSec>
<structMap TYPE="ASSET">
  <div ID="div0" TYPE="ASSET">
    <fptr FILEID="FILE0001" />
    <fptr FILEID="FILE0002" />
  </div>
</structMap>
</mets>
```



5 APPENDIX A: Extended Backus-Naur Form

The Extended Backus-Naur Form (EBNF) is often used to specify data structures. There is a lot of information available on the Internet about this notation. EBNF is an extension of the standard Backus-Naur Form. Standard BNF is a formal notation to describe the syntax of a given language or more generally a grammar. Since a data structure can be expressed as a grammar it is very useful to describe the possible constructs.

The following explains some of the constructs:

| | |
|--------------------|---|
| ::= | is defined as |
| + | and |
| | or |
| [] | optional |
| { } | repetition |
| x { <i>item</i> }y | minimally x occurrences of the item and maximally y |