Metadata

Content

Further information

Submission information packages (SIP)

Archival information packages (AIPs)

Dissemination information packages (DIPs)

Preservation Management

Rosetta product documentation: Rosetta DNX Profile in *Rosetta AIP Data Model*

Rosetta METS Profile

Descriptive metadata

Descriptive metadata is recorded in the digital long-term archiving system with the aim of uniquely describing and identifying the objects. The descriptive metadata is intended to ensure the long-term assignment of the content of the object and is created by the relevant TIB specialist teams, delivered directly to the long-term archiving team by the data producers or collected by the long-term archiving team from various data sources. Descriptive metadata in the DC section of ie.xml (see Specifications for archival information packages (AIPs) must be available as Dublin Core. This metadata is indexed. Various metadata standards (MARC, Dublin Core, MODS, EAD, NISO, MIX and others) can be integrated into the source MD section of ie.xml.

There are currently several methods for recording descriptive metadata:

- enrichment with metadata from the union catalog K10plus
- the collection of Dublin Core metadata supplied via the OAI interface from the institutional repository of Leibniz Universität Hannover
- the capture of supplied Dublin Core metadata in the dc section of ie.xml
- the collection of supplied metadata from source systems as source metadata in ie.xml

Additional catalogue systems may be connected as required.

Enrichment with metadata from the union catalogue K10plus

Librarians collect metadata on the objects according to the RDA cataloguing standard. Older catalogue records are available based on the RAK-WB standard.

CMS enrichment is conducted during the transition from operational to permanent archival storage. CMS enrichment involves querying metadata via the SRU interface of the Gemeinsamer Verbundkatalog and mapping the output to Dublin Core. Mapping governs the assignment of PICA+(only in german) fields to the relevant Dublin Core qualified elements, as well as the scope, structure and content of the descriptive metadata.

The metadata are written to a separate catalogue.xml and given an identifier; the identifier is written to the ie.xml. The metadata from the catalogue.xml are indexed.

Mapping table from PICA+ to Dublin Core

Dublin Core	Pica+	Remark	Mandatory
Title	036C/00	Collective title of the multi-part monograph and the subcategorisations (in master form)	Yes, in combination with 021A
isPartOf	036C/00	Collective title of the multi-part monograph and the subcategorisations (in master form)	No
title	021A	Main title, other title information, information on responsibilities	Yes
alternative	046B	Specification of parallel titles that are not on the title page	No
Alternative	021F*	Parallel titles	No
Alternative	046C*	Deviating titles	No
Creator	028A	Person/family as first creator (formerly: first author)	No
Creator	028B/..	Second author and additional authors	No
creator or contributor	028C /00*	Person/family as additional creators, other contributing persons and families	No
Creator	029A	Body / first originator	No
Contributor	028M	Creator from superordinate C set	No
Contributor	028G-028L	Other person, dedicatee (old prints), censor (old prints), artistic contributor (old prints), other non-involved persons or persons named in the title (old prints)	No

creator or contributor	029F /00*	Secondary body, other bodies involved	No
Contributor	030F*	Congress	No
Publisher	033A*	Publication details (place of publication and publisher)	No
Publisher	037C*	Note in university publications	No
Issued	011@	Publication date	No
language	010@	Language codes	No
identifier	005A*	ISSN	No
identifier	004U*	Persistent identifier: URN	No
identifier	004V	Persistent identifier: DOI	No
identifier	004R*	Persistent identifier: Handle	No
identifier	004A*	ISBN	No
identifier	007F*	Report number	No
identifier	007G	ID number given by first cataloguing institution (EKI)	Yes
identifier	003@	PICA production number (PPN)	No
isPartOf	036E*	Monographic series	No
isPartOf	036F*	Monographic series (link)	No
isPartOf	039B*	Link to larger entity (in the case of articles)	No
Bibliographic Citation	031A	Differentiating information about the source	No
description	032@	Edition statement	No
description	032B	Reprint note	No

Metadata delivered by data producers or harvested by a platform

For metadata supplied by the data producer and harvested by platforms, the Long-Term Preservation team has defined minimum sets for different forms of publication in Dublin Core. Metadata that is not available as Dublin Core can be included in the archive package as source metadata.

Monographs

Content	captured in	mandatory
title	dc:title	✅
author names (repeatable)	dc:creator / dc:contributor	✅
ISBN	dc:identifier xsi:type=”dcterms:ISBN"	❌
DOI	dc:identifier xsi:type=”dcterms:URI"	❌
other unique identifiers (repeatable)	dc:identifier	❌
language	dc:language	❌
publication year	dcterms:issued	❌
abstract	dcterms:abstract	❌
publisher	dc:publisher	❌

Journal articles

Content	captured in	mandatory
article title	dc:title	✓
author names (repeatable)	dc:creator / dc:contributor	✓
journal title; *volume* , *issue* , publication year	dcterms:isPartOf	✓/✗/✗/✗
DOI	dc:identifier xsi:type="dcterms:URI"	✗
ISSN	dc:identifier xsi:type="dcterms:ISSN"	✗
language	dc:language	✗
publication year	dcterms:issued	✗
abstract	dcterms:abstract	✗

Identifying metadata

Identifiers used

Internal-system identifiers at the object level

Rosetta creates and allocates various internal-system identifiers.

- Identifier for objects: system internal identifier generated by Rosetta to identify IEs, representations, files and packets during deposit and SIP processing.
- Event type identifier: Rosetta-defined ID for an event category (see Event).
- Identifier for processes: ID assigned by Rosetta for executed processes, for example a Preservation Action (see Administrative metadata and Logging of preservation actions).
- Rights identifier: the ID of a policy, for example, a configured usage right (see), a retention policy, or a delivery license.
- Identifier for agents: the ID of an agent in the sense of PREMIS, for example, a producer, a plug-in, a connected system, or a user.

The internal-system identifiers are unique and permanent within the system.

If new policies or processes are defined by a user, the system assigns a new unique ID. Additional identifiers are recorded in the metadata.

Catalogue metadata

Another optional external identifier in the ie.xml is the catalogue identifier from the Gemeinsamer Verbundkatalog (Union Catalogue, GVK). By means of the SRU interface to the catalogue system, configured in Rosetta, the catalogue identifier is used to enrich the object with descriptive metadata.

The catalogue metadata of each individual object are deposited in a dedicated XML file, which is linked to the IE via metadata identifiers (mId)

Identifiers are allocated PREMIS-compliant for objects, agents, events and rights. The following table lists several examples of identifiers.

Examples of identifiers based on the PREMIS model

Object	Example
SIP ID	539308
IE ID	IE2980431
REP ID	REP2980432
File ID	FL2980433

Identifier for the catalogue system	GBV881139254
mId	1032839
Versioning	V9-IE1024027.xml
Agent	
Producer ID	40030044
Producer agent ID	2122740
Plug-in ID	58638365
Catalogue system	TIB
User ID	2122740
Event	
Material flow ID	641084
Deposit ID	548243
Event ID	62
Process ID	50532321
Rights	
Boilerplate ID	TIB_OA_mit_CC
Access right policy ID	16728
Retention policy ID	NO_RETENTION

External identifiers

External identifiers can be recorded in Dublin Core format, such as a DOI, a handle or a URN.

Allocation of identifiers

Internal-system identifiers are automatically allocated by the system as unique identifiers. The identifiers are given different additions, depending on the object type.

Structural metadata

Structural metadata are stored in the ie.xml as DNX and METS elements.

TIB stores 1-n representations per IE, each consisting of 1-n files. Representations are described using the DNX element “Preservation type”. Each ie.xml contains the IDs of all associated representations and files. In the file group, files are assigned to a file ID via their path, and each file ID is also assigned to a representation ID. In the StructMap, the files per representation are arranged in a logical sequence that can be transferred to a viewer.

Structural metadata assignment to METS and DNX elements

Metadatum	Element and metadata standard	Value
Representations		
Original files	Preservation type (DNX)	MASTER

Modified copy of original files before ingest	Preservation type (DNX)	PRE-INGEST_MODIFIED_MASTER
Modified copy of original files after ingest	Preservation type (DNX)	MODIFIED_MASTER
Access copy	Preservation type (DNX)	DERIVATIVE_COPY
Relationships		
Belonging of files to a representation	fileGrp (METS)	REP ID, File ID, storage path to the file
Coherence of files within a representation	structMap (METS)	Representation ID, label structure, file ID
Restoration of authentic data structure		
Original file name	fileOriginalName (DNX)	Original file name
Original file path	fileOriginalPath (DNX)	Original file path

The relationships between files within a representation are recorded in the “structMap” METS element. In addition, the original file name and path of every file are recorded in the metadata, documenting which directory structure a file was stored in during deposit.

Technical metadata

Technical metadata are captured in Rosetta as DNX metadata. DNX was specified by the software manufacturer Ex Libris and is based on PREMIS, but extends the standard by further elements. DNX documentation is publicly available. Updating of DNX is managed and monitored by the Rosetta user community.

The PREMIS standard defines a number of “basic concepts” as technical metadata in the semantic units ObjectCharacteristics, SignificantProperties, OriginalName and Storage. The relevant concepts of the unit are provided in the table below. In this case, the PREMIS concept is mapped to the DNX element, as well as information about at which point the concept can be allocated values and whether TIB has implemented the recording.

Technical metadata mapping from PREMIS to DNX

PREMIS semantic unit / component from	DNX element	Method of recording	Used by TIB
ObjectCharacteristics			
compositionLevel	compositionLevel	Pre-ingest	No
fixity			
messageDigestAlgorithm	fileFixty. fixityType	See K10	Yes
messageDigest	fileFixty. agent	See K10	Yes
messageDigestOriginator	fileFixity. fixityValue	See K10	Yes
size	generalFileCharacteristics. fileSizeBytes	Determined automatically during ingest	Yes
format			
formatDesignation			

formatName	fileFormat.formatName	Automatically during ingest	Yes
formatVersion	fileFormat.formatVersion	Automatically during ingest	Yes
formatRegistry			
formatRegistryName	fileFormat.formatRegistry	Automatically during ingest	Yes
formatRegistryKey	fileFormat.formatRegistryId	Automatically during ingest	Yes
formatRegistryRole	fileFormat.formatRegistryRole	Automatically during ingest	Yes
formatNote	fileFormat.formatNote	Manually by the technical analyst during ingest involving manual allocation to format	Yes
creatingApplication			
Last name	creatingApplication.creatingApplicationName	As part of the pre-ingest process, manually via the web editor or automatically as part of a preservation plan	No TIB does not use this semantic concept to capture the creatingApplication, but records the values – provided they can be recorded by the technical metadata extractor – under significant properties as part of the technical metadata
Version	creatingApplication.creatingApplicationVersion	See above	See above
dateCreatedByApplication	creatingApplication.dateCreatedByApplication	See above	See above
creatingApplicationExtension	creatingApplication.creatingApplicationExtension	See above	See above
inhibitors			
inhibitorType	inhibitors.inhibitorType	As part of the pre-ingest process or manually via the web editor	Yes
inhibitorTarget	inhibitors.inhibitorTarget	See above	See above
inhibitorKey	inhibitors.inhibitorKey	See above	See above
significantProperties			
significantPropertiesType	significantPropertiesType	Metadata extraction in the validation stack	Yes
significantPropertiesValue	significantPropertiesValue	See above	Yes
significantPropertiesExtension	significantPropertiesExten	See above	Yes

originalName	fileOriginal Name	Automatically during ingest	Yes
	fileOriginal Path	Automatically during ingest	Yes
storage			
contentLocation			
contentLocationType	fileLocation Type	Automatically during ingest (system – loading stage)	Yes
contentLocationValue	fileLocation	Is not used by Rosetta at present.	

Logging of preservation actions

Defined events

Modifications to AIPs are recorded at the IE level as DNX metadata. The DNX schema was specified by the software manufacturer Ex Libris and is based on PREMIS, but extends the standard by additional elements. DNX documentation is publicly available. Updating of DNX is managed and monitored by the Rosetta user community.

Several examples of defined events are described in the table below. The complete list of defined events is documented in the Rosetta Configuration Guide.

Examples of events

Event ID	Description
23	Started Validation Stack Stage
24	Virus check performed on file
25	Format Identification performed on
27	Fixity check performed on file
147	Arranger Decline IE
164	Object viewing is denied due to Access Rights restrictions
165	Technical Metadata extraction performed on file
166	Completed Validation Stack Stage
167	Metadata enrichment (CMS fetching)
217	Failed MD Validation Stage
339	Preservation plan has been created
372	Manually Set Format Library ID on File
380	Representation has been added
381	Risk identification performed on file
397	METS Validation Failed

A user with the role of “Administrator” can define which events from the list should be logged.

Logging of event metadata

The system automatically records the defined event metadata. Event metadata are written to the ie.xml for every defined event.

Administrative metadata

Defined administrative metadata

Administrative metadata are captured as DNX metadata at different levels in Rosetta. DNX was specified by the software manufacturer Ex Libris and is based on PREMIS, but extends the standard by further elements. DNX documentation is publicly available.

At the IE level, the standardised name of the applicable licence agreement is recorded as the Dublin Core element dctersm:license. The applicable licence text is deposited in Rosetta as a “boilerplate”; the text contains information about which actions may be performed on the object.

TIB understands administrative metadata to mean:

- Metadata that document the provenance of objects
- Legal metadata
- Metadata recorded for the purpose of organising objects

Provenance information

Provenance information	DNX element
Acquisition team responsible	**producer**
	producerId
	userIdAppId
	defaultLanguage
	authorativeName
	firstName
	lastName
	middleName
	address1
	address2
	address3
	address4
	zip
	emailAddress
	telephone1

Legal metadata

Legal metadata	Element

Access rights	accessRightsPolicy (DNX)
	policyId (DNX)
	policyDescription (DNX)
Title of the transfer agreement as concluded between TIB and the data producer or the long-term archiving team and the transferring TIB team, or standardized name of the applicable license text.	Dcterms: license (Dublin Core)
Access right to the document as granted by the data producer/rights holder /copyright holder	dcterms: accessRights
Legal basis for long-term archiving	dc:rights
Right of use in trigger case	dc:rights
Authorized users in trigger case	dcterms: accessRights
Rights holder	dcterms: rightsHolder

Organisational metadata

Organisational metadata	DNX element
General object characteristics (at the IE representation and file level, respectively)	**objectCharacteristics**
	ObjectType
	parentID
	groupID
	creationDate
	createdBy
	modificationDate
	modifiedBy
	owner
IE characteristics	**generalIECharacteristics**
	submissionReason
	status
	statusDate
Identification of object type	IEEntityType
Identifier for the collection and production process	UserDefinedFieldA
Marking for non-valid or password-protected objects in the context of Preservation as a Service	UserDefinedFieldB
Marking for images from defective media devices	UserDefinedFieldC

Preservation level	**preservationLevel**
	preservationLevelValue
Representation characteristics	**generalRepCharacteristics**
	label
	preservationType
	usageType