eCH-0160 Archival Submission Interface

Name	Archival Submission Interface (SIP)
Standard number	eCH-0160
Category	Standard
Maturity level	Implemented
Version	1.0
Status	Remplace
Approval date	2012-11-21
Issue date	2015-06-24
Replaces standard	
Languages	German (original), French (translation), English (translation)
Authors	Digital archiving expert group Marguérite Bos, Swiss Federal Archives, marguerite.bos@bar.admin.ch Georg Büchler, KOST georg.buechler@kost.admin.ch Anouk Dunant Gonzenbach, Geneva State Archives anouk.dunant-gonzenbach@etat.ge.ch David Gubler, Fachlabor Gubler AG d.gubler@swiss-mikrosave.ch Martin Kaiser, KOST martin.kaiser@kost.admin.ch Markus Lischer, Lucerne State Archives markus.lischer@lu.ch Nicole Martini, Swiss Federal Archives nicole.martini@bar.admin.ch Urs Meyer, Swiss Federal Archives urs.meyer@bar.admin.ch Ursina Rodenkirch-Brändli, Graubünden State Archives ursina.rodenkirch@sag.gr.ch Stefan Ryter, Bern State Archives stefan.ryter@sta.be.ch
Publisher / distributor	Verein eCH, Mainaustrasse 30, Postfach, 8034 Zürich Tel. 044 388 74 64, fax 044 388 71 80 www.ech.ch / info@ech.ch

Summary

This document specifies an interface for submitting dossiers and documents from records and process management systems, specialist applications / databases and file collections to an archive. It sets out the specifics of the Submission Information Package (SIP), one of the core concepts of the Open Archival Information System OAIS (ISO 14721:2003), the central reference model for the archiving of digital documents.

The basis for this document is the Submission Information Package (SIP) Specification, version 4.0, of the Swiss Federal Archives SFA. Additions and corrections from the archives represented in the Koordinationsstelle für die dauerhafte Archivierung digitaler Unterlagen (Coordination Agency for the Preservation of Electronic Files, KOST) have been incorporated into this Specification. The interface is already being used by the SFA and various Swiss state archives.

Table of contents

1 Introduction

1.1 Status of the document

Replaced: The document has been replaced with a new latest version. Its use is still possible but it is recommended to apply the latest version.

1.2 Objectives

The **objectives** of the Archival Submission Interface (SIP) Specification are as follows:

- The Specification contains the requirements for the creation of a Submission Information Package (SIP), the digital information package for submitting digital archive records to an archive. It stipulates what form a digital submission to the archive must take, and therefore provides submitting authorities with a specification for internal use and a tool for communication with their service providers and software manufacturers.
- The Specification provides information on the requirements that must be complied with when implementing digital interfaces in GEVER systems for submission purposes and when creating digital submissions from databases and file systems.

The Archival Submission Interface Specification is aimed at the following **target groups:**

- submitting authorities
- service providers and software manufacturers of applications that implement archival submission interfaces
- members of archive staff

1.3 Structure of the document

1.3.1 Structure of chapters

Each chapter in this Specification is constructed according to the same pattern. After a brief introduction, the requirements are listed in a table.

ID	Description of requirement	M/O
contains the ID of the requirement	contains the text of the requirement	stipulates whether mandatory or optional

A requirement is frequently further explained by means of recommendations and examples, both of which are specifically indicated as such.

ID	Description of requirement	M/O
G_4.1-1	Text of requirement **Example** Text of example ***Recommendation*** *The text of recommendations is always in italics.*	M

1.3.2 ID for requirements

The requirements are unambiguously identifiable by means of an ID.

ID
G_4.1-1

This ID is constructed according to the following pattern:

G_ Letter + _ identifies main chapters

G_	=	General requirements
P_	=	Requirements for primary data
M_	=	Requirements for metadata
S_	=	Requirements for package structure
T_	=	Requirements for package transfer

4.1-1 The number begins with the number of the chapter (which groups together requirements on the same topic), and the number after the dash is consecutive, thus designating all the requirements in the chapter.

1.3.3 Distinction between mandatory and optional requirements

Each requirement is either mandatory or optional. This is indicated by a single letter:

Abbreviation	Meaning
M	Mandatory requirement This requirement must be met in order to obtain a valid submission.
O	Optional requirement This requirement should be met. It simplifies the handling of a submission for both the submitting authority and the archive and thus constitutes best practice.

1.3.4 Notation of folders, files and folder structures

The following symbols and parameters are used for the notation of folders, files etc.

Symbol	Meaning
`/`	Folder
`header/`	A folder with the name “header”
`xy.txt`	File (with file extension “txt”)
`dir1/`	Example folders (in red)
`abc.pdf`	Example files (in red)
…	Placeholder for files or folders that are not relevant to the explanation.
`[]`	Placeholder for an expression or basic type such as “string”, “integer” etc.

1.4 Requirement for further detail

Institutions implementing this Specification may be required to supply further detail on the following points:

- permitted archivable formats for primary data (see 3.2)
- maximum size of a SIP (see 5)
- maximum number of files in a folder within a SIP (see 5.2)
- name of the top-level folder / SIP identifier (see 5.4).
- ingest requirements for the classification system (see 4.8-2)

2 Overview

2.1 Scope

This document contains the specification for the Submission Information Package (SIP), the form in which digital documents are to be submitted to an archive. (See Fig. 1).

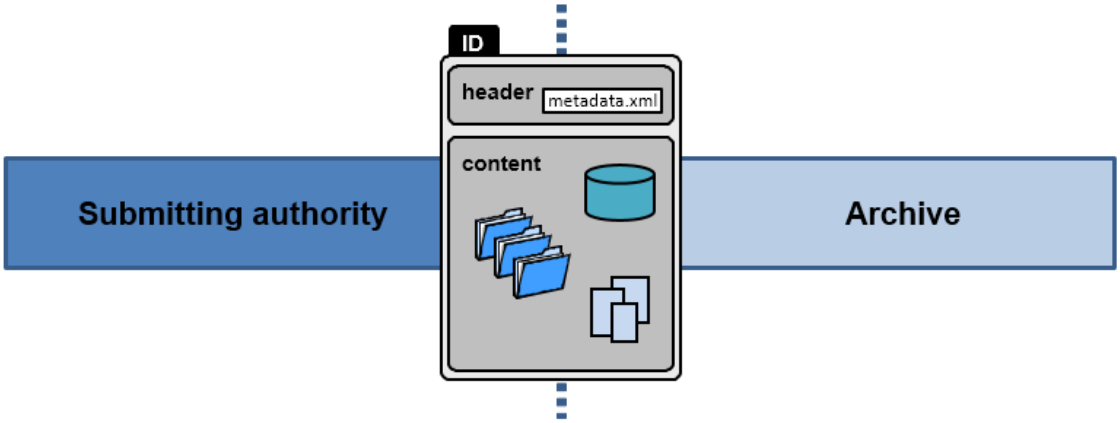


Fig. 1: Scope of the SIP submission format

This Specification sets out the requirements for all technical implementations designed to extract a SIP for the archive from an existing system or create a SIP from a file archive (e.g. from a GEVER system or using data from a database or a collection of related digital documents from a file archive).

2.1.1 Definitions

Within this Specification, a distinction is made at the top level between two submission types: GEVER submissions and FILES submissions. The FILES type can also occur as FILES with integrated documentation. This is not a separate submission type but rather a sub-type of the FILES submission. This typology is derived from the pre-archive origin of the digital archive records.

Submission type		Description
GEVER		The digital archive records in the submission are derived from a records and process management (GEVER) system.
FILES		The digital archive records in the submission are derived from a file archive or a relational database or another system.
	FILES-SIP with integrated documentation	If the FILES submission contains digital documents from a database or other system for which system documentation is additionally submitted to the archive, there are further structural requirements in addition to the general requirements for the FILES-SIP. Such submissions are FILES-SIPs with integrated documentation on the data.

2.1.2 Requirements for each submission type

Not all chapters of this Specification have to be applied when creating a SIP. The requirements for a SIP may differ, depending on the pre-archive origin of the documents. Figure 2 helps to clarify which chapters of this Specification are relevant for each submission type. A precise listing of the relevant chapters is set out in the overview in Appendix C.

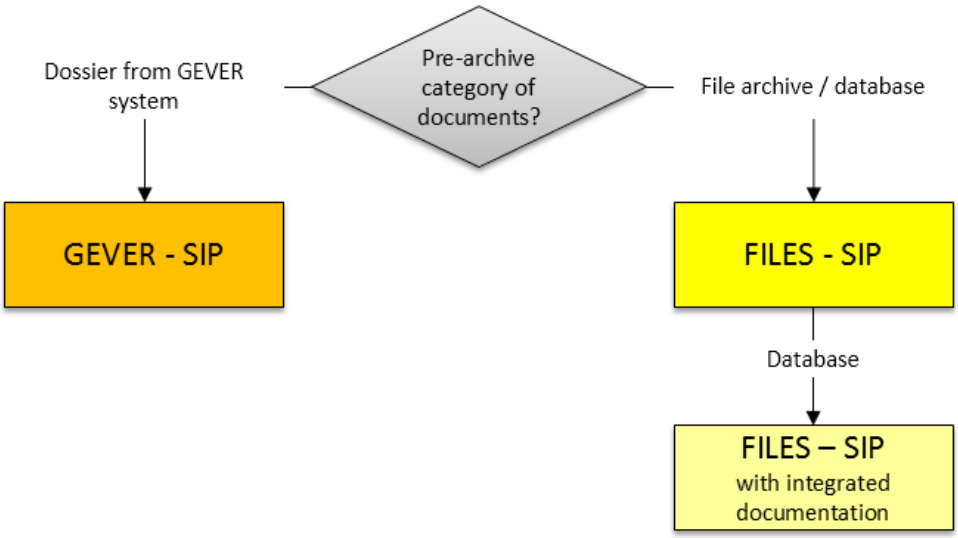


Fig. 2: SIP submission categories

2.2 Area of application

2.2.1 Process

This Specification was developed on the basis of the Swiss Federal Archives' submission process for digital documents. Documents that are deemed archivable must be prepared, technically and in terms of content, in accordance with the requirements of this Specification and submitted in a correctly created SIP. If they are not, the documents will be returned to the submitting authority for the necessary corrections to be made or a new SIP to be created that meets the requirements.

2.3 The submission package – SIP

The submission takes the form of an information package based on the model in the OAIS (Open Archival Information System) standard (see Fig. 3). This standard was developed by the Consultative Committee for Space Data Systems (CCSDS). One of the main elements of the OAIS is that all related information is collated into a "package".

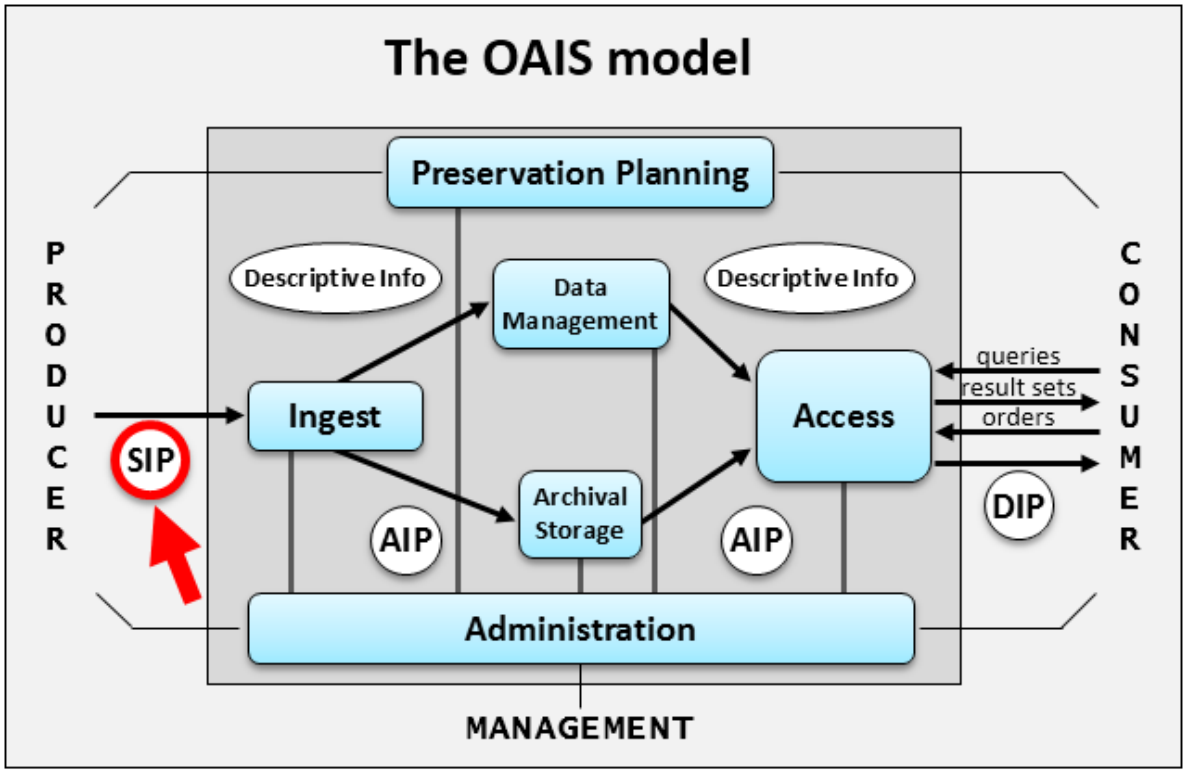


Fig. 3: OAIS model showing the SIP

There are three package types: SIP, AIP and DIP.

Package type		
SIP	Submission Information Package	Package for submitting digital documents
AIP	Archival Information Package	Package for archiving digital documents
DIP	Dissemination Information Package	Package for disseminating digital documents

This Specification defines the requirements for a SIP, both archivally and technically. Compliance with this Specification ensures that all packages are compatible with a corresponding archiving infrastructure and archiving processes for digital archive records.

2.4 Contents

Based on the OAIS model, the SIP contains two separate parts according to this Specification, which must always be submitted together. The first part is the *header*. This contains all the metadata (descriptive information about the package and the primary data) in XML format. The second part is the *content*. This contains all the primary data. (See Fig. 4).

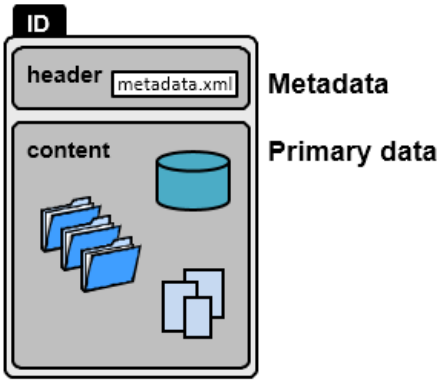


Fig. 4: Structure of a SIP – metadata and primary data

2.5 Relationship between submission and package

Digital documents must be submitted in the form of a SIP.

ID	Description of requirement	M/O
G_2.5-1	A submission to the archive consists of one or more packages (SIPs). This reduces complexity and simplifies handling for both the submitting authority and the archive.	M

2.6 Security settings for files in the package

The files contained in a SIP must be accessible without restriction. If the submitting authority is transferring digital documents that are subject to enhanced protection to the archive (and only in such cases), it may be necessary to protect the SIP as a whole, for example using a password. The requirements for the transfer are to be agreed in advance by the submitting authority and the archive.

ID	Description of requirement	M/O
G_2.6-1	The files in a SIP must not be encrypted.	M
G_2.6-2	The files in a SIP must not be password-protected.	M

3 Primary data

3.1 Basics

Primary data are all digital documents produced in the submitting authorities (as part of business matters, in dossiers, in records and business management systems, in databases). There are various types, such as text files, images, databases, presentations, etc. Primary data can be born-digital documents or digitised documents. All documents created in this way are designated as primary data.

3.2 Formats

The primary data contained in a SIP submitted to the archive must be in an archivable format. The archive responsible defines the formats that are accepted as archivable. These are not part of this Specification.

ID	Description of requirement	M/O
P_3.2-1	The primary data must be integrated into the SIP in a suitable archivable format for the documents. ***Recommendation*** *It is expedient to create the documents in an archivable format from the outset or convert them to an archivable format as soon as possible. In a GEVER system, conversion must take place no later than when the dossier is closed. In any event, the primary data must be converted before the documents are integrated into a SIP.*	M
P_3.2-2	The file extensions must comply with the usual conventions regarding formats of the files concerned. **Example** An image file in TIFF format: p00010.tif A sound file in WAVE format: das_lied_1.wav	M

4 Metadata

Metadata are information about data. They describe the primary data more precisely. Metadata may be stored in separate files (as “separate metadata”, for example in an XML file) or already contained in a primary file as “embedded metadata”. An MS Word document, for example, contains embedded metadata in the form of the title, subject, author etc. Owing to possible format conversions in the archive, the preservation of embedded metadata cannot be guaranteed.

This chapter specifies the metadata for SIPs of the following submission types:

- GEVER: submission from a GEVER system (“GEVER submission”)
- FILES: submission of a collection of files (“FILES submission”). FILES submissions also include submissions from relational databases with integrated documentation.

4.1 General requirements

ID	Description of requirement	M/O
M_4.1-1	All metadata that are to be preserved throughout all archiving processes must be present as separate metadata in a separate file in XML format named `metadata.xml` (distinct from the primary data). The Specification prescribes both the structure of the metadata file and the content of the metadata (see the chapters on the XSD and data dictionary). These requirements must be complied with exactly.	M

4.2 Conceptual data model

The conceptual data model provides an overview of all the fundamental components (entities) in the `metadata.xml` and their representation in the XSD underlying the `metadata.xml`. The entities that belong to the digital information package as well as the relationships between these entities are represented graphically in the conceptual data model. Each entity comprises part of the metadata in the `metadata.xml`. The conceptual data model is present for the `metadata.xml` of both packages of the GEVER submission type and packages of the FILES submission type.

ID	Description of requirement	M/O
M_4.2-1	Compliance with / implementation of the conceptual data model is a binding requirement for all submission types. It provides guidance and an overview of all the existing entities.	M
M_4.2-2	The conceptual data model distinguishes between two submission types: GEVER and FILES. The correct submission type must be selected, depending on the provenance of the archive records.	M

Notation of the conceptual data model

The notation used for the conceptual data model in this Specification complies with the UML (Unified Modeling Language) standard.

Description of the entities

Entity	Description
Package	The *package* entity describes the “packaging” of the submission in a technical container (folder). The name of the package changes depending on the phase in its lifecycle (SIP/AIP/DIP). The package used for submitting digital archive records is always a SIP.
Submission	The *submission* entity contains all the information on the submitting authority and the submission to the archive.
Provenance	The *provenance* entity represents the records creator and, in the case of submissions of the FILES type, also the information system from which the submission comes.
Classification system	The *classification system* entity describes the registry plan, filing plan or structure plan used by the authority, or the classification of the archive records.
Classification system position	The *classification system position* entity describes the properties of the individual classification system positions in the classification system.
Dossier	The *dossier* entity describes the properties of the related archive records from a business matter or another classification structure (database, file archive). The metadata from the entity are to be used both for dossiers and for subdossiers.
Document	The *document* entity represents the smallest level on which the content of the archive records is represented. This entity contains the metadata necessary to understand the documents themselves or their content.
Table of contents	The *table of contents* entity represents the table of contents contained in the `metadata.xml` which comprises all the folders and files.
Folder	The *folder* entity represents the folders in the package in which the package is structured.
File	The *file* entity represents the primary files in the file system and also contains information for the identification and description of the individual files.
Unstructured appendix	The *unstructured appendix* entity represents the description of files that are not actual archive records and are added to the package as additional information in a folder specially designated for this purpose. This entity is “deprecated”. Its use is not recommended. It is possible that it will not be included in future versions of the standard.

Entity	Description
Archival process	The *archival process* entity is not filled with metadata until it reaches the archive and documents activities such as acceptance checks in a SIP or format conversions in an AIP. This entity is not of significance for the creation of a SIP. It may not be used when creating one.
Archival note	The *archival note* entity is used by the archive to create notes on items from the individual entities. This entity is not of significance for the creation of a SIP. It may not be used when creating one.

Relationships between the entities

The relationships between the entities can be viewed in the conceptual data model from the package-internal perspective. The conceptual data model thus reproduces the world of the individual package (and therefore a single digital submission) as it is represented in the `metadata.xml`. Entities may be linked by references or may reference themselves, as is the case with the classification system position, dossier and folder. In the case of the classification system position, this is necessary because various hierarchies down to the lowest classification system position (the rubric) must be reproduced in a classification system. A dossier may contain various subdossiers that are, however, all subsumed within the *dossier* entity in the world of the data model. The *folder* entity can also reference itself, and reproduces the various hierarchies in the folder structure of a package.

Within the package, the relationships between the *provenance* entity and the *submission* entity, and between the *classification system* entity and the *submission* entity, are always 1:1. There can be only one provenance and one classification system for the digital archive records in a package. However, a number of submissions of one provenance can be made, and over time, more than one submission may be made from a classification system. This fact is not reproduced in the conceptual data model, however, as this contains only the view of an individual package.

The *archival process* and *archival note* entities contain metadata that is only of importance once the submission is in the archive (when the package has been transferred from the submitting authority to the archive). These entities are not significant for the creation of a package in the submitting authority. For this reason, they and the *unstructured appendix* entity, which is not significant for the SIP, are listed here solely for the sake of completeness and to document all the entities in the XSD.

4.3 Data model for the GEVER submission type

ID	Description of requirement	M/O
M_4.3-1	The GEVER submission type applies to all submissions that contain digital documents from a records and business management (GEVER) system. The data model for GEVER submissions is shown in Fig. 5. The following entities occur in the GEVER-SIP (in the metadata): • Provenance • Package • Submission • Classification system • Classification system position • Table of contents • Dossier • Document • File • Folder The *archival process* entity only becomes relevant once the package has been transferred to the archive. The number of archival processes in a SIP from a submitting authority prior to transfer is therefore always 0. The *archival note* entity only becomes relevant once the package has been transferred to the archive. The number of archival notes in a SIP from a submitting authority prior to transfer is therefore always 0. The *unstructured appendix* entity is not intended to occur in a GEVER-SIP.	M

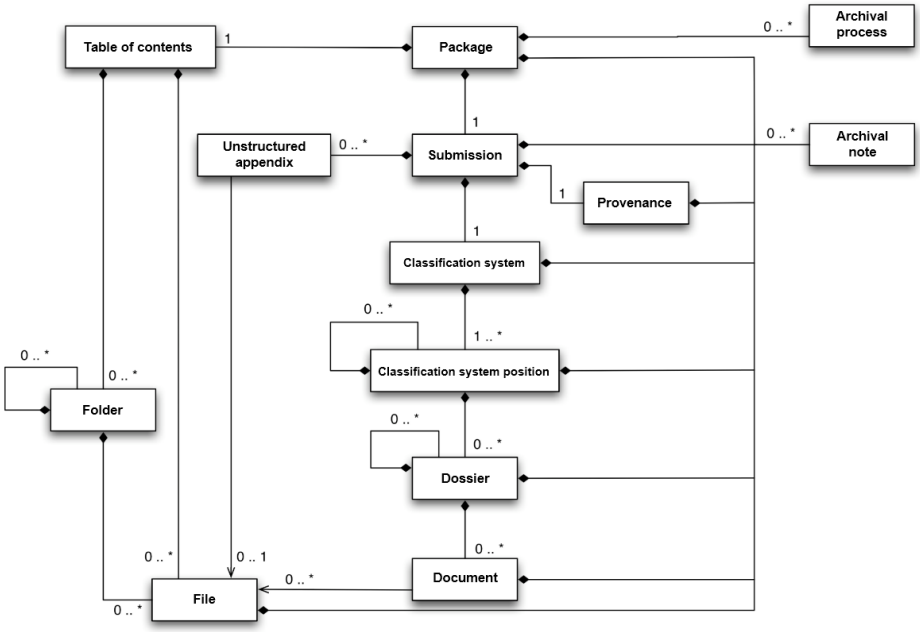


Fig. 5: The conceptual data model applied to the GEVER submission type

4.4 Data model for the FILES submission type

ID	Description of requirement	M/O
M_4.4-1	The FILES type applies to all submissions that contain digital documents from a relational database or file archive. The data model for FILES submissions is shown in Fig. 6. The following entities occur in the FILES-SIP (in the metadata): • Provenance • Package • Submission • Classification system • Classification system position • Table of contents • Dossier • Document • File • Folder The *document* entity need not necessarily be included in a FILES-SIP. It can be used as required. The *archival process* entity only becomes relevant once the package has been transferred to the archive. The number of archival processes in a SIP from a submitting authority prior to transfer is therefore always 0. The *archival note* entity only becomes relevant once the package has been transferred to the archive. The number of archival notes in a SIP from a submitting authority prior to transfer is therefore always 0. The *unstructured appendix* entity is not intended to occur in a FILES-SIP.	M

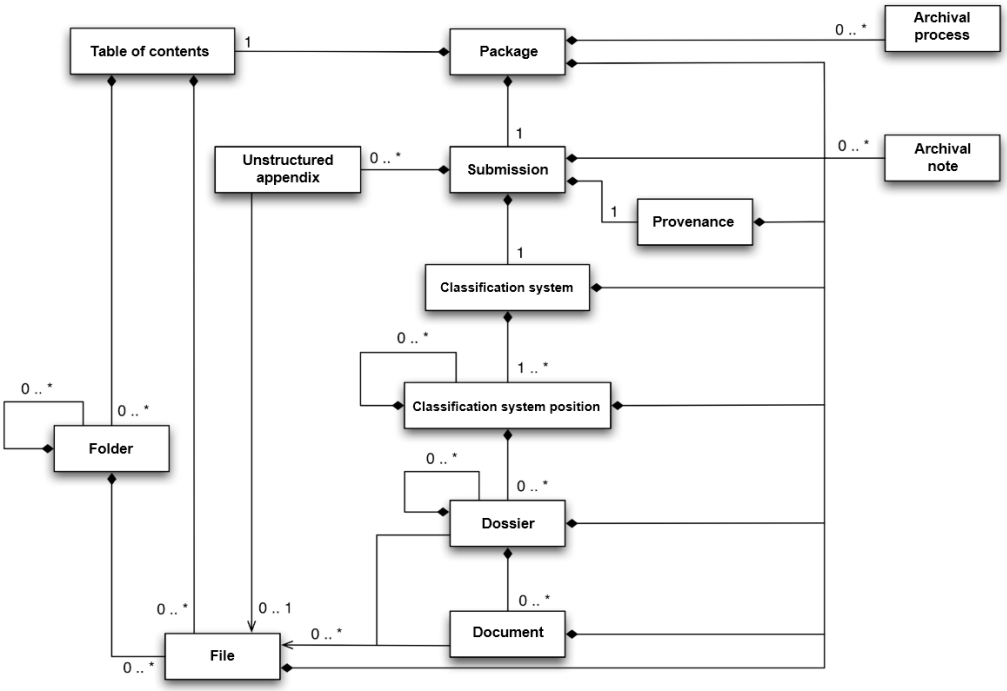


Fig. 6: The conceptual data model applied to the FILES submission type

ID	Description of requirement	M/O
M_4.4-2	There are two cases of relationships between the *dossier*, *document* and *file* entities in the FILES-SIP (see Fig. 7). Case A (marked in blue): A file is directly assigned to a dossier. A dossier has 0..* files. There are no documents. Case B (marked in green): A file is assigned to a document. The document is assigned to the dossier. A dossier has 0..* files. 1 document consists of 0..* files. (as in the GEVER type) Case A and case B can occur simultaneously in a package. Case A and case B can occur simultaneously in a dossier.	M

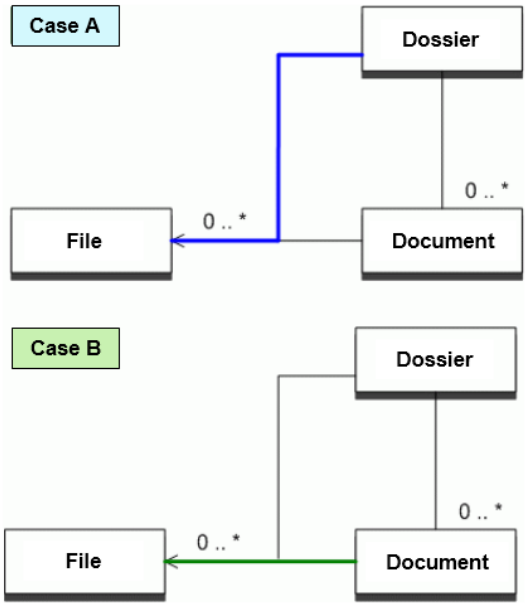


Fig. 7: Different possible relationships between dossier, document and file

4.5 Data dictionary

In the data dictionary, the individual metadata elements are divided up by entity and described in greater detail. The data dictionary is based on the data model already presented. Each chapter of the data dictionary deals with an entity.

ID	Description of requirement	M/O
M_4.5-1	All metadata designated mandatory in the data dictionary must be transferred to the SIP.	M

ID	Description of requirement	M/O
M_4.5-2	Metadata that are known from the originating system (GEVER, FILES) and via all other elements of the submission and for which a suitable attribute is present in the metadata of the SIP must be transferred to the SIP. ***Recommendation*** *The more precisely the digital documents are described by means of metadata, the easier it is to handle the archive records as soon as they are made available and disseminated again.*	M

The data dictionary, with all the metadata elements of the SIP and the attributes used for the specification of the metadata, is appended to the Specification.

4.6 XSD

The XSD (XML Schema Definition) including documentation describes the structure (or grammar) of the XML file containing the metadata of a submission (`metadata.xml`). It is the technical implementation of the data dictionary and the data model and at the same time the basis for validating the `metadata.xml` in the SIP.

ID	Description of requirement	M/O
M_4.6-1	The XSD must be complied with for the `metadata.xml`. This means that each `metadata.xml` in a SIP must be capable of being positively validated against the XSD.	M

The XSD is appended to the Specification in digital form, along with documentation.

4.7 Structure of the table of contents in the `metadata.xml`

ID	Description of requirement	M/O

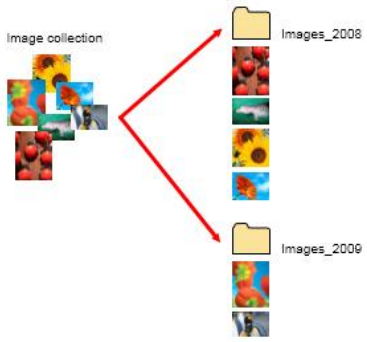
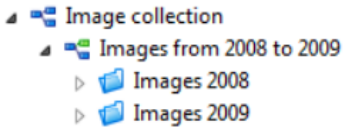
M_4.7-1	All folders and files in the package (contents of the `content/` folder and `header/` folder) must be listed in the table of contents of the `metadata.xml` in accordance with the requirements, with all the metadata.
	The only exception to this rule is for the `metadata.xml` file in the `header/` folder. This file must not be listed in the `metadata.xml`, as it would otherwise refer to itself.
	
	Example
	Structure of the SIP (schematic)
	`SIP_20090824_Bm/`
	`header/`
	`metadata.xml`
	`xsd/`
	`submission.xsd`
	`….xsd`
	`content/`
	`d0001/`
	`d0002/`
	`p0001.pdf`
	
	The corresponding table of contents of a `metadata.xml`
	`<?xml version="1.0" encoding="UTF-8"?>`
	`<paket xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://bar.admin.ch/arelda/v4" xsi:type="paketSIP" schemaVersion="4.0">`
	`<paketTyp>SIP</paketTyp>`
	`<inhaltsverzeichnis>`
	`<ordner>`
	`<name>header</name>`
	`<originalName>header</originalName>`
	`<ordner>`
	`<name>xsd</name>`
	`<originalName>xsd</originalName>`
	`<datei id="dat00001">`
	`<name>ablieferung.xsd</name>`
	`<originalName>ablieferung.xsd</originalName>`
	`<pruefalgorithmus>MD5</pruefalgorithmus>`
	`<pruefsumme>aa1f7ad13064e1643ac85478296165af</pruefsumme>`
	`</datei>`
	`</ordner>`
	`</ordner>`
	`</ordner>`
	`...`
	`<ordner>`
	`<name>content</name>`
	`<originalName>content</originalName>`
	`<ordner>`
	`<name>d000001</name>`
	`<originalName>Veranstaltungen</originalName>`
	`<ordner>`
	`<name> d000002</name>`
	`<originalName>Ausstellung</originalName>`
	`<datei id="dat000015">`
	`<name>p000015.pdf</name>`
	`<originalName>flyer_ausstellung.pdf</originalName>`
	`<pruefalgorithmus>MD5</pruefalgorithmus>`
	`<pruefsumme>f20168fd9b04d7385b8a68e59f092c69</pruefsumme>`
	`</datei>`
	`</ordner>`
	`</ordner>`
	`</ordner>`
	`</inhaltsverzeichnis>`
	`...`
		M

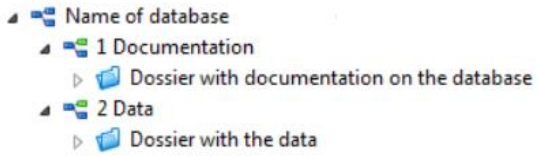
4.8 Mapping metadata from the originating system onto SIP metadata

In order for the metadata to be correctly entered in the SIP and extracted from the system, mapping between the metadata fields in the system of the records creator and those in the SIP is required.

ID	Description of requirement	M/O
M_4.8-1	In the case of metadata that are already available at the submitting authority, the content, format and any data models already present often do not correspond to the structures required to create a SIP. For this reason, mapping between the metadata created by the records creator / submitting authority and the metadata in the SIP for submitting the archive records to the archive must be carried out. The submitting authority is responsible for this task, for both GEVER-SIPs and FILES-SIPs.	M
M_4.8-2	In the case of a GEVER-SIP, the classification system is to be adopted into the SIP. ***Recommendation*** *It is advisable to first carry out mapping between the metadata of the original GEVER system and the metadata of the SIP. An example of part of such a mapping for a GEVER system can be found in Appendix D.*	M
M_4.8-3	In the case of a FILES-SIP, both the archive structure (physical classification) at folder and file level and the corresponding system at classification system and document level (logical classification) must be described in the metadata. By this means, the documents are combined into sensible units in accordance with their content and compiled into a dossier that can be disseminated. ***Recommendation*** *If no such classification is already present, it should be created on the basis of the contents of the documents.* *It is advisable to reproduce systems and classifications of file structures that are already present or sensible above the dossiers in the classification system and to combine related documents into dossiers.*	M

ID	Description of requirement	M/O

	Example
	A file archive contains a collection consisting of a large number of unclassified images. The images could now, for example, be classified by year of creation and combined into folders at file archive level. The content of these folders is then described as a dossier in the metadata.
	 
	
	The content folder in the SIP will then look like this:
	`content/`
	`Images_2008/`
	`Bug.tif`
	`Dolphin.tif`
	`Sunflower.tif`
	`Orange_Flower.tif`
	`Images_2009/`
	`Several_Flowers.tif`
	`Penguins.tif`
	By way of a model, this classification, together with the mapping to the dossiers, could be represented as follows:
	 
	
	= Classification system
	= Classification system position
	= Dossier	

ID	Description of requirement	M/O
M_4.8-4	For FILES-SIPs with integrated documentation (e.g. databases archived using SIARD), a simple classification system is to be created so that the dossiers can be integrated with the documentation and the data. **Example** By way of a model, a classification of this type could be depicted as follows:  = Classification system = Classification system position = Dossier	M

4.9 Metadata on closure periods

Information on the closure periods of the documents submitted must be included in the metadata of the SIP.

ID	Description of requirement	M/O
M_4.9-1	Information on the closure periods of the documents in the package must be entered in the metadata available for this purpose in the `metadata.xml`. The metadata are as follows: Closure period category: Details of the relevant article in the legal basis (where applicable) Closure period: Duration of the closure period in years The metadata pair stipulating the closure periods is available in three entities: Submission Classification system position Dossier The closure periods must be included in the metadata either globally for the entire submission (same closure period for all documents) or for each classification system position or for each dossier. *Recommendation* *It is advisable to maintain the metadata on data protection, classification and publication status for dossiers and documents in the GEVER system itself. The closure periods can be derived from this information when creating a SIP and described directly in the metadata.*	M

4.10 Metadata on the time periods of the dossiers

ID	Description of requirement	M/O
M_4.10-1	As soon as an indication of the period during which a dossier was created (creation period metadatum in the *dossier* entity) is estimated, the justification for this estimate is to be entered in the creation period note metadatum of the dossier concerned.	M

4.11 Metadata on the checksums of the files

ID	**Description of requirement**	**M/O**
M_4.11-1	A checksum must be calculated for each file in the package (except the `metadata.xml` file) and entered in the metadata for that file in the `metadata.xml`. The following metadata are available for this purpose in the *file* entity: checksum algorithm : Designation of the algorithm used to create the checksum. checksum : Checksum value, depending on the checksum algorithm used The following hash functions can be used as checksum algorithms: MD5 SHA-1 SHA-256 SHA-512	M

4.12 Referencing the attribution to dossiers in the metadata

ID	Description of requirement	M/O

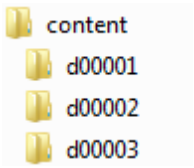
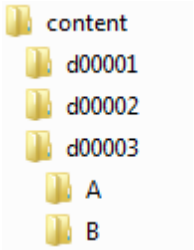
M_4.12-1	The file ref metadatum is used to establish the link between the files in the package and their attribution to the documents in the dossiers or direct to a dossier. This link is recorded by appropriate referencing of the file ref to the corresponding file in the table of contents of the `metadata.xml`.
	This referencing is the unambiguous allocation of the contents of the files to the dossiers.
	`<paket>`
	`  <inhaltsverzeichnis>`
	`  <ordner>`
	`  <name>header</name>`
	`...`
	`  </ordner>`
	`  <ordner>`
	`  <name>content</name>`
	`  <ordner>`
	`  <name>FLZ_1</name>`
	`  <datei id="dat15">`
	`  <name>Modell_A.tif</name>`
	`  <originalName>Modell_A.tif</originalName>`
	`  <pruefalgorithmus>MD5</pruefalgorithmus>`
	`  <pruefsumme>3b0b980e1a1bb76ee49abd9db66b0a47<pruefsumme>`
	`  </datei>`
	`...`
	`  </ordner>`
	`  </inhaltsverzeichnis>`
	`  <ablieferung xsi:type="ablieferungFilesSIP">`
	`  <ablieferungstyp>Files</ablieferungstyp>`
	`...`
	`  <dossier id="dos1">`
	`  <titel>Flugzeuge 2009</titel>`
	`  <erscheinungsform>digital</erscheinungsform>`
	`  <entstehungszeitraum>`
	`  <von>`
	`  <datum>2009</datum>`
	`  </von>`
	`  <bis>`
	`  <datum>2009</datum>`
	`  </bis>`
	`  </entstehungszeitraum>`
	`  <bemerkung></bemerkung>`
	`  <dateiRef>dat15</dateiRef>`
	`  </dossier>`
	`...`
	`  </ablieferung>`
	`</paket>`
	N.B. From XSD version 4.0 onwards, a file in a FILES-SIP can also be assigned simultaneously to a number of documents or dossiers. However, this option should only be used when absolutely necessary for subject matter-related reasons Example: a number of documents describe successive sections of the content of a single video file.
		M

5 Structure of the SIP

5.1 Framework requirements and limitations on size of the package

ID	Description of requirement	M/O
S_5.1-1	A SIP may not be larger than 8 GB. ***Recommendation*** *For faster transfer and dissemination, it is advisable to keep the size of an individual SIP for submission below 2 GB.* *This can be achieved by regular submission of documents to the archive and by good segregation planning.*	M
S_5.1-2	The archive can only accept a SIP larger than 8 GB in exceptional cases and for good reasons. In such cases, the submitting authority must contact the archive before creating the submission and transferring the package.	M

5.2 Framework requirements and limitations on number of files in the package

ID	Description of requirement	M/O
S_5.2-1	A SIP must not contain more than 1,000,000 files.	M
S_5.2-2	A single folder within a SIP should not contain more than 5,000 files. ***Recommendation*** *This requirement should be complied with, because otherwise problems may occur when displaying the SIP in Explorer. If a single folder contains more than 5,000 files, it is recommended to add folders to further subdivide this number.* **Example**  Folder `d00003/` originally contains 9,000 files.  	O

ID	Description of requirement	M/O
	Two folders are added within folder `d00003/` and the 9,000 files are divided up between them so that no one folder contains more than 5,000 files. In principle, these folders may be given any desired name (in this case `A/` and `B/`), but names within the folder `d00003/` should be unambiguous and kept as short as possible so as to avoid making the path too long.	

5.3 Permitted characters for folder and file names

Only a limited set of characters may be used in the names of files and folders. This prevents incorrect conversions within the names that could otherwise occur when packages are transferred and processed between different operating systems.

ID	Description of requirement	M/O
S_5.3-1	The requirements regarding the set of characters to be used for folder and file names must be complied with in their entirety.	M
S_5.3-2	The permitted set of characters for folder and file names is a subset of US-ASCII. The following characters are permitted: Letters: A-Z a-z Numbers: 0-9 Other: ! # $ % ( ) + , - . = @ [ ] { } ~_ SPACE The character codes for the permitted characters are identical in the US-ASCII, ISO-8859 and UTF-8 character sets.	M
S_5.3-3	If the names of folders and files contain characters that are not permitted, these must be normalised before submission to the archive. ***Recommendation*** *Appendix E contains best-practice recommendations that can be followed when normalising names.*	M

ID	Description of requirement	M/O
S_5.3-4	If two files or folders that are stored in the same folder are normalised to the same name (which does not happen often), conflicts occur. To avoid this, the names of the files concerned must be supplemented during or before normalisation, for example by adding a suffix to the basic name. The original file name must be recorded appropriately in the metadata before it is changed, so that it is not lost. **Example** Conflict during normalisation: Berufe_J Jaeger.pdf Jäger.pdf Jäger.pdf → Jaeger.pdf Jaeger.pdf → Jaeger.pdf Suitable supplement: Jäger.pdf → Jäger_1.pdf → Jaeger_1.pdf Jaeger.pdf → Jaeger_2.pdf → Jaeger_2.pdf	M
S_5.3-5	If existing files in the package are renamed during creation of the package (e.g. for technical reasons), the original file names must be entered in the metadata element “originalName” in the *file* entity. **Example** Renaming of file in the package Jäger.pdf → Jaeger.pdf Entries in the metadata elements in the metadata.xml `<dateiName>Jaeger.pdf</dateiName>`[1] `<originalName>Jäger.pdf</originalName>`	M

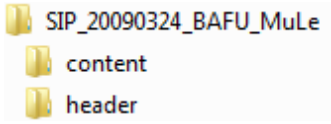
[1] “dateiName” = “fileName”

5.4 Structure of a SIP

The basis of a SIP is a file-folder structure. There are rules governing the way this is constructed as well as the naming conventions (content and structure of names) for the individual folders and files. The initial overview below shows the top-level folders in a SIP (Fig. 8).

```
SIP_[identifiers]/
      header/
            metadata.xml
            xsd/
                  arelda.xsd
                  ….xsd
                  ….xsd
      content/
```

Fig. 8: Basic structure of an information package

ID	Description of requirement	M/O
S_5.4-1	Each package has exactly one top-level folder, which comprises the entire content of an individual package. **Examples** In the example below, the top-level folder is named `SIP_20090324_BAFU_MuLe`. 	M

<table>
<tr><th>ID</th><th>Description of requirement</th><th>M/O</th></tr>
<tr><td>S_5.4-2</td><td>In a SIP, the name of the top-level folder always begins with the character sequence <code>SIP_</code> followed by specific identifying information [identifiers].

Recommendation
The name of the top-level folder is to be constructed in accordance with the following pattern:
<table>
<tr><td>SIP_</td><td>[submission date]</td><td>_</td><td>[name of submitting authority]</td><td>_</td><td>[reference]</td></tr>
</table>
[submission date]: date in the format: YYYYMMDD
[name of the submitting authority]: The name of the submitting authority is given in the form of the official abbreviation (e.g. EPA, SDC, FSO). If no official abbreviation is available, a short but meaningful designation of the submitting authority should be chosen.
[reference]: Use of the reference by the submitting authority is optional. It can consist, for example, of the abbreviation for the name of the contact or database or the submission number.
Example
Three SIPs, with the top-level folder shown in each case.
SIP_20090324_BAFU_MuLe
SIP_20100425_DEZA_AmAn
SIP_20090612_BFS_DBWohnungen</td><td>M</td></tr>
<tr><td>S_5.4-3</td><td>The top-level folder always contains exactly two folders. These are named <code>header/</code> and <code>content/</code>.
It is not permitted to place other folders or files directly in the top-level folder.

Example
In all SIPs, the content of the top-level folder looks like this:
content
header</td><td>M</td></tr>
</table>

ID	Description of requirement	M/O
S_5.4-4	The `header/` folder contains the metadata and their XSD files. It never contains primary data. The predefined folder in the `header/` folder of a SIP for submission is called: `xsd/` No further folders are permitted in the `header/` folder. The predefined file in the `header/` folder is always called `metadata.xml`. This is the only file permitted in the `header/` folder. The metadata of the package are stored in this file. Example The content of the `header/` folder of a SIP as delivered. xsd metadata.xml	M
S_5.4-5	All XSD files that belong to the version of XSD used are stored in a separate folder in the `header/` folder with the name `xsd/`. **Example** All the XSD files for the metadata version ARELDA_v4.0 in the folder `xsd/`. ablieferung.xsd archivischeNotiz.xsd archivischerVorgang.xsd arelda.xsd base.xsd datei.xsd dokument.xsd dossier.xsd ordner.xsd ordnungssystem.xsd ordnungssystemposition.xsd paket.xsd provenienz.xsd zusatzDaten.xsd	M
S_5.4-6	The `content/` folder contains the primary data of a SIP. This means that the folder contains all the files that make up the content of the dossiers in the SIP.	M

ID	Description of requirement	M/O

S_5.4-7	It is recommended to place all folders that correspond to a dossier on the same folder level within the `/content` folder.
	
	Example
	GEVER
	The folders in blue each contain files that belong to the same dossier. One folder (e.g. `d00001`) corresponds to one dossier
	`content/`
	`  d00001/`
	`  d00002/`
	`  d00003/`
	`  d00004/`
	`  d00005/`
	
	FILES
	The folders in blue each contain files that belong to the same dossier. One folder (e.g. `Tree_1`) corresponds to one dossier.
	`content/`
	`  part_1/`
	`  Tree_1/`
	`  Tree_2/`
	`  part_2/`
	`  Fish_1/`
	`  Fish_2/`
	`  Fish_3/`
		O

5.5 Path lengths and hierarchies in the SIP

ID	Description of requirement	M/O

S_5.5-1	The length of the path to each file and each folder within the information package must be less than 180 characters. The path always contains the top-level folder as well. The `/` symbols must also be counted.
	The length of the name of a folder or file must be shortened if the path to it is more than 180 characters in length. The names in a path must be shortened until the length of the complete path is less than 180 characters.
	
	Example
	`SIP_20091220_EPA_hp/header/metadata.xml`
	This path consists of 39 characters.
	`SIP_20091220_SBF_hp/content/d00001245/p00123453.pdf`
	This path consists of 51 characters.
	`SIP_20091220_BFS_hp/content/orange_zettel/zettel_1.tif`
	This path consists of 54 characters.
		M

5.6 Specific rules for a SIP for GEVER

In addition to the general rules for the structure of a SIP, there are specific rules for GEVER-SIPs that apply only to SIPs from GEVER systems. The structure of a SIP from a GEVER system is shown in Fig. 9.

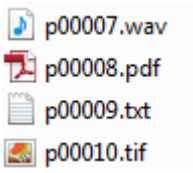
```
SIP_20090130_FSO_hpm/
      header/
            metadata.xml
            xsd/
                  arelda.xsd
                  …
      content/
            d00001/
                  p00001.pdf
                  p00002.txt
                  p00003.tif
                  d00002/
                        p00004.pdf
                  d00003/
                        p00005.pdf
            d00004/
                  p00006.pdf
                  p00007.tif
```

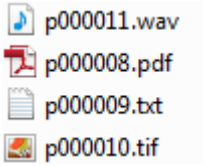
Fig. 9: Structure of a GEVER-SIP

ID	Description of requirement	M/O
S_5.6-1	It is recommended best practice for each dossier and subdossier in the submission to correspond to a folder in the `content/` folder or lower.	O

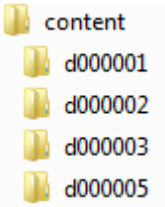
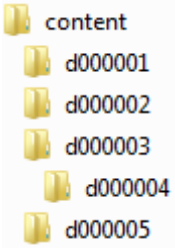
ID	Description of requirement	M/O
S_5.6-2	It is recommended best practice for the name of each folder that corresponds to a dossier or subdossier to start with the letter `d`. All folders are numbered with a consecutive number. Model: d n n n d n n n n n n d: is the abbreviation for (sub)dossier n: is an integer with a fixed number of digits for the consecutive number within the package. Unused places are filled using a 0. The maximum number of digits in the consecutive number is 6. The consecutive numbers (consisting of `d` and the integer) are unambiguous identifiers and are allocated only once within the SIP. **Example** A folder that corresponds to a dossier or subdossier is named as follows: d n n n n d0001 d0002 d0003 d0004 or d n n n n n n d000001 d000002 d000003 d000004	O

<table>
<tr><th>ID</th><th>Description of requirement</th><th>M/O</th></tr>
<tr><td>S_5.6-3</td><td>
Naming of files inside the <code>content/</code> folder:
It is recommended best practice for the name of each file to begin with the letter <code>p</code>. All files are numbered with a consecutive number. The original file names are to be entered in the metadata as described in requirement S_5.3-5 so that they are preserved.
Model:
<table>
<tr><td>p</td><td>n</td><td>n</td><td>n</td><td>n</td><td>.</td><td>[file extension]</td></tr>
</table>
<table>
<tr><td>p</td><td>n</td><td>n</td><td>n</td><td>n</td><td>n</td><td>n</td><td>.</td><td>[file extension]</td></tr>
</table>
p: is the abbreviation for (primary) file.
n: is an integer with a fixed number of digits for the consecutive number within the package. Unused places are filled using a 0. The maximum number of digits in the consecutive number is 6.
[file extension]: The file extension is part of the file name. It must also be indicated after the point.
The consecutive numbers (consisting of <code>p</code> and the integer) are unambiguous identifiers and are allocated only once within the SIP.

Example
<table>
<tr><td>p</td><td>n</td><td>n</td><td>n</td><td>n</td><td>n</td><td>.</td><td>[file extension]</td></tr>
</table>


or
<table>
<tr><td>p</td><td>n</td><td>n</td><td>n</td><td>n</td><td>n</td><td>n</td><td>.</td><td>[file extension]</td></tr>
</table>


</td><td>O</td></tr>
</table>

ID	Description of requirement	M/O
S_5.6-4	All folders that correspond to dossiers from the GEVER system should expediently be described on the top level in the `content/` folder. The classification system structure should not be reproduced in folders. **Example** Top level → folder for dossiers in the `content/` folder.  Further levels → folders for subdossiers  The folder `d000004` corresponds to a subdossier.	O

5.7 Specific rules for a SIP for FILES

In addition to the general rules for the structure of a SIP, there are specific rules for FILES-SIPs that apply only to SIPs from file archives or SIPs from relational databases.

The structure of a FILES-SIP is shown in Fig. 10.

```
SIP_20080130_FOC_hanspeter.meier/
      header/
            metadata.xml
            xsd/
                  arelda.xsd
      content/
            Meier_Art_Collection/
                  Introduction/
                        Documentation.txt
                  Early_Works/
                        Lion.tif
                        Red_Period/
                              Camel_large.tif
                              Camel_small.tif
                        Blue_Period/
                              The_Sea.tif
                              Dark_Blue_Period_December1867_April1868/
                                    The_Ocean.tif
                  Middle_Period_Works/
                        Cat.tif
                  Late_Works/
                        A_Reindeer.tif
                  Notebooks/
                        Notes_2000_2002.tif
```

Fig. 10: Structure of a FILES-SIP

ID	Description of requirement	M/O
S_5.7-1	The entire folder structure of the primary data in the submission is contained and reproduced in the `content/` folder. For this reason, the folders in `content/` do not necessarily correspond to dossiers and subdossiers. *Recommendation* *For reasons of clarity, it is recommended to reproduce the folders in a FILES-SIP that correspond to dossiers on a single hierarchical level.*	M
S_5.7-2	There are no restrictions on content or rules for the names of folders and files within the `content/` folder. The names of the folders and files can be chosen at will as long as they satisfy the requirements with regard to character set and path length.	M

ID	Description of requirement	M/O
S_5.7-3	It is the responsibility of the submitting authority to establish the mapping of the files in the package to their grouping in the dossiers in the `metadata.xml`. Technically, this is done via the `<dateiRef>`[2] in the `<ablieferung>` section. This references a `<datei>` element in the `<inhaltsverzeichnis>` via the id attribute. The `<datei>` element contains information about the file such as the name and checksum. The `<inhaltsverzeichnis>` lists all the folders and files in the submission. The path of a file is derived from the position in the folder structure in the table of contents. (See Fig. 11).	M

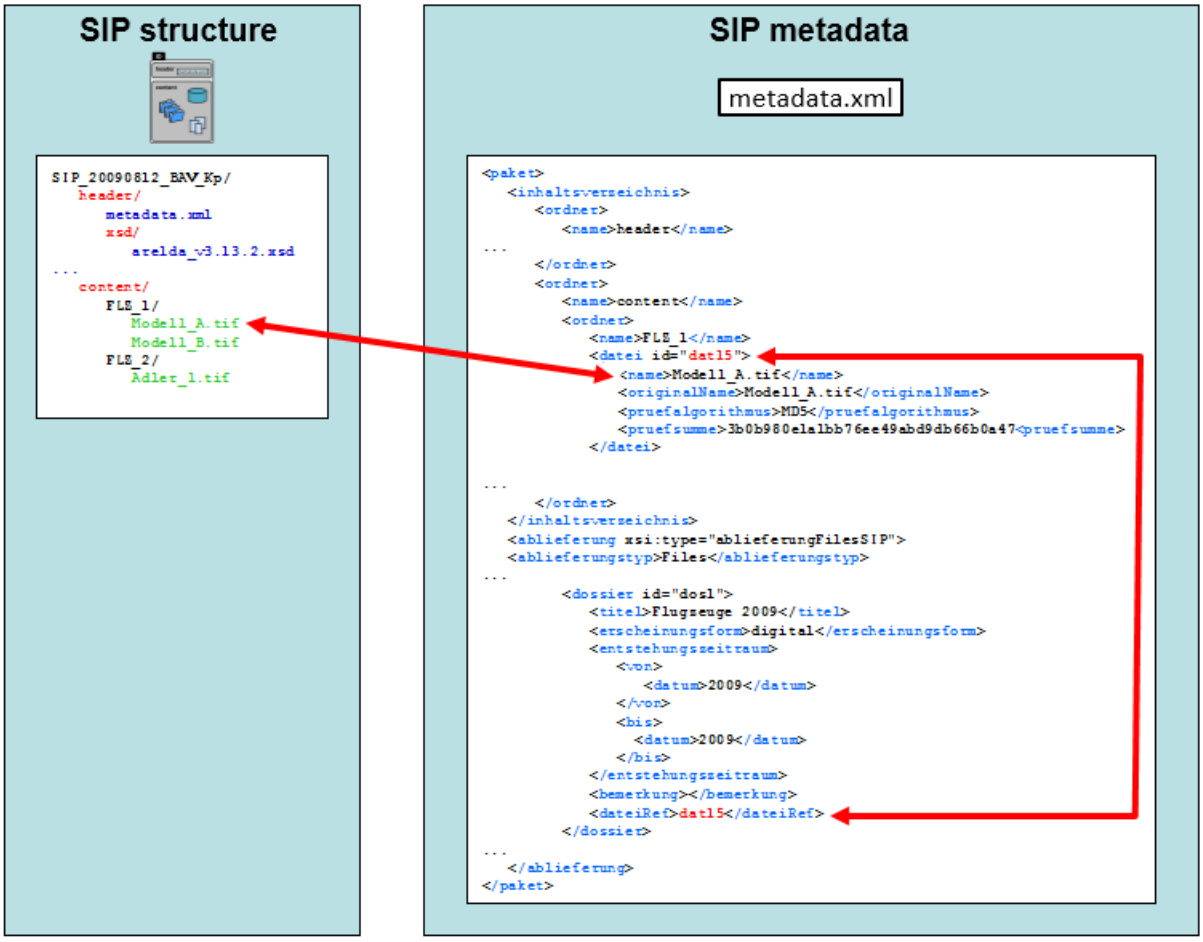


Fig. 11: Structure of a FILES-SIP – referencing of the file from the dossier to the file description in the table of contents

[2] Translation of the German terms used in this section: “dateiRef” = “fileRef”; “ablieferung” = “submission”; “datei” = “file”; “inhaltsverzeichnis” = “tableOfContents”.

5.8 Structure of a FILES-SIP with integrated documentation

A SIP from a submission with archive records from a relational database or specialist application is essentially a SIP with a SIP structure for FILES submissions. However, there are further specific rules for such SIPs. Above all, it is important for the system from which the data are drawn (database or specialist application) to be described in documentation.

The structure of such a SIP (in this case a SIP with the data from a relational database that was archived in SIARD format) can be seen in Fig. 12.

```
SIP_20070130_FSO_DB_Statistics_Traffic
      header/
            metadata.xml
            xsd/
                  arelda.xsd
                  …
      content/
            1_DOC/
                  Description_DB_traffic.pdf
                  Data_model.pdf
                  Data_listA.pdf
                  Data_list1.pdf
                  Data_list2.pdf
                  Data_list3.pdf
                  Training_documents_2000.tif
            2_DATA/
                  Database_Statistics_Traffic.siard
```

Fig. 12: Structure of a FILES-SIP with integrated documentation

ID	Description of requirement	M/O
S_5.8-1	The documentation in a FILES-SIP with integrated documentation must be stored in the `1_DOK/` folder.	M
S_5.8-2	The files in a FILES-SIP with integrated documentation (e.g. the data from an archived relational database) must be stored in a folder with the name `2_DATEN/`.	M
S_5.8-3	A FILES-SIP with archive records from a relational database that was archived in SIARD format therefore contains at least 1 dossier that corresponds to the folder `2_DATEN/`. **Example** In the case of FILES-SIPs with data from databases that were archived using SIARD, the folder `1_DOK/` contains the documentation while the folder `2_DATEN/` contains the SIARD extract (file with the extension .siard). 	M

6 Transfer

6.1 Format of the package container

ID	Description of requirement	M/O
T_6.1-1	In order to enable faster and easier handling of information packages, the SIP can be packed in a container (e.g. ZIP). The container format must be agreed between the submitting authority and the archive before transfer.	O

7 Changes since the previous version of the Specification

There is no earlier eCH version of the Specification.

8 Version and validity of the Specification

This version 1.0 of eCH-0160 is technically identical in terms of content with version 4.0 of the *Submission Information Package Specification* of the Swiss Federal Archives. When the latter was adopted for the eCH version, only those passages in the text that were specific to the SFA or the federal authorities were amended. For reasons of congruence with the numbering used by the SFA, the version number 4.0 was retained.[3]

The content of the Specification is periodically reviewed by the eCH digital archiving expert group and amended as necessary.

9 Change request process

Until further notice, the release cycle for new versions of this standard is annual. The Swiss Federal Archives act as the clearing house for change requests. These must be submitted informally to the SFA by e-mail (bundesarchiv@bar.admin.ch, with the comment "ChangeRequest SIP Specification"). The SFA carry out an initial triage. In particular, they contact the applicant if the change request appears unrealistic or less urgent.

The SFA publishes requests that have been reviewed on the eCH-Share platform of the digital archiving expert group. When a request is received that would result in a major release, i.e. the changes are not backwards compatible, the SFA will promptly convene a meeting of the archival submission interface subject group. Where necessary, the members of the group will seek the opinions of the manufacturers concerned.

[3] A detailed overview of the version history of the SFA Specification can be found in the document *Changes SIP Spezifikation, Data Dictionary und XSD*, available (in German) at http://www.bar.admin.ch/themen/00876/00877/01561/.

The subject group sifts and responds to change requests. All change requests and the reactions to them are documented. If the subject group sees a need for an update to the standard, it prepares the corresponding submissions to the expert group.

The schedule for the release cycle is as follows:

End of January	Deadline for submitting change requests
February to April	Discussion of change requests and drafting of the new version by the subject group
End of April	Dispatch of the new version including list of change requests, with comments, to the expert group
End of June	Approval of the new version by the expert group
Autumn	Publication of the new version by eCH

10 Exclusion of liability / notice regarding third-party rights

eCH standards which are made available to users by the **eCH** association (Verein **eCH**) for use free of charge, or which **eCH** references, only have the status of recommendations. The **eCH** association accepts no liability for decisions or measures taken by the user on the basis of these documents. It is the responsibility of the user to review the documents before using them and obtain advice where necessary. **eCH** standards cannot, nor are they intended to, replace technical, organisational or legal advice in specific cases.

Documents, processes, methods, products and standards referenced in **eCH** standards may be protected as trademarks or by copyright or patent law. It is the sole responsibility of the user to obtain any rights that may be necessary from the individuals and/or organisations that hold them.

Although the **eCH** association takes every care to draw up the **eCH** standards diligently, it can offer no warranty or guarantee that the information and documents made available are up to date, complete, correct or free from errors. The content of **eCH** standards may be amended at any time without notice.

To the extent permitted by law, no liability is accepted in respect of loss or damage incurred by the user as the result of using the **eCH** standards.

11 Copyright

Any person who draws up **eCH** standards retains intellectual property rights in respect thereof. However, such person undertakes to make his/her relevant intellectual property or rights in respect of the intellectual property of others available, to the extent possible, to the respective expert groups and the **eCH** association free of charge for unrestricted use and further development within the context of the association's purpose.

The standards drawn up by the expert groups may be used, disseminated and further developed free of charge and without restriction, provided the authors at **eCH** are cited.

eCH standards are fully documented and free from licence and/or patent restrictions. The associated documentation may be obtained free of charge.

These provisions apply solely to the standards drawn up by **eCH**, and not to standards or products of third parties that are referred to in the **eCH** standards. The standards contain corresponding references to the rights of third parties.

Appendix A – Participation and review

Marguérite Bos, Swiss Federal Archives

Georg Büchler, KOST

Anouk Dunant Gonzenbach, Geneva State Archives

David Gubler, Fachlabor Gubler AG

Martin Kaiser, KOST

Markus Lischer, Lucerne State Archives

Urs Meyer, Swiss Federal Archives

Ursina Rodenkirch-Brändle, Graubünden State Archives

Stefan Ryter, Bern State Archives

The English translation has been sponsored by scope solutions AG

Appendix B – Abbreviations and glossary

Term	Description
AIP	Archival Information Package. AIPs result from SIPs during the process of archiving digital documents. They represent the form of information packages in which digital documents are stored in the digital repository.
Archivable	Archivable file formats have to meet the archive's requirements for the preservation of digital documents.
Archival process	An archival process takes place in the archive. It comprises and documents activities related to ingest or preservation.
(Of) archival value	Documents that are defined as having archival value encompass documents that are of legal or administrative importance or contain valuable information (i.e. documents that are valuable from a historical, social or cultural point of view).
Archive	1. Institution or body responsible for cataloguing, keeping and preserving archive records and making them available. 2. Archived documents of an organisation. 3. Building or institution that was constructed or established for the purpose of archiving documents.
Archive records	Refers to documents that have been accepted by the archive for safekeeping, or that are independently archived by other bodies in accordance with the same principles.
Closure period	Access to archive records is regulated by closure periods. Documents that are still subject to a closure period may only be viewed if this has been approved in an official approval procedure. (This does not apply to the submitting authority itself.)
Digital archive	Refers to the institution that appraises, secures and describes digital archive records and makes them available.
Digital repository	Place where digital archive records are kept.
DIP	Dissemination Information Package. A DIP is a container for dossiers that are requested by a user via an ordering procedure.
Documents	Documents are all recorded information, irrespective of the medium, that is received or produced in the fulfilment of public duties, as well as all finding aids and supplementary data that are required in order to understand and use this information.
Dossier	This term refers to all documents relating to a specific business matter. A dossier basically corresponds to a business matter. However, by combining similar business matters or dividing dossiers into subdossiers, this basic structure can be adapted to meet the corresponding needs. The compilation of dossiers is carried out on the basis of the classification system.
File archive	Refers primarily to a quantity of files. Within the scope of this Specification, it is used for submissions containing files that are submitted without a classification system in the sense of records management using a GEVER system. However, the files may well be organised using another management system.

Metadata	Metadata can be described as “information about primary data” (data about data), since they have a descriptive nature.
OAIS	Open Archival Information System. A reference model approved in accordance with ISO 14721, OAIS describes an archive as an organisation in which people and systems work together to preserve information and make it available to a designated community.
Primary data	Primary data are data that are solely created directly by each records creator.
Records creator	Refers to the authority or organisational unit that created and managed the documents.
Records and process management (GEVER)	Refers to all activities and rules for the planning, steering and control, as well as verification, of business matters. GEVER encompasses the management of documents and dossiers, as well as business-related process management. It secures the effective and efficient business activity of administrative units.
SFA	Swiss Federal Archives
SIP	Submission Information Package: SIPs are information packages that are submitted to the archive by the records-creating authorities. They contain digital documents (primary data and metadata).
Submission	Refers both to the process by which an authority transfers documents to the archive, and to the overall volume of documents transferred by an authority during a submission.
Submitting authority	Refers to the authority or organisational unit that submits documents to the archive. It is often (but not necessarily) identical to the records creator.

Appendix C – Validity of chapters

Legend of colours used

	is to be used for the GEVER submission category
	is to be used for the FILES submission category
	is to be used for the FILES with integrated documentation submission category

X	All the requirements of this chapter are to be applied
P	Only the requirements that explicitly refer to GEVER or FILES are to be applied.

		Valid for submission categories		
Chapter		**GEVER**	**FILES**	**FILES with integrated documentation**
1	**INTRODUCTION**			
1.1	Status of the document	X	X	X
1.2	Objectives	X	X	X
1.3	Structure of the document	X	X	X
1.3.1	Structure of chapters	X	X	X
1.3.2	ID for requirements	X	X	X
1.3.3	Distinction between mandatory and optional requirements	X	X	X
1.3.4	Notation of folders, files and folder structures	X	X	X
1.4	Requirement for further detail	X	X	X
2	**OVERVIEW**			
2.1	Scope	X	X	X
2.1.1	Definitions	X	X	X
2.1.2	Requirements for each submission type	X	X	X
2.2	Area of application	X	X	X
2.2.1	Process	X	X	X
2.3	The submission package – SIP	X	X	X
2.4	Contents	X	X	X
2.5	Relationship between submission and package	X	X	X
2.6	Security settings for files in the package	X	X	X
3	**PRIMARY DATA**			
3.1	Basics	X	X	X
3.2	Formats	X	X	X
4	**METADATA**			
4.1	General requirements	X	X	X
4.2	Conceptual data model	X	X	X
4.3	Data model for the GEVER submission type	X		
4.4	Data model for the FILES submission type		X	X

Chapter		Valid for submission categories		
		GEVER	**FILES**	**FILES with integrated documentation**
4.5	Data dictionary	X	X	X
4.6	XSD	X	X	X
4.7	Structure of the table of contents in the metadata.xml	X	X	X
4.8	Mapping metadata from the originating system onto SIP metadata	P	P	P
4.9	Metadata on closure periods	X	X	X
4.10	Metadata on the time periods of the dossiers	X	X	X
4.11	Metadata on the checksums of the files	X	X	X
4.12	Referencing the attribution to dossiers in the metadata	X	X	X
5	**STRUCTURE OF THE SIP**			
5.1	Framework requirements and limitations on size of the package	X	X	X
5.2	Framework requirements and limitations on number of files in the package	X	X	X
5.3	Permitted characters for folder and file names	X	X	X
5.4	Structure of a SIP	X	X	X
5.5	Path lengths and hierarchies in the SIP	X	X	X
5.6	Specific rules for a SIP for GEVER	X		
5.7	Specific rules for a SIP for FILES		X	X
5.8	Structure of a FILES-SIP with integrated documentation			X
6	**TRANSFER**			
6.1	Format of the package container	X	X	X

Appendix D – Excerpt of the mapping of SIP metadata to the GEVER system

The table below shows an **example** of the form that mapping between the metadata for the SIP and the metadata of the originating system (e.g. a GEVER system) for the documents to be archived / their metadata can take. The data in the table are examples. In the example below, only the first four metadata of the *document* entity are listed. This mapping must be created for all the relevant entities in a submission.

Entity – DOCUMENT – GEVER-SIP							
XSD						**GEVER system**	
Element name	**Definition**	**Mandatory (m) or optional (o)**	**Data type**	**Value range**	**Occurs more than once**	**Mapping**	**To field(s)**
id	unambiguous ID for whole package (primary key)	m	text		-	yes	COO-Id
title	Brief description of the subject matter dealt with in the document	m	text	length 3	-	yes	Property (field XXXX)
author	Author of a document	o	text	length 2	0..n	yes	Administrator responsible (field XXXX)
form	Indication of whether the document is digital or non-digital (paper, audiovisual) at the time of submission. A document can only be assigned one of the two forms (digital or non-digital). Documents that consisted of a digital and a non-digital component before submission to the archive must be submitted as two separate documents.	m	enumeration	digital, non-digital		fixed	fixed entry: digital Reason: all documents are digital (no analogue components)
document type	Appearance of the document that is to be preserved.	o	text	length 3	-	no	Reason: cannot be specifically indicated for the documents
...	...	...	...	...	...	...	...

Appendix E – Character sets

This appendix defines the permitted characters in file and folder names in the SIP and contains normalisation tables for some frequently encountered character sets. This prevents incompatibilities that can arise due to different character sets or special characters in various application environments. The original file names must be recorded in the metadata so that they are not lost.

For reasons of simplicity the text below refers only to file names. However, both file and folder names are meant.

E.1: The code page problem

A code page is a table containing a character coding of various characters. Characters displayed on computers are coded in accordance with a code page during programming. There are normally 2^8 characters for the coding, giving a total of 256. In the table, these characters are shown from 0 to 255. The first half (characters 1 to 126) is identical with US-ASCII in almost all code pages.

A different code page can be installed on each computer. Regions and countries have code pages adapted to their languages. When data are exchanged across linguistic frontiers that use different code pages, texts with umlauts or accents cause a data scramble because they are not displayed correctly. This can also happen during exchanges between computers using different operating systems. What applies to the contents of a file also applies to the file names.

For instance, in western European countries Windows uses the 1252 and Unicode code pages (UTF-16 encoding), while Linux and Unix systems tend to use the ISO-8859-1 standard.

The following examples show what happens when file names are coded differently and are displayed in changed environments:

File name	Created in environment with	Displayed in environment with	Result	Comments
René Zürcher	UTF-8	ISO-8859-1	RenÃ© ZÃ¼rcher	
René Zürcher	ISO-8859-1	ISO-8859-5	Renщ Zќrcher	
René Zürcher	ISO-8859-1	UTF-8	Ren Zrcher Ren? Z?rcher	Illegal UTF-8 sequence. Characters are often displayed as ? or not at all

E.2: File name normalisations

To avoid this problem when working with SIPs in environments with different code pages, file names need to be normalised. The basis used is the US-ASCII character set (E.2.1). With some additional restrictions for file names (E.2.2) imposed by operating systems, normalisation tables for converting file names are proposed below (E.2.3ff).

E.2.1 US-ASCII as basic character set

Both UTF-8 and ISO-8859 are supersets of US-ASCII, i.e. each US-ASCII character sequence is also a valid UTF-8 character sequence or ISO-8859 character sequence. Accordingly, US-ASCII character sequences are always interpreted and displayed in the same way, irrespective of whether the software was designed or configured for US-ASCII, UTF-8 or ISO-8859 processing. (The term “software” is used here to mean not just applications such as editors, web browsers etc. but also operating systems and file system drivers.)

E.2.2 Permitted and reserved (non-permitted) characters in file names

Overview of restrictions and special meanings of characters in operating systems and language definitions (for a more extensive overview see the Wikipedia entry “Filename” (http://en.wikipedia.org/wiki/Filename), which contains detailed descriptions and further references):

System	Upper case / lower case	Reserved characters	Comments
WinXP NTFS	optional	< > : " / \ | ? *	
WinXP HPFS	not relevant	< > : " / \ | ? *	
Unix / Linux	relevant	/	
Mac OS HFS+	not relevant	: / (in the Unix layer Mac OS X)	
XML	relevant	< > & ' "	
Uniform Resource Identifiers (URI)	relevant	: / ? # [ ] @ (generic component delimiters) ! $ & ' ( ) * + , ; = (subcomponent delimiters)	not considered

URIs to limit the permitted characters from the US-ASCII character set are not considered for the normalisation of file names. If a file name contains reserved characters from a URI and this is essential to further processing within a URI, the file name is adapted accordingly using percent encoding (RFC 3986 Uniform Resource Identifier (URI): Generic Syntax).

E.2.2.1 Permitted characters

The permitted characters are found only in the range of the US-ASCII character set, i.e. between 0x20 and 0x7E. However, not all US-ASCII characters are permitted. The following table lists the permitted characters.

US-ASCII / ISO-8859-1	Unicode	Unicode Description	Symbol
0x20	U+0020	Space	*SP*
0x21	U+0021	Exclamation mark	!
0x23	U+0023	Number sign	#
0x24	U+0024	Dollar sign	$
0x25	U+0025	Percent sign	%
0x28	U+0028	Left parenthesis	(
0x29	U+0029	Right parenthesis	)
0x2B	U+002B	Plus sign	+
0x2C	U+002C	Comma	,
0x2D	U+002D	Hyphen/Minus sign	-
0x2E	U+002E	Full stop	.
0x30	U+0030	Digit Zero	0
	...	...	...
0x39	U+0039	Digit Nine	9
0x3D	U+003D	Equal sign	=
0x40	U+0040	At sign	@
0x41	U+0041	Latin Capital letter A	A
	…	…	…
0x5A	U+005A	Latin Capital letter Z	Z
0x5B	U+005B	Left Square Bracket	[
0x5D	U+005D	Right Square Bracket	]
0x5F	U+005F	Low line	_
0x61	U+0061	Latin Small Letter A	a
	…	…	…
0x7A	U+007A	Latin Small Letter Z	z
0x7B	U+007B	Left Curly Bracket	{
0x7D	U+007D	Right Curly Bracket	}
0x7E	U+007E	Tilde	~

E.2.2.2 Non-permitted characters

Characters that, by definition, are not permitted for file names in operating systems and file systems as well as characters that can cause problems during data transfer between different systems (all characters outside the US-ASCII range) are not permitted.

US-ASCII / ISO-8859-1	Unicode	Unicode Description	Symbol
0x00	U+0000	*Control Characters*	
...	...		
0x1F	U+001F	*Control Characters*	
0x22	U+0022	Quotation mark	"
0x26	U+0026	Ampersand	&
0x27	U+0027	Apostrophe	'
0x2A	U+002A	Asterisk	*
0x2F	U+002F	Slash	/
0x3A	U+003A	Colon	:
0x3B	U+003B	Semicolon	;
0x3C	U+003C	Less-than sign	<
0x3E	U+003E	Greater-than sign	>
0x3F	U+003F	Question mark	?
0x5C	U+005C	Backslash	\
0x5E	U+005E	Circumflex accent	^

US-ASCII / ISO-8859-1	Unicode	Unicode Description	Symbol
0x60	U+0060	Grave accent	`
0x7C	U+007C	Vertical bar	|
	U+007F	Delete	
...	...	...	

Character codes that lie above 0x7F or U+007F are not permitted.

E.2.3 Normalisation of US-ASCII in the range 0x20–0x7E

Control characters (0x00..0x1F, 0x7F) are illegal in file names and are not considered further for the purposes of normalisation. If they occur, they are to be eliminated and an error message is to be generated. Characters not permitted in file names (see chapter E.2.2.2) are shown as "_" during normalisation; permitted characters remain unchanged:

From...				To...		
US-ASCII / ISO-8859-1	Unicode	Unicode Description	Symbol	Unicode	Unicode Description	Symbols
0x20	U+0020	Space	*SP*	U+005F	Space	*SP*
0x21	U+0021	Exclamation mark	!	U+0021	Exclamation mark	!
0x22	U+0022	Quotation mark	"	U+005F	Low Line	_
0x23	U+0023	Number sign	#	U+0023	Number sign	#
0x24	U+0024	Dollar sign	$	U+0024	Dollar sign	$
0x25	U+0025	Percent sign	%	U+0025	Percent sign	%
0x26	U+0026	Ampersand	&	U+005F	Low Line	_
0x27	U+0027	Apostrophe	'	U+005F	Low Line	_
0x28	U+0028	Left parenthesis	(	U+0028	Left parenthesis	(
0x29	U+0029	Right parenthesis	)	U+0029	Right parenthesis	)
0x2A	U+002A	Asterisk	*	U+005F	Low Line	_
0x2B	U+002B	Plus sign	+	U+002B	Plus sign	+
0x2C	U+002C	Comma	,	U+002C	Comma	,
0x2D	U+002D	Hyphen/Minus sign	-	U+002D	Hyphen/Minus sign	-
0x2E	U+002E	Full stop	.	U+002E	Full stop	.
0x2F	U+002F	Slash	/	U+005F	Low Line	_
0x30	U+0030	Digit Zero	0	U+0030	Digit Zero	0
	...	...	...	...	...	...
0x39	U+0039	Digit Nine	9	U+0039	Digit Nine	9
0x3A	U+003A	Colon	:	U+005F	Low Line	_
0x3B	U+003B	Semicolon	;	U+005F	Low Line	_
0x3C	U+003C	Less-than sign	<	U+005F	Low Line	_
0x3D	U+003D	Equal sign	=	U+003D	Equal sign	=
0x3E	U+003E	Greater-than sign	>	U+005F	Low Line	_
0x3F	U+003F	Question mark	?	U+005F	Low Line	_
0x40	U+0040	At sign	@	U+0040	At sign	@
0x41	U+0041	Latin Capital letter A	A	U+0041	Latin Capital letter A	A
	…	…	…	…	…	…
0x5A	U+005A	Latin Capital letter Z	Z	U+005A	Latin Capital letter Z	Z
0x5B	U+005B	Left Square Bracket	[	U+005B	Left Square Bracket	[
0x5C	U+005C	Backslash	\	U+005F	Low Line	_
0x5D	U+005D	Right Square Bracket	]	U+005D	Right Square Bracket	]
0x5E	U+005E	Circumflex accent	^	U+005F	Low Line	_
0x5F	U+005F	Low line	_	U+005F	Low Line	_
0x60	U+0060	Grave accent	`	U+005F	Low Line	_
	U+0061	Latin Small Letter A	a	U+0061	Latin Small Letter A	a
	…	…	…	…	…	…
	U+007A	Latin Small Letter Z	z	U+007A	Latin Small Letter Z	z
0x7B	U+007B	Left Curly Bracket	{	U+007B	Left Curly Bracket	{
0x7C	U+007C	Vertical bar	|	U+005F	Low Line	_

From...				To...		
US-ASCII / ISO-8859-1	Unicode	Unicode Description	Symbol	Unicode	Unicode Description	Symbols
0x7D	U+007D	Right Curly Bracket	}	U+007D	Right Curly Bracket	}
0x7E	U+007E	Tilde	~	U+007E	Tilde	~

E.2.4 Normalisation of code page 1252, ISO-8859 and Unicode in the range 0x80–0x9F (U+0080 – U+009F)

Code page 1252 is designated *Western European*. It deviates from ISO-8859-1 in the range 0x80–9F, the 32 positions of which contain 27 characters that can be displayed, including those added in ISO-8859-15 and some that are needed for better typography. In the case of characters that cannot be displayed, the *Symbol* field remains empty in the table below. The differences between all these codings and the general lack of consistency in support for different character sets are a frequent source of interoperability problems.

The characters (0x80..0x9F and U+0080..U+009F, respectively) are control characters in ISO-8859 and Unicode. They are illegal in file names and are not considered further for the purposes of normalisation. If they occur, they are to be eliminated and an error message is to be generated.

The following table contains the conversion of the characters concerned into *visually similar* US-ASCII characters:

From...		To...	
CP-1252	Symbol	US-ASCII	Symbols
0x80	€	0x45 0x3D	E=
0x81			
0x82	‚	0x27	'
0x83	ƒ	0x66	f
0x84	„	0x27	'
0x85	…	0x2E 0x2E 0x2E	...
0x86	†	0x5F	_
0x87	‡	0x5F	_
0x88	ˆ	0x5F	_
0x89	‰	0x25 0x30	%0
0x8A	Š	0x53	S
0x8B	‹	0x27	'
0x8C	Œ	0x4F 0x45	OE
0x8D		0x5F	_
0x8E	Ž	0x5A	Z
0x8F		0x5F	_
0x90		0x5F	_
0x91	‘	0x27	'
0x92	’	0x27	'
0x93	“	0x27	'
0x94	”	0x27	'
0x95	•	0x5F	_
0x96	–	0x2D 0x2D	--
0x97	—	0x2D 0x2D 0x2D	---
0x98	˜	0x7E	~
0x99	™	0x54 0x4D	TM
0x9A	š	0x73	s
0x9B	›	0x27	'
0x9C	œ	0x6F 0x65	oe
0x9D		0x5F	_

From...		To...	
CP-1252	Symbol	US-ASCII	Symbols
0x9E	ž	0x7A	z
0x9F	Ÿ	0x59	Y

E.2.5 Normalisation of code page 1252, ISO-8859 and Unicode in the range 0xA0–0xFF (U+00A0 – U+00FF)

These and the following mappings translate characters from UTF-8 (part) and ISO-8859 that are not in the US-ASCII range to ***visually similar*** US-ASCII characters.

The relevant and permitted code pages of the ISO-8859 standard are:

Code page 1 Latin-1, Western European

Code page 15 Latin-9, Western European

The following table contains the mapping of ISO-8859-1 characters and Unicode characters outside the US-ASCII range (0xA0 – 0xFF) to visually similar US-ASCII characters and character sequences, taking account of the restrictions in section E.2.2.1 above.

From...				To...	
ISO-8859-1	Unicode	Unicode Description	Symbol	US-ASCII	Symbols
0XA0	U+00A0	Non-breaking space		0x20	SP
0xA1	U+00A1	Inverted Exclamation Mark	¡	0x5F	_
0xA2	U+00A2	Cent sign	¢	0x63	c
0xA3	U+00A3	Pound sign	£	0x4C 0x3D	L=
0xA4	U+00A4	Currency sign	¤	0x49 0x3D	I=
0xA5	U+00A5	Yen sign	¥	0x59 0x3D	Y=
0xA6	U+00A6	Broken bar	¦	0x5F	_
0xA7	U+00A7	Section sign	§	0x53 0x53	SS
0xA8	U+00A8	Diaeresis	¨	0x5F	_
0xA9	U+00A9	Copyright sign	©	0x28 0x63 0x29	(c)
0xAA	U+00AA	Feminine Ordinal Indicator	ª	0x61	a
0xAB	U+00AB	Left-pointing double angle quotation mark	«	0x5F	_
0xAC	U+00AC	Not sign		0x5F	_
0xAD	U+00AD	Soft hyphen		0x5F	_
0xAE	U+00AE	Registered sign	®	0x28 0x72 0x29	(r)
0xAF	U+00AF	Macron	¯	0x5F	_
0xB0	U+00B0	Degree symbol	°	0x64 0x65 0x67	deg
0xB1	U+00B1	Plus-minus sign	±	0x2B 0x2D	+-
0xB2	U+00B2	Superscript two	²	0x32	2
0xB3	U+00B3	Superscript three	³	0x33	3
0xB4	U+00B4	Acute accent	´	0x5F	_
0xB5	U+00B5	Micro sign	µ	0x75	u
0xB6	U+00B6	Pilcrow sign	¶	0x50	P
0xB7	U+00B7	Middle dot	·	0x2E	.
0xB8	U+00B8	Cedilla	¸	0x2C	,
0xB9	U+00B9	Superscript one	¹	0x31	1
0xBA	U+00BA	Masculine ordinal indicator	º	0x6F	o
0xBB	U+00BB	Right-pointing double-angle quotation mark	»	0x5F	_
0xBC	U+00BC	Vulgar fraction one quarter	¼	0x5F	_
0xBD	U+00BD	Vulgar fraction one half	½	0x5F	_
0xBE	U+00BE	Vulgar fraction three quarters	¾	0x5F	_
0xBF	U+00BF	Inverted Question Mark	¿	0x5F	_
0xC0	U+00C0	Latin Capital Letter A with grave	À	0x41	A

From...				To...	
ISO-8859-1	**Unicode**	**Unicode Description**	**Symbol**	**US-ASCII**	**Symbols**
0xC1	U+00C1	Latin Capital letter A with acute	Á	0x41	A
0xC2	U+00C2	Latin Capital letter A with circumflex	Â	0x41	A
0xC3	U+00C3	Latin Capital letter A with tilde	Ã	0x41	A
0xC4	U+00C4	Latin Capital letter A with diaeresis	Ä	0x41 0x65	Ae
0xC5	U+00C5	Latin Capital letter A with ring above	Å	0x41	A
0xC6	U+00C6	Latin Capital letter AE	Æ	0x41 0x65	Ae
0xC7	U+00C7	Latin Capital letter C with cedilla	Ç	0x43	C
0xC8	U+00C8	Latin Capital letter E with grave	È	0x45	E
0xC9	U+00C9	Latin Capital letter E with acute	É	0x45	E
0xCA	U+00CA	Latin Capital letter E with circumflex	Ê	0x45	E
0xCB	U+00CB	Latin Capital letter E with diaeresis	Ë	0x45	E
0xCC	U+00CC	Latin Capital letter I with grave	Ì	0x49	I
0xCD	U+00CD	Latin Capital letter I with acute	Í	0x49	I
0xCE	U+00CE	Latin Capital letter I with circumflex	Î	0x49	I
0xCF	U+00CF	Latin Capital letter I with diaeresis	Ï	0x49	I
0xD0	U+00D0	Latin Capital letter Eth	Ð	0x44	D
0xD1	U+00D1	Latin Capital letter N with tilde	Ñ	0x4E	N
0xD2	U+00D2	Latin Capital letter O with grave	Ò	0x4F	O
0xD3	U+00D3	Latin Capital letter O with acute	Ó	0x4F	O
0xD4	U+00D4	Latin Capital letter O with circumflex	Ô	0x4F	O
0xD5	U+00D5	Latin Capital letter O with tilde	Õ	0x4F	O
0xD6	U+00D6	Latin Capital letter O with diaeresis	Ö	0x4F 0x65	Oe
0xD7	U+00D7	Multiplication sign	×	0x78	x
0xD8	U+00D8	Latin Capital letter O with stroke	Ø	0x4F	O
0xD9	U+00D9	Latin Capital letter U with grave	Ù	0x55	U
0xDA	U+00DA	Latin Capital letter U with acute	Ú	0x55	U
0xDB	U+00DB	Latin Capital Letter U with circumflex	Û	0x55	U
0xDC	U+00DC	Latin Capital Letter U with diaeresis	Ü	0x55 0x65	Ue
0xDD	U+00DD	Latin Capital Letter Y with acute	Ý	0x59	Y
0xDE	U+00DE	Latin Capital Letter Thorn	Þ	0x54 0x68	Th
0xDF	U+00DF	Latin Small Letter sharp S	ß	0x73 0x73	ss
0xE0	U+00E0	Latin Small Letter A with grave	à	0x61	a
0xE1	U+00E1	Latin Small Letter A with acute	á	0x61	a
0xE2	U+00E2	Latin Small Letter A with circumflex	â	0x61	a
0xE3	U+00E3	Latin Small Letter A with tilde	ã	0x61	a
0xE4	U+00E4	Latin Small Letter A with diaeresis	ä	0x61 0x65	ae
0xE5	U+00E5	Latin Small Letter A with ring above	å	0x61	a
0xE6	U+00E6	Latin Small Letter AE	æ	0x61 0x65	ae
0xE7	U+00E7	Latin Small Letter C with cedilla	ç	0x63	c
0xE8	U+00E8	Latin Small Letter E with grave	è	0x65	e
0xE9	U+00E9	Latin Small Letter E with acute	é	0x65	e
0xEA	U+00EA	Latin Small Letter E with circumflex	ê	0x65	e
0xEB	U+00EB	Latin Small Letter E with diaeresis	ë	0x65	e
0xEC	U+00EC	Latin Small Letter I with grave	ì	0x69	i
0xED	U+00ED	Latin Small Letter I with acute	í	0x69	i
0xEE	U+00EE	Latin Small Letter I with circumflex	î	0x69	i
0xEF	U+00EF	Latin Small Letter I with diaeresis	ï	0x69	i
0xF0	U+00F0	Latin Small Letter Eth	ð	0x64	d
0xF1	U+00F1	Latin Small Letter N with tilde	ñ	0x6E	n
0xF2	U+00F2	Latin Small Letter O with grave	ò	0x6F	o
0xF3	U+00F3	Latin Small Letter O with acute	ó	0x6F	o
0xF4	U+00F4	Latin Small Letter O with circumflex	ô	0x6F	o
0xF5	U+00F5	Latin Small Letter O with tilde	õ	0x6F	o
0xF6	U+00F6	Latin Small Letter O with diaeresis	ö	0x6F	oe
0xF7	U+00F7	Division sign	÷	0x5F	_
0xF8	U+00F8	Latin Small Letter O with stroke	ø	0x6F	o
0xF9	U+00F9	Latin Small Letter U with grave	ù	0x75	u

From...				To...	
ISO-8859-1	**Unicode**	**Unicode Description**	**Symbol**	**US-ASCII**	**Symbols**
0xFA	U+00FA	Latin Small Letter U with acute	ú	0x75	u
0xFB	U+00FB	Latin Small Letter U with circumflex	û	0x75	u
0xFC	U+00FC	Latin Small Letter U with diaeresis	ü	0x75 0x65	ue
0xFD	U+00FD	Latin Small Letter Y with acute	ý	0x79	y
0xFE	U+00FE	Latin Small Letter Thorn	þ	0x74 0x68	th
0xFF	U+00FF	Latin Small Letter Y with diaeresis	ÿ	0x79	y

The procedure for ISO-8859-15 characters that differ from ISO-8859-1 is the same as above.

E.2.6 Normalisation of Unicode starting from character U+0100

If characters occur in file names that are Unicodes U+0100 or higher, these are all to be converted into permitted US-ASCII characters or character sequences using the same procedure as above.