



**UN/CEFACT**

Simple, Transparent and Effective Processes  
for Global Commerce

**BUSINESS REQUIREMENTS SPECIFICATION  
(BRS)**

**Business domain: Archiving and records management**

**Business process: Transfer of digital records**

**Document identification:**

**Title: Record Exchange Standard**

**Trade Facilitation and Business Process Working Group: 19  
eGovernment**

**Version: 1.0**

**Release: 1**

**Date: 23 June 2008**

## Document Change History Log

Date of change	Version	Paragraph changed	Summary of changes
20070903	0.1		Initial version
20070906	0.2		Revised a/c initial comments
20071012	0.3.1		Revised a/c Stockholm meeting
20071025	0.3.2		Changed records to digital objects, restructured 1, 3, & 4. Added business rules
20071114	0.3.3		Allowed either party to initiate and terminate the relocation. Changed terminology.
20071218	0.3.4		Changed terminology.
2008	0.4		Revised after public comment period
20080527	0.4.2		Further revisions
20080530	0.4.3		Further revisions
20080603	0.5		Version for review
20080623	1.0		Final version

## Table of Contents

<b>1</b>	<b>PREAMBLE</b> .....	<b>5</b>
<b>2</b>	<b>REFERENCES</b> .....	<b>6</b>
<b>3</b>	<b>OBJECTIVE</b> .....	<b>7</b>
3.1	BACKGROUND .....	7
3.2	BENEFITS.....	8
<b>4</b>	<b>SCOPE</b> .....	<b>9</b>
<b>5</b>	<b>BUSINESS REQUIREMENTS</b> .....	<b>13</b>
5.1	BUSINESS PROCESS ELABORATION.....	13
5.1.1	<i>Transfer session for digital records</i> .....	13
5.1.2	<i>Example of a normal transfer session</i> .....	16
5.1.3	<i>Records and SIPs</i> .....	18
5.2	INFORMATION FLOW DEFINITION .....	19
5.2.1	<i>Sequence diagrams</i> .....	19
5.3	INFORMATION MODEL DEFINITION.....	27
5.3.1	<i>Message class</i> .....	29
5.3.2	<i>Manifest Proposal class</i> .....	31
5.3.3	<i>SIP class</i> .....	31
5.3.4	<i>Transfer Session Completed class</i> .....	31
5.3.5	<i>Status class</i> .....	32
5.3.6	<i>Manifest Agreement class</i> .....	32
5.3.7	<i>Final Status class</i> .....	32
5.3.8	<i>Final Status Acknowledgement class</i> .....	33
5.3.9	<i>Reject Transfer Session class</i> .....	33
5.3.10	<i>Error class</i> .....	33
5.3.11	<i>Record Status class</i> .....	34
5.3.12	<i>SIP Status class</i> .....	34
5.3.13	<i>Component class</i> .....	35
5.3.14	<i>Metadata Set class</i> .....	35
5.3.15	<i>Transfer Metadata Set class</i> .....	36
5.3.16	<i>Event History class</i> .....	36
5.3.17	<i>Event Plan class</i> .....	37
5.3.18	<i>Externally Defined Metadata class</i> .....	37
5.3.19	<i>Relation class</i> .....	38
5.3.20	<i>Proposed Record class</i> .....	38
5.3.21	<i>Proposed SIP class</i> .....	39
5.3.22	<i>Representation class</i> .....	39
5.3.23	<i>Digital Representation class</i> .....	39
5.3.24	<i>Physical Representation class</i> .....	40
5.3.25	<i>Included Content class</i> .....	40
5.3.26	<i>Referenced Content Class</i> .....	41
<b>6</b>	<b>BUSINESS RULES</b> .....	<b>41</b>
6.1	DEFINITION OF TERMS .....	44



# 1 Preamble

This Business Requirements Specification (BRS) describes a portion of the process of transferring of custody of digital records from one system to another in either the public or private sector. ‘Transfer of custody’ means the formal transfer of responsibility for the digital records, including responsibility for on-going preservation and access. Transfer can be of physical custody, legal custody, or both physical and legal. Examples of transfer of custody are:

- From the organization that created the records to an archive
- From a creating organization to its successor
- From an archive back to the creating organization
- From a creating organization to secondary storage and back again
- From one archive to another
- From one system to another within one organization

This specification is concerned with only one aspect of transfer of custody, specifically with the actual relocation of digital records from one records system to another. Aspects of transfer of custody not dealt with in this specification include, for example, the negotiation of the transfer, decisions about what to transfer and monitoring the quality of the transfer,

The specification describes transfer of custody from a business perspective. It is compatible with ISO 15489, OAIS, and MoReq2 (references for all of these are given in the next section).

The standard is expected to be of use to:

- Those responsible for records within an organization
- Those carrying out transfers of digital records
- Those designing or implementing records systems that must conform with this standard
- Those testing records systems for conformance to this standard.

The production of this standard was undertaken by a group of experts<sup>1</sup> of the CEN/ISSS/eBES/EG13 group<sup>2</sup> in coordination with the UN/CEFACT TBG 19 group<sup>3</sup>; both groups dealing with eGovernment business processes. It was supported by the ICA<sup>4</sup> and liased with ISO<sup>5</sup>. The project was approved by the UN/CEFACT

---

<sup>1</sup> Project editor: Andrew Waugh (Public Record Office Victoria, Australia); project coordinator: Olivier de Solan (Direction des Archives de France). Participants in the project meetings: Adeniran Ajibade (Ministry of Mines and Steel Development, Nigeria), Adam Arndt (IT- og Telestyrelsen, Denmark), Jan Bergström (SIE Gruppen, Sweden), Richard Blake (National Archives of the United Kingdom), Mats Berggren (Riksarkivet, Sweden), Jacques Bogaarts (Nationaal Archief, The Netherlands), Adrian Cunningham (National Archives of Australia), Magnus Geber (Riksarkivet, Sweden), Andrea Hänger (Bundesarchiv, Germany), Pekka Henttonen (Kansallisarkisto, Finland), Sören Lennartsson (Sveriges Kommuner och Landsting, Sweden), Gabriel Ramanantsoavina (Ministère du Budget, France), Alan Shipman (BSI, United Kingdom). The University of Michigan (Bentley Historical Library: Francis X. Blouin, Nancy Bartlett, Nancy Deromedi) kindly provided a collaborative workspace (Ctools) for the project.

<sup>2</sup> Sylvie Colas (Ministère du Budget, France), chair of EEG13; Alain Dechamps (CEN).

<sup>3</sup> Jean-Pierre Henninot (Ministère de l'économie, des finances et de l'emploi, France), chair of TBG 19.

<sup>4</sup> George Mackenzie (National Archives of Scotland), leader of priority area “Electronic records and automation”; Joan van Albada (ICA Secretary General); David Leitch (ICA Secretary General Designate); Marion Beyea (Provincial Archives of New Brunswick, Canada), chair of the committee on best practices and standards.

meeting in New Delhi in October 2006. The initial work built on activities undertaken by the Australasian Digital Recordkeeping Initiative<sup>6</sup>. An international survey on the required features of a record exchange standard was carried out in November 2006<sup>7</sup>. The first draft was completed in August 2007. A call for comments was issued in January 2008<sup>8</sup>, the results of which were taken into account to produce the definitive version of the specification.

## 2 References

- Date and Time Formats, W3C Note-datetime, September 1997  
<http://www.w3.org/TR/NOTE-datetime> visited 2 June 2008.
- Dublin Core Metadata Element Set, Version 1.1  
<http://dublincore.org/documents/dces/> visited 26 May 2008. Also available as ISO 15836:2003
- Encoded Archival Context (EAC) <http://www.iath.virginia.edu/eac/> visited 14 November 2007
- Encoded Archival Description (EAD) <http://www.loc.gov/ead/> visited 14 November 2007
- Information and Documentation - Records Management - Part 1: General, ISO/IEC 15489-1:2001
- Information and documentation - Records management processes - Metadata for records - Part 1: Principles, ISO 23081-1:2006
- Information and documentation - Records management processes - Metadata for records - Part 2: Conceptual and implementation issues, ISO 23081-2:2007
- Information technology – Multimedia framework (MPEG-21) – Part 2: Digital Item Declaration, ISO/IEC 21000-2:2005,  
[http://standards.iso.org/ittf/PubliclyAvailableStandards/c041112\\_ISO\\_IEC\\_21000-2\\_2005\(E\).zip](http://standards.iso.org/ittf/PubliclyAvailableStandards/c041112_ISO_IEC_21000-2_2005(E).zip) visited 14 November 2007
- ISAD (G): General International Standard Archival Description, Second edition, International Council on Archives, 1999, ISBN 0-9696035-5-X,  
<http://www.ica.org/en/node/30000> visited 14 November 2007.
- Metadata Encoding & Transmission Standard (METS)  
<http://www.loc.gov/standards/mets/> visited 14 November 2007.

---

<sup>5</sup> Catherine Dhérent (Bibliothèque nationale de France), chair of ISO/TC 46.

<sup>6</sup> <http://www.adri.gov.au/>

<sup>7</sup> Participants in the survey: Nancy Allard (National Archives and Records Administration, National Historical Publications and Records Commission, US), Euan Cochrane (Statistics New Zealand), Janine Delaney (Archives New Zealand), Nancy Deromedi (Bentley Historical Library, University of Michigan, US), Catherine Dhérent (Bibliothèque nationale de France), Christine Frodl (Deutsche Nationalbibliothek, Germany), Richard Gore (State Records Authority of New South Wales, Australia), Karen Horsfall (State Records of South Australia), Glen McAninch (Kentucky Department for Libraries and Archives, US), Grayham Mount (National Archives of Scotland), Susan Rigney (Queensland State Archives, Australia), Alan Shipman (BSI, United Kingdom), Bill Taylor (Department of Education Tasmania, Australia), Hanno Vares (Rahvusarhiiv, Estonia).

<sup>8</sup> Participants in the call for comments: Kuldar Aas (Rahvusarhiiv, Estonia), Françoise Banat-Berger (Direction des Archives de France), Sylvain Bellengier (Ministère du Budget, France), Stephen Clarke (Archives New Zealand), Sylvie Colas (Ministère du Budget, France), Catherine Dhérent (Bibliothèque nationale de France), Claude Huc (Centre national d'études spatiales, France), Natasha Khramtsovsky (Electronic Office Systems, Russia), Alan Shipman (BSI), Kenneth Thibodeau (National Records and Archives Administration, US), US group of respondents. The detailed report provided by the US National Records and Archives Administration was of great value for the review process.

- Model requirements for the management of electronic records (MoReq2 specification), Update and extension, 2008 <http://www.moreq2.eu/> visited 17 March 2008
- Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies, section 6.8, Base64 Content-Transfer-Encoding, IETF RFC 2045, N. Freed & N. Borenstein, November 1996, <http://www.ietf.org/rfc/rfc2045.txt?number=2045> visited 2 June 2008
- PREMIS, <http://www.loc.gov/standards/premis/> visited 14 November 2007
- Principles and Functional Requirements for Records in Electronic Office Environments, International Council on Archives, 2008.
- Producer-Archive Interface Methodology Abstract Standard (PAIMAS), Consultative Committee for Space Data Systems, CCSDS 651.0-B-1, Blue Book, May 2004, <http://public.ccsds.org/publications/archive/651x0b1.pdf> visited 3 December 2007. Also available as Space data and information transfer systems - Producer-Archive interface - Methodology abstract standard, ISO 20652:2006
- Reference Model for an Open Archival Information System (OAIS), Consultative Committee for Space Data Systems, CCSDS 650.0-B-1, Blue Book, January 2002, <http://public.ccsds.org/publications/archive/650x0b1.pdf> visited 14 November 2007. Also available as Space data and information transfer systems – Open archival information system – Reference model, ISO 14721:2003
- UN/CEFACT Modeling Methodology (UMM) v1.0
- The WARC File Format (Version 0.10) [http://archive-access.sourceforge.net/warc/warc\\_file\\_format-0.10.html](http://archive-access.sourceforge.net/warc/warc_file_format-0.10.html) visited 26 May 2008

### 3 Objective

The purpose of this specification is to simplify the transfer of custody of digital records from one records system to another. This is achieved by specifying a standard representation for a record during the actual relocation of the records from one system to the other, and a simple process for performing this relocation. The specification could be used for transfer:

- From the organization that created the records to an archive
- From a creating organization to its successor
- From an archive back to the creating organization
- From a creating organization to secondary storage and back again
- From one archive to another
- From one system to another within one organization

This specification has been particularly designed for the transfer of custody of records between a creating organization and an archive, but could be used for the other situations listed above.

#### 3.1 Background

Records are evidence of an organization or individual carrying out their day to day activities. Understanding what happened in the past and why is critical to the continued day to day activities of an organization or individual. It is also the basis on which legal systems are built, and it is the basis of all historical understanding. For all these reasons, preserving the ability to access records is critical to organizations and individuals.

Digital records are simply records represented digitally. Digital records range from written documents (such as reports and emails) through to images (photographs, plans), sound (voice mail), video, and databases. The continued integration of the computer into infrastructure means that new types of digital records are being continually developed.

In an organization these digital records are stored and managed by many records systems. These records systems may range from the very simple (e.g. the corporate file system) to the sophisticated (e.g. a dedicated electronic document management system). All of these records systems have one characteristic in common: they have a finite, relatively short life. Often the records system has a much shorter lifespan than the records that it holds.

The short lifespan of records systems leads to a key challenge in preserving digital records: ensuring that the digital records can be extracted from the records systems that currently store and manage them and be reliably transferred to another system. Four types of transfer are of particular importance:

- Transfer of records from one system to its replacement within an organization. This type of transfer occurs with a relatively low frequency, but involves the transfer of all records.
- Transfer of long term temporary records from the organization that created them to a specialist storage organization and back again. This is equivalent to the provision of secondary storage of paper records. This relieves an organization from having to preserve and store records that have no day to day use in operational systems.
- Transfer of permanent records from the organization that created them to an archive. This ensures that the long term preservation of permanent records is carried out by a specialist organization and relieves the creating organization from the burden of preserving, storing, and providing access to these records.
- Transfer of records between business partners or between units within an organization when those units do not use the same records system.

This standard is not intended for use in situations where records are relocated, but custody is not transferred such as relocation for backup/recovery and continuity of business operations.

The purpose of this standard is to reduce the risk of loss or compromised records, and the overall cost, of transferring digital records from one system to another. It does this by defining a standard transfer process and transfer format for records.

### **3.2 Benefits**

This standard transfer process and standard transfer format (also known as a Submission Information Package or SIP) can be used as a common exchange language between records systems. Without such a standard, any transfer of digital records between records systems requires specific custom arrangements between each pair of systems. Such custom arrangements have a number of issues:

- High risk of loss or compromised records
- High cost

The high risk and cost can often be managed in the limited case where a records system is replaced within an organization by another system. In general, however,

managing the risk and cost is far more difficult where one organization (such as an archive or secondary storage supplier) is receiving records from many organizations.

The use of this specification means that less custom software is required to transfer records, and this software can be re-used. Re-use may occur by the same organization (e.g. a secondary storage supplier using the same software to accept transfers from many different organizations), or by different organizations (e.g. the vendor of a records system implementing the transfer software once and providing it to all users of the system).

The use of this specification also means less intervention is required during the transfer of records. Intervention may occur at export, relocation, and import of records.

In other words, this specification is part of risk minimization and therefore ultimately results in cost reduction.

## 4 Scope

<i>Categories</i>	<i>Description and Values</i>
<i>Business Process</i>	<i>Transfer of digital records</i>
<i>Product Classification</i>	<i>Records management</i>
<i>Industry Classification</i>	<i>Government – Archiving sectors</i>
<i>Geopolitical</i>	<i>Global</i>
<i>Official Constraint</i>	<i>International and national regulations</i>
<i>Business Process Role</i>	<i>IT provider and integrator</i>
<i>Supporting Role</i>	<i>Producer/Archive</i>
<i>System Capabilities</i>	<i>No limitation</i>

The scope of this specification is to define a transfer process and a Submission Information Package that may be used by all records systems in all organizations when transferring records between records systems. The systems may be located in the same organization or in different organizations. The specification is primarily intended for use when transferring records from a producer to an archive, but may be used for other types of transfer.

To prevent confusion, the following terms will be used in this specification:

- A *transfer* as a whole is the business activity involved in transferring custody from one system to another. This may involve the transfer of physical custody or legal custody. Transfer includes tasks ranging from the extremely high level (e.g. agreement at the ministerial level as to access provisions), to the extremely low level (e.g. physical relocation of the records). This specification only covers some of the tasks involved in carrying out a transfer of digital records.

- A *transfer agreement* is negotiated between the staff of the respective organisations as part of the transfer, and would typically cover: whether the transfer would occur at all; what records are to be transferred (including their types); when they are to be transferred (and how often); access arrangements; and minimum metadata standards. Once negotiated, the agreement may remain in existence for a long time and cover many instances of actually transferring records. For example, a transfer agreement may specify that records are to be transferred annually. The negotiation and subsequent maintenance of the transfer agreement, as a whole, is not part of this specification.
- A *transfer session* is the activities involved in administering and performing an actual transfer of records within a transfer agreement. These activities include, for example, identifying the records that are to be transferred within this transfer session, ensuring that appropriate documentation has been completed, setting up quality assurance mechanisms to ensure that the correct records have been transferred, ensuring agreement between the records systems at either the producer or archive as to the records that will be transferred, ensuring that all agreed records have been accurately moved, and that the recipient has accepted responsibility for the preservation of the records. There may be many transfer sessions conducted under the authority of a single transfer agreement.

The following issues may need to be considered when conducting a transfer session:

- Transferring the records may involve various processes that take different amounts of time, some of which may be quite lengthy. For example, several levels of approval may be required before a transfer of custody is completed, or manual inspections may be needed to ensure the right records were transferred or that restricted content is properly identified.
- Records may be lost, corrupted, or detached from their metadata during the transfer session. When considering loss, it is important to note that records may be lost at other points than just during transfer over a network (or other physical transfer mechanism). Records can be lost, for example, due to software failure in the producer or archive, or a decision by the archive to delete them (e.g. due to a processing failure).
- During the processing of a transfer session, it may be decided that some records that are intended to be transferred will not actually be transferred. This might occur because the process of transferring detects errors in the records and it is decided that it is too difficult or expensive to correct these errors for this transfer session.
- The data model of a 'record' may vary significantly between jurisdictions. A typical model is hierarchical and might consist, for example, of a functional classification, which contains files, which contain items, which contain (recursively) sub items. Other models are possible and may be considerably different.
- The organization of the information within a record in a transfer session may differ between jurisdictions and between transfers. Options for organizing the information include: separating a record into its physical parts and sending them separately; packaging all of the parts of a record into a single object; and packaging multiple records into a single object. There are many standards for packaging the information in a record into physical objects including METS and MPEG-21 DIDL. Jurisdictions may also have their own packaging standards.

- The metadata associated with a record may vary significantly between jurisdictions and between applications. There are many sets of metadata used for a variety of purposes and defined by a variety of bodies, and it is expected that more will be defined in the future. However, arbitrary use of metadata sets will increase the cost of transfer to both the producer and the archive. The lack of a standardized set of metadata will require customization of software for the producer and, possibly, the archive, and will make it difficult for an archive to build a consistent set of metadata for its collection.
- Record content can be represented by a variety of data formats. These range from native formats (e.g. Open Document Format, PDF, JPEG), to container formats (e.g. METS). Record content can be explicitly included in the transfer message, or be referenced from the transfer message using URLs. In the later case it is expected that the archive will use the URLs to retrieve a copy of the content before formally accepting custody.
- While the basic purpose of this specification is to transfer custody to a new system, there are a number of variations on this theme. This includes whether the source system destroys their copy after custody has been transferred, transfer of responsibility for the management of the records to an appropriate authority while physical storage of the record is retained by the producer, or transfer of physical records (where only the metadata associated with the records is transferred electronically).

These issues are supported by this specification.

This specification defines a reliable submission process for actually transferring custody of records from one system (producer) to another (archive). It provides the following features:

- Variable duration. It does not assume that responses will occur within a short period of time, and recognizes that the responses could be delayed for lengthy periods.
- Error recovery to recover from loss of records during transfer. The transfer process is reliable in that it will recover from any loss of records during the process at the archive (during processing), the producer, or in the network.
- Mechanisms for reducing the scope of the transfer. It may be necessary to cut the transfer short even though not all records have been transferred. This might occur because it is decided that some records are not to be transferred during this session. The transfer status of every record is fully documented.
- A flexible representation of a record. A record is an arrangement of objects, but the meaning of the objects is for an individual jurisdiction. This accommodates the different data models of 'record' that exist in different jurisdictions.
- Flexibility to associate any metadata with a component during the transfer session. The specification defines a minimal set of metadata that contains the information it is expected that the transfer process will need to access. However, the specification permits producers to add other sets of metadata to be used to describe or manage digital records during or after the transfer.
- Flexibility to transfer any data format. A simple representation of a data object is defined, but jurisdictions can use other representations defined in other standards.
- Variations in the transfer scenario are possible. It is possible to move just metadata, and hence to move the either digital or paper records between a producer and an archive, or not to move the records at all. It is not required that the producer destroy their copy after transfer of custody.

This standard does not define:

- The negotiation or management of a transfer agreement between the producer and the archive at an organizational level<sup>9</sup>.
- The legal aspects of transfer, which depend on the agreement between parties and/or national legislations.
- The administration of a transfer session.
- Processes between more than two parties. For example: an agency asking a secondary storage which temporarily holds its records to transfer these records to another archive, or where records are simultaneously transferred to multiple archives.
- The definition of a record or the organization of records. The specification represents the information being transferred as any arrangement of records, collections of records, and parts of records.
- The behaviour of the records systems at either the producer or archive beyond that necessary to process the messages defined in this specification. The specification assumes that the records system is designed to hold records in a manner that ensures their authenticity and integrity. The standard does not define the long term preservation package format (i.e. the archival information package format).
- The data formats (e.g. PDF, TIFF) in which an archive will accept the content of digital records. The SIP is designed to contain any content in any format. These formats may be other transfer formats (e.g. METS) in addition to native formats such as Open Document Format or PDF. An archive may, outside this standard, restrict the data formats that it accepts.
- The metadata sets that are to be transferred with the record, except a minimal set of metadata used for transfer. The SIP in particular is designed to contain any metadata sets. An archive may, outside this standard, restrict the metadata sets that it accepts. An archive that accepts any metadata set may find that the records transferred lack sufficient context (metadata) to retain authenticity or integrity. It also may have difficulty in providing coherent access to the records.
- The physical mechanisms by which the records are transferred from one records system to another or from a producer to an archive. This physical representation could be moved between the records systems in many different ways (e.g. by the internet or on media such as CDs).
- The low level protocols that ensure accurate data transmission between the producer and the archive.
- Physical mapping of the business requirements to any technical syntax (for example XML). This mapping will be found in the associated Requirement Specification Mapping (RSM).

The audience for this standard is:

- Those responsible for records within organizations (e.g. records managers, archivists, IT specialists, and corresponding senior managers). This group needs to understand the purpose of the standard and, broadly, how it is achieved. This group also needs to know what the standard does not cover so that these areas can be negotiated prior to a transfer.

---

<sup>9</sup> The high level transfer process is addressed by ISO 20652:2006 (Producer-archive interface - Methodology abstract standard) which defines the methodology for the structure of actions that are required from the initial time of contact between the producer and the archive until the objects of information are received and validated by the archive.

- Those responsible for carrying out a transfer of records between systems (e.g. recordkeepers, archivists, and information technology staff). This standard enables them to know how a transfer is represented so that they can carry out a specific transfer.
- Those responsible for designing and implementing software for exporting and importing records from/to records systems (e.g. vendors, software developers). This standard enables them to have a precise, clear, and unambiguous specification of the transfer process and/or submission information package so that they can accurately implement it.
- Those responsible for testing systems to ensure that systems accurately implement the transfer process and/or submission information package. This standard provides a precise, clear, and unambiguous specification to ensure defensible decisions about whether an implementation fulfils the specification or not.

## **5 Business Requirements**

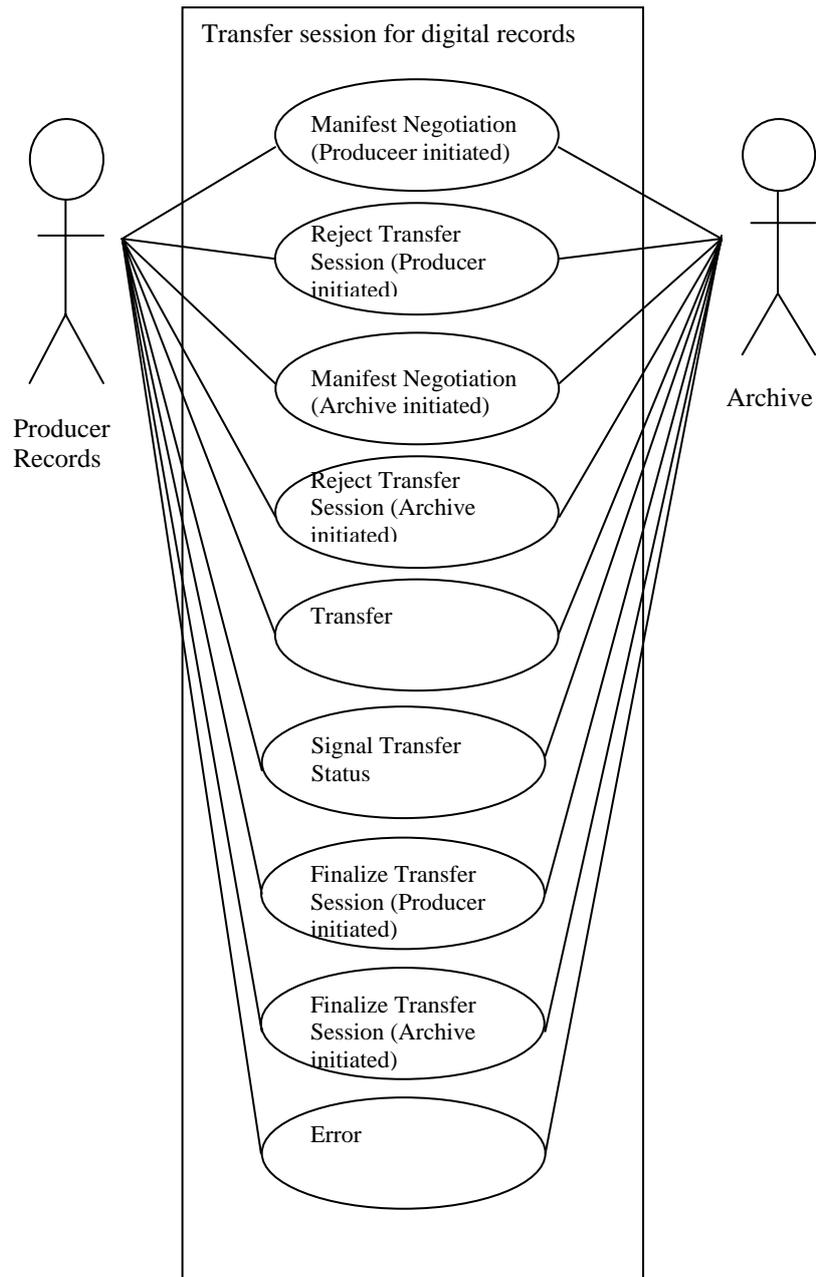
### ***5.1 Business process elaboration***

#### ***5.1.1 Transfer session for digital records***

*Name of the use scenario:* Transfer session for digital records

*Actors of the use case:* Producer / Archive

The producer is the individual or organization that is the current custodian of the records. The archive is the individual or organization that will receive the records. The producer is typically an agency and the archive is typically an archival institution or a secondary storage supplier. But the producer may be an archival institution or a secondary storage supplier and the archive may be an agency (for example when records are transferred back from the archival institution or the secondary storage supplier to the agency). Therefore, many cases are covered by the specification: agency to archive, archive to archive, archive back to agency, archive to secondary storage, and so on.



*Description of the use case:*

This use scenario transfers custody of records from the producer to the archive in a transfer session.

This use case is small portion of a larger business process managing the transfer of custody. This broader business process is largely manual, and includes: negotiation about what records (at an organizational level) are to be transferred and when they are to be transferred, agreements as to access and retention periods, etc.

Before the records are transferred, the producer and archive agree upon a list of records (a manifest) that are to be transferred in this transfer session. Transfer

continues until the producer or archive signals that the transfer session is to be finalized.

There are nine use cases covered by this scenario:

- *Manifest Negotiation (Producer initiated)*. This use case involves the producer proposing a set of records to be transferred, and the archive accepting some or all of those records.
- *Manifest Negotiation (Archive initiated)*. This use case involves the archive informing the producer of the records that the archive is willing to take custody of, and the producer agreeing to transfer some or all of those records.
- *Reject Transfer Session (Producer initiated)*. This use case occurs when the archive rejects the proposed transfer session. This starts in the same fashion as the Manifest Negotiation (Producer initiated) with the producer proposing a set of records to be relocated in this transfer session. However, the archive then rejects the proposed transfer session in its entirety. This would occur, for example, if the archive has no knowledge of the proposed transfer session (e.g. it was not taking place within a negotiated transfer), the reference to the negotiated transfer in the proposal was invalid, or the archive was in a state where it could not commence the transfer session. This use case terminates the transfer session.
- *Reject Transfer Session (Archive initiated)*. This use case occurs when the producer rejects the proposed transfer session. This starts in the same fashion as the Manifest Negotiation (Archive initiated), with the archive proposing a set of records to be relocated in this transfer session. However, the producer rejects the proposed transfer session in its entirety. This would occur if the producer has no knowledge of the proposed transfer session (e.g. it was not taking place within a negotiated transfer), the reference to the negotiated transfer in the proposal was invalid, or the producer was in a state where it could not commence the transfer session. This use case terminates the transfer session.
- *Transfer*. This use case involves the producer moving a record to the archive (in a submission information package). This use case must be preceded by a Manifest Negotiation use case. The record content may either be included in the submission information package or referenced.
- *Signal Transfer Status*. This use case involves the archive informing the producer of the transfer status of all of the records in the manifest agreement. One particular transfer status is 'custody accepted' which indicates that the archive accepts responsibility for that record. There is no direct relationship between Transfer and Signal Transfer Status use cases. An archive could, for example, perform a Signal Transfer Status use case for each Transfer use case, or it could perform a Signal Transfer Status use case periodically.
- *Finalize Transfer Session (Producer initiated)*. This involves the producer informing the archive that the transfer session has been completed. This may be sent at any time after the Manifest Negotiation use case, even before the archive has accepted custody of all of the records. Finalizing the transfer session before accepting custody of all the records might occur, for example, if the archive signals that some records cannot be accepted due to errors, and the producer determines that it is not worth while correcting the errors and consequently decides that it is not worth relocating those records. The archive may complete processing of any records it has received before responding to the finalize message, or it may abandon them.

- *Finalize Transfer Session (Archive initiated)*. This involves the archive informing the producer that the transfer session has been completed. This may be sent at any time after the Manifest Negotiation use case.
- *Error*. This involves one party informing the other of message that was received that either was out of sequence, or contained erroneous information..

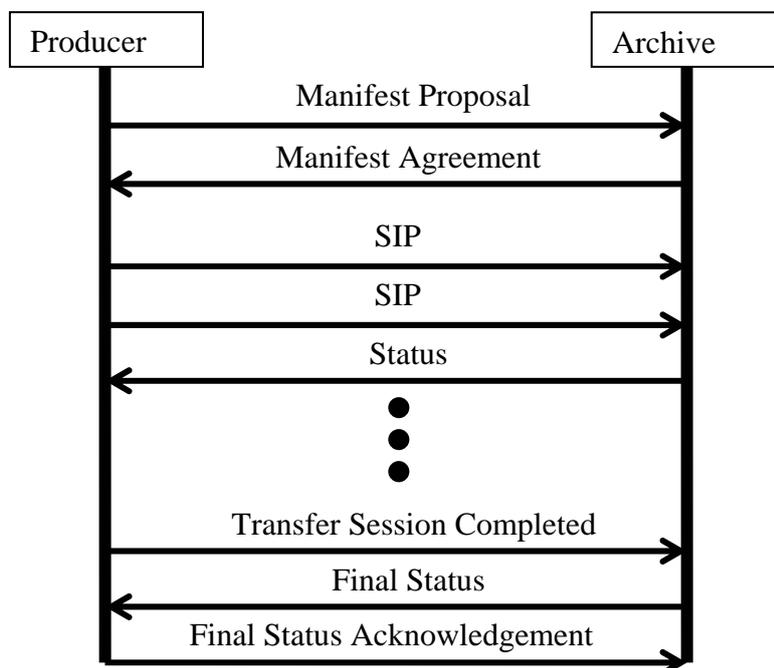
Note that the specification does not support aborting the transfer session. The producer or archive may terminate a transfer session early using the respective Finalize Transfer Session use case, but any records that have already been transferred at this time remain with the archive. This is because once custody of a record has been accepted by the archive it requires a second, formal, transfer session to pass custody off to another system (even the system that originally held the records).

The specification does not specify the following functions when transferring records:

- Temporarily suspending the transfer session.
- Scheduling activities
- Bi-directional exchange of metadata
- Retaining links with the original records system
- Provision of a destruction notice from the producer to the archive.

### 5.1.2 Example of a normal transfer session

The following sequence diagram is an example of the interactions between producer and the archive during a normal transfer session.



A normal Transfer Session commences with a Manifest Proposal/Manifest Agreement exchange, then the producer transmits one or more SIP messages, and the archive responds with one or more Status messages. The transfer session concludes with a Transfer Session Completed/Final Status/Final Status Acknowledgement exchange.

The Manifest Proposal message lists all of the records that the producer intends to transfer to the archive. The Manifest Agreement indicates which of the proposed records the archive is willing to accept. The Manifest Agreement positively indicates the acceptance or rejection of every record proposed by the producer to ensure a definite understanding by both the producer and the archive of the records to be transferred. It is envisaged that the process of accepting or rejecting the records may be partially manual. When the archive receives a Manifest Proposal, an archivist may inspect the Manifest Proposal to determine if the archive will accept the records (e.g. checking the titles or classification of the records). In order to assist this process, one or more collections of metadata (known as metadata sets) can be associated with each record.

Once the producer has received a Manifest Agreement message it may begin transferring the records. Records are transferred in SIP messages. The producer may assign records to SIP messages in any convenient fashion. For example, the producer may place a single part of a record in each SIP message, all the parts of one record in a single SIP message, or multiple records in a single SIP message. Assigning records to SIP messages may form part of the higher level transfer negotiation. For example, staff representing the archive and the producer may negotiate a maximum SIP size to optimize processing of the messages in the archive. The assignment of records to SIPs is listed in the Manifest Proposal so that the archive is aware when all the parts of a record have been received.

The archive returns Status messages to the producer to indicate the current status of all the records in the agreement. The status of an individual record includes:

- Rejected for transfer
- Agreed to be transferred. The record has been accepted by the archive in the Manifest Agreement message but not yet received.
- Received by archive. The record has been received by the archive in a SIP message and is being processed.
- Rejected, resubmit. The archive rejected this record during processing for an internal reason (e.g. lack of resources to continue processing). The producer should resubmit the record.
- Rejected, correct and resubmit. The archive rejected this record because of a correctable error in the record. Examples would include missing mandatory metadata, contradictory metadata, virus infection, DRM protection, or a format that could not be accepted by the archive. The producer is expected to correct the error and resubmit the record.
- Rejected, do not resubmit. The archive rejected this record because of a fatal error in the record that the producer is not expected to be able to correct. Examples of a non-correctable error are a determination that the record is not to be transferred, and the record is too massive for processing. The distinction between correctable and non-correctable errors is not hard and fast, and the producer and archive may disagree over whether the error is correctable. The producer may attempt to correct and resubmit the record.
- Custody accepted. The archive has accepted responsibility for this record.

It is important to note that there is no direct relationship between the SIP and Status messages. An archive is not required to return a Status message for each SIP message that it receives and can generate Status messages at its convenience or as agreed in the broader transfer agreement. For example, the archive might return a Status message

daily, or upon actually accepting custody. The reason for this lack of a direct relationship between SIP and Status messages is that there may be a very significant period of time between an archive receiving a record in a SIP message and accepting custody (this period could amount to weeks or months). There may, for example, be manual processing or approval steps in the accession workflow. Alternatively, there may be lengthy processing steps such as file format conversion, quarantine periods for virus checking, or ensuring a copy of the record is held off site. Breaking the link between the receipt of a SIP message and the transmission of a Status message means that the archive can effectively send multiple Status messages for a record (recording receipt and eventual disposition). It also allows the archive to effectively collect the status of individual records and batch them in a single message sent at a convenient time.

The transfer session is finished when the producer transmits a Transfer Session Completed message. The archive acknowledges the receipt of this message by transmitting a Final Status message. As the name implies, this gives the final status of all records in the original Proposal. The producer acknowledges receipt of the Final Status by a Final Status Acknowledgement message.

### **5.1.3 Records and SIPs**

To understand this specification it is necessary to understand the distinction between records and SIPs. Essentially, transfer sessions are negotiated and managed using records, but the transfer is performed using SIPs.

The Manifest Proposal/Manifest Agreement exchange is a negotiation to ensure that both the producer and the archive agree on the records to be transferred. This agreement also lists the SIPs that will contain the records when the records are actually transmitted from the producer to the archive. Listing the SIPs in the agreement allows the archive to determine when it has received a complete record.

When the producer and archive are deciding how to ship a record to an archive they have complete flexibility in assigning records to SIPs. They may, for example, decide to break the record into separate parts (e.g. individual files) and put one part in each SIP. A second option would be to put a complete record into one SIP. A third option would be to package multiple records in one SIP.

The producer and archive similarly have flexibility in deciding how records are encoded within SIPs. It is possible to simply include the binary content of records directly in the SIP. Alternatively, records may be packaged in a range of encapsulation standards such as METS or MPEG-21 DIDL. The method of encoding records within SIPs must be negotiated outside this standard in the broader transfer agreement.

The Status and Final Status messages indicate the status of the records being transferred. These messages also indicate whether the SIPs that contain the records have been received. This allows the archive to signal which SIPs it has received, and also signal any issues it has encountered in processing the SIPs.

## 5.2 Information flow definition

### 5.2.1 Sequence diagrams

A transfer session commences with the exchange of Manifest Proposal and Manifest Agreement messages<sup>10</sup> (forming the Manifest Negotiation), then the producer transmits one or more SIP messages (forming the Transfer), and the archive responds with one or more Status messages (forming the Signal Transfer Status). The transfer session concludes with a Completed/Final Status/Final Status Acknowledgement exchange (forming the Finalize Transfer Session).

It is important to note that restrictions on the possible sequencing of messages are found in the section on Business Rules (section 6). In this section, comments on possible sequences are only discussed where they are important to understand the functioning of the protocol.

#### 5.2.1.1 Manifest Negotiation (Producer initiated) sequence diagram



The Manifest Proposal message lists all of the records that the producer intends to transfer to the archive in this transfer session. For each record it also lists the SIPs that will be used to contain the record. The archive will respond with either the Manifest Agreement or Reject Transfer Session messages<sup>11</sup> (the Reject Transfer Session response will be described in the next section).

The Manifest Agreement message indicates which of the proposed records the archive is willing to accept. It is important to note that this is intended to allow the archive to reject records that have been included in error by the producer but which are not part of the transfer agreement (e.g. are not part of the series being transferred). It is not intended to allow the archive to selectively reject parts of a series or fond.

The Agreement positively indicates the acceptance or rejection of every record proposed by the producer. This ensures a definite understanding by both the producer and the archive of each record proposed to be transferred in this transfer session.

<sup>10</sup> It was decided not to make the Manifest Proposal/Agreement exchange optional as was felt that the benefits would be outweighed by the additional complexity in the protocol. Making the exchange optional would mean that the Status messages sent by the archive could only contain information about those objects actually received by the archive.

<sup>11</sup> Note that there is minimal opportunity for negotiation. The producer proposes, and the archive agrees (possibly to a subset of the records proposed). The archive must make the decision on the information contained in the proposal, which may include metadata describing the objects and relations linking the records. The archive cannot request more information. This approach drastically simplifies the complexity of the specification and implementations. It is assumed that the amount of detail to be supplied to the archive to make its decision is agreed upon by the staff of the two organizations as part of the broader transfer agreement.

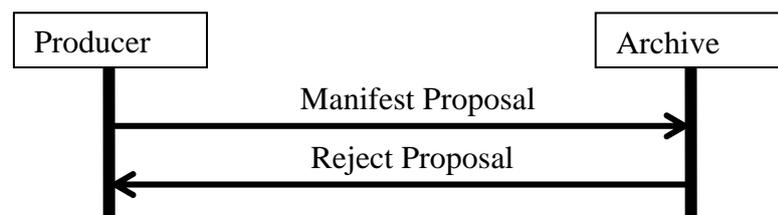
Accepting a record means that all the SIPs holding the record must also be accepted. An archive cannot accept a record, but reject a SIP holding the record.

It is envisaged that the process of accepting or rejecting the records could be partially manual in some jurisdictions. When the archive receives a Manifest Proposal, an archivist could inspect the Manifest Proposal to determine if the archive will accept the records (e.g. checking the titles or classification of the associated records). In order to assist this process, one or more metadata sets can be associated with each proposed record.

If the archive does not respond within a reasonable period of time to the Manifest Proposal, the producer must assume that the Manifest Proposal (or the response) has been lost and resubmit the Manifest Proposal. The 'reasonable period of time' could be defined for each transfer session in the broader transfer agreement. Alternatively, the retransmission could be manually triggered by the personal managing the transfer session.

### 5.2.1.2 Reject Transfer Session (Producer initiated) sequence diagram

The following sequence diagram shows the interactions between producer and the archive when a Manifest Proposal is rejected.



In this sequence the archive has responded to a Manifest Proposal message with a Reject Proposal message. This indicates that the transfer session has been rejected entirely by the archive. The intention is that the transfer session is rejected when the archive cannot process the transfer session at all, typically because the archive has no knowledge of the proposed transfer session (e.g. it was not taking place within a negotiated transfer), or the archive was in a state where it could not commence the transfer session. If the Manifest Proposal could be processed, but all the records are rejected for transfer, a Manifest Agreement message rejecting all the records must be returned. The producer must then send a Transfer Session Completed message, to which the archive will respond with a Final Status message.<sup>12</sup>

This sequence terminates the transfer session.

<sup>12</sup> An alternative would be to treat this as we have treated other failed messages. We chose to have a specific message because failed proposals are likely to be slightly more common (e.g. due to inaccurate set-up of the transfer session at either the Producer or the Archive).

### 5.2.1.3 Manifest Negotiation (Archive initiated) sequence diagram



This sequence diagram differs from 5.2.1.1 in representing a ‘pull’ model where the archive informs the producer of the records that the archive is willing to take custody of. An example of this approach is where an organization retrieves records from a secondary storage supplier.

The Manifest Proposal message lists all of the records that the archive intends to be transferred in this transfer session. The producer will respond with either the Manifest Agreement or Reject Transfer Session messages (the Reject Transfer Session response will be described in the next section).

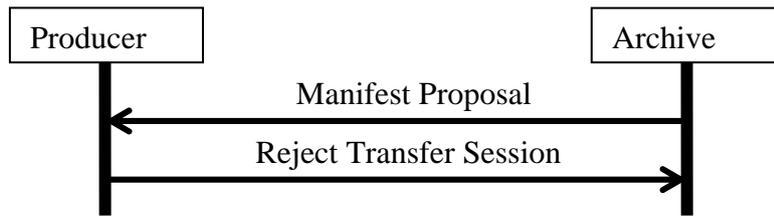
The Manifest Agreement message indicates which of the proposed records the producer is willing to transfer. The Agreement positively indicates the acceptance or rejection of every record proposed by the archive. In this use case, the Manifest Proposal may not contain the SIPs that contain the record (as only the producer may know this information). The Manifest Agreement must, however, contain the breakdown of the record into its SIPs.

The archive must ignore any message it receives until it receives a Manifest Agreement (or Reject Transfer Session). Note that the producer may commence to transmit SIP or Transfer Session Completed messages as soon as it has sent the Manifest Agreement message. If the Manifest Agreement message is lost (or the messages are reordered), the archive may receive SIP or Transfer Session Completed messages before receiving a Manifest Agreement. In this case the archive must ignore the SIP or Transfer Session Completed messages until it has received a Manifest Agreement (probably resulting from the retransmission of the Manifest Proposal). This is different to the Manifest Negotiation (Producer initiated) where the producer will not transmit a SIP or Transfer Session Completed message until it receives the Manifest Agreement.

If the producer does not respond within a reasonable period of time to the Manifest Proposal, the archive must assume that the Manifest Proposal (or the response) has been lost and resubmit the Manifest Proposal. The ‘reasonable period of time’ could be defined for each transfer session in the broader transfer agreement. Alternatively, the retransmission could be manually triggered by the personnel managing the relocation.

### 5.2.1.4 Reject Transfer Session (Archive initiated) sequence diagram

The following sequence diagram shows the interactions between producer and the archive when a Manifest Proposal is rejected.

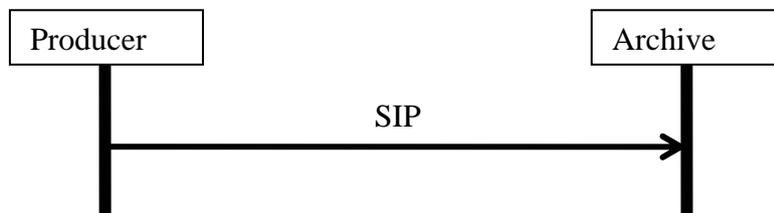


In this sequence the producer has responded to a Manifest Proposal message with a Reject Transfer Session message. This indicates that the transfer session has been rejected entirely by the producer<sup>13</sup>. The intention is that the transfer session is rejected when the producer cannot process the session at all, typically because the producer has no knowledge of the proposed transfer session (e.g. it was not taking place within a negotiated transfer), of an error in the Manifest Proposal (e.g. the producer has no knowledge of the Transfer Id) or the producer was in a state where it could not commence the transfer session. If the Manifest Proposal could be processed, but all the records are rejected for transfer, a Manifest Agreement message rejecting all the records should be returned. The producer should then send a Transfer Session Completed message, to which the archive will respond with a Final Status message.

This sequence terminates the transfer session.

#### 5.2.1.5 Transfer sequence diagram

Once the producer has received or sent a Manifest Agreement message it may begin transmitting the SIPs that contain the records. It is important to note that SIP messages may transport part of a record, a complete record, or a collection of records.



Depending on the needs of the producer and the archive, the producer may place a single part of a record in each SIP message. Alternatively, it may place all parts of all the records in a single SIP message encoded in a suitable format (e.g. METS, MPEG-21 DIDL, or WARC containers). More realistically, however, closely related parts are likely to be placed in a single SIP message in a single encoding. The method of assigning records to SIP messages forms part of the organizational level transfer negotiation. For example, staff representing the archive and the producer may negotiate a maximum SIP message size to optimize processing of the messages in the archive.

<sup>13</sup> An alternative would be to treat this as we have treated other failed messages (ignore and rely on the administrator to signal out-of-band). We chose to have a specific message because failed proposals are likely to be slightly more common (e.g. due to inaccurate set-up of the transfer session at either the Producer or the Archive).

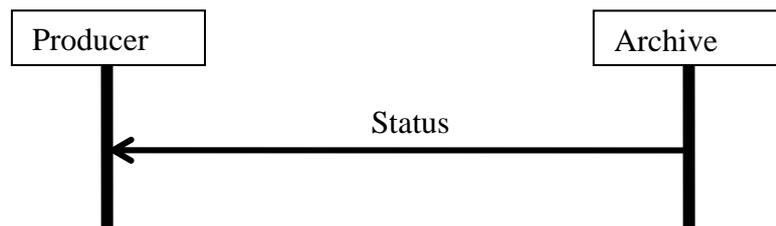
A SIP can contain alternative representations of a record or part of a record. For example, a SIP could contain both a PDF and an Open Document Format representation.

The content of a representation may be digital or physical. Physical content is a reference to a physical object. This reference is not further defined in this specification, but it is intended to uniquely identify a physical artefact in the producer's records system. Digital content may either be directly included in the SIP, or referenced by a URL.

The producer must be prepared to retransmit SIPs if necessary. This will occur if requested by the archive (indicated by the 'rejected, resubmit', or 'rejected, correct and resubmit' record status). The producer may also need to retransmit the SIP if it is not acknowledged by the archive within a reasonable period of time (possibly indicating that the SIP has been lost<sup>14</sup>). A 'reasonable period of time' is not defined in this standard as it will depend on the individual transfer and participants. The period of time may be defined in terms of weeks. The 'reasonable period of time' could be defined for each transfer session in the broader transfer agreement. Alternatively, the retransmission could be manually triggered by the personnel managing the relocation.

#### 5.2.1.6 Signal Transfer Status sequence diagram

The archive returns Status messages to the producer to indicate the current status of all the records and SIPs covered by the Manifest agreement<sup>15</sup> or received.



The status of an individual record can be:

- Rejected for transfer (i.e. proposed in the Manifest Proposal message, but rejected in the Manifest Agreement message)
- Agreed to be transferred. (i.e. agreed to be transferred in the Manifest Agreement message but not yet received by the archive).
- Received by archive. (i.e. all the SIPs containing this record have been received by the archive and are being processed)

<sup>14</sup> It is not assumed that the underlying network is reliable (for example, messages may be exchanged by snail mail on CDs). Further, even if the underlying transmission of messages is reliable, the software running on the archive and the producer may not be perfectly reliable. For example, the producer's system may identify a record to be relocated, but fail to extract all the objects that form it from the local system and include them in a SIP message. Equally, the archive's system may lose an object during processing. It is consequently necessary to have some form of recovery mechanism.

<sup>15</sup> With the exception of the Reject Transfer Session message, all of the messages transmitted by the archive to the producer are variations on the Status message, and all explicitly list the status of each record and SIP originally proposed in the Manifest Proposal message. Although this requires the transmission of more information, it simplifies processing by both the archive and producer and prevents confusion and loss of synchronization.

- Rejected, resubmit. (i.e. the archive rejected this record during processing for an internal reason such as a lack of resources to continue processing. The producer should resubmit the SIPs forming the record.)
- Rejected, correct and resubmit. (i.e. the archive rejected this record because of a correctable error in the record. Examples of a correctable error could be missing mandatory metadata, contradictory metadata, virus infection, DRM protection, or a format that could not be accepted by the archive. The producer is expected to correct the error and resubmit the affected SIPs containing the record.)
- Rejected, do not resubmit. (i.e. the archive rejected this record because of a fatal error in the record that the producer is not expected to be able to correct. Examples of non-correctable errors could be a determination that the record is not listed in the Manifest Agreement and hence is not to be transferred, or that the record is too massive for processing. The distinction between correctable and non-correctable errors is not hard and fast, and the producer and archive may disagree over whether the error is correctable. The producer may attempt to correct and resubmit a record even if the archive rejected it with 'do not resubmit' status.)
- Custody accepted. (i.e. the archive has accepted responsibility for this record.)

The status of an individual SIPs can be:

- Not yet received.
- Received by archive. (i.e. the SIP has been received by the archive and is being processed)
- Rejected, resubmit. (i.e. the archive rejected this SIP during processing for an internal reason such as a lack of resources to continue processing. The producer should resubmit the SIP.)
- Rejected, correct and resubmit. (i.e. the archive rejected this SIP because of a correctable error. Examples of a correctable error could be missing mandatory metadata, contradictory metadata, virus infection, DRM protection, or a format that could not be accepted by the archive. The producer is expected to correct the error and resubmit the affected SIP.)
- Rejected, not included in Transfer Agreement. (i.e. the archive rejected this SIP because the SIP is not listed in the Manifest Agreement and hence is not to be transferred.)
- Finalized. (i.e. the SIP is part of a record for which the archive has accepted custody.)

Note that it is not permitted to reject permanently a SIP as permanently rejecting a SIP would imply that the record could not be completely transferred and hence custody could not be accepted.

Other status values can be defined by an archive, but it is expected that the values in the above list would be used if applicable.

It is important to note that there is no direct relationship between receiving a SIP message and sending a Status message<sup>16</sup>. An archive is not required to return a Status

---

<sup>16</sup> The reason for this is that there may be a very significant period of time between an archive receiving an object in a SIP message and accepting custody (this period could amount to weeks or months). There may, for example, be manual processing or approval steps in the accession workflow at the archive. Alternatively, there may be lengthy processing steps such as file format conversion, quarantine periods for virus checking, or ensuring a copy of the object is held off site. Breaking the link between the receipt of a SIP message and the transmission of a Status message means that the archive can effectively send multiple Status messages for an object (recording receipt and eventual disposition). It

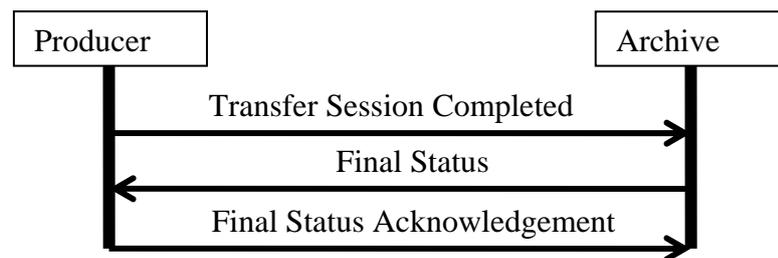
message for each SIP message that it receives. Instead, an archive is free to generate Status messages at its convenience or as agreed in the broader transfer agreement. For example, the archive might return a Status message daily, upon actually accepting custody, or it may never return a Status message and rely on the Final Status message.

The producer must not process any re-ordered Status messages. This must be detected to avoid confusion (e.g. records for which custody had been accepted suddenly changing back to received). This is detected by comparing the MessageId elements.

### 5.2.1.7 Finalize Transfer Session (Producer initiated) sequence diagram

A producer terminates a transfer session by transmitting a 'Transfer Session Completed' message. The archive acknowledges the receipt of this message by transmitting a Final Status message which gives the final status of all records in the original Manifest Proposal. The producer acknowledges receipt of the Final Status message by sending a Final Status Acknowledgement which the archive can store as evidence that the producer was informed about the custody status of the records transferred.

Any records with a status of 'Custody Accepted' in a Status message (including the Final Status message) have been accepted by the archive. For all other records, custody remains with the producer (even if the records have been transmitted to the archive in SIP messages).



Apart from the normal termination of a transfer session when all records have been accepted by the archive, the Finalize Transfer Session sequence allows the producer to abort the transfer session or to cut it short if the producer decides that some records will not be transferred after the transfer has commenced. This might occur if it proves impracticable to transfer some of the records, for example, if the archive detected significant problems with the records (such as they were not in a format approved for transfer) and it was decided that it was not possible or worthwhile to fix the problems. If it was not possible to cut the transfer session short it may prove to be impossible to finalize the session with the archive refusing to accept a particular record, and a producer that is unable or unwilling to fix the record to be acceptable. Deciding that a particular record will not be transferred within this transfer session does not preclude the producer from transferring it in a subsequent transfer session.

Note that it is not possible to simply abort a transfer session and roll back to the original state with the records at the producer. This is because once custody has been accepted by the archive, the record must be formally transferred back to the producer.

The producer may send the Transfer Session Completed message at any time after receiving or sending a Manifest Agreement message, irrespective of the transfer status

---

also allows the archive to effectively collect the status of individual object and batch them in a single message sent at a convenient time.

of the records being relocated. The Transfer Session Completed message may be sent before the producer has sent all of the SIPs in the agreement, and it may be sent before the archive has accepted custody of the records that it has received.

There is no requirement that the archive must respond immediately to a Transfer Session Completed message. The expectation is that ingesting records into the archive is likely to take a significant period of time. An archive may receive a 'Transfer Session Completed' message and continue to process the records it has already received (possibly up to the point that custody has been accepted). Alternatively, the archive may immediately abandon processing and discard any records for which it has not yet accepted custody. At some point the archive will cease processing the records and respond with a 'Final Status' message giving the final status of all records.

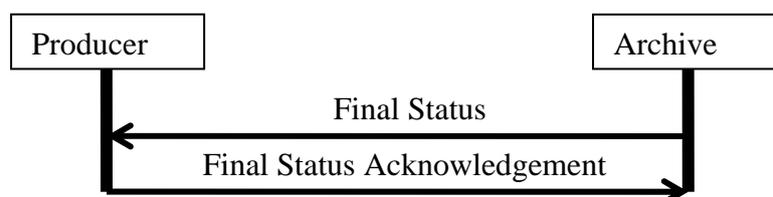
The archive must, however, respond to a 'Transfer Session Completed' message within a reasonable period. A 'reasonable period' is not defined in this specification. Given the lengthy processing time likely to be required for accepting custody of records, a reasonable period may be as long as a month. If the archive does not respond within a reasonable period to a 'Transfer Session Completed' message, the producer must resend the 'Transfer Session Completed' message. The archive may consequently receive duplicate 'Transfer Session Completed' messages. In this case the archive must always respond with a duplicate Final Status message (if one has already been sent).

If the producer does not respond within a reasonable period to a 'Final Status' message, the archive must resend the 'Final Status' message. The archive may consequently receive duplicate 'Final Status Acknowledgement' messages. A 'reasonable period' is not defined in this specification.

#### 5.2.1.8 Finalize Transfer Section (Archive initiated) sequence diagram

The archive may terminate a transfer session at any time by transmitting a Final Status message. As the name implies, this gives the final status of all records in the original Manifest Proposal. The producer acknowledges receipt of the Final Status message by sending a Final Status Acknowledgement which the archive can store as evidence that the producer was informed about the custody status of the records transferred.

Any records with a status of 'Custody Accepted' in a Status message (including the Final Status message) have been accepted by the archive. For all other records, custody remains with the producer (even if the SIPs that contain the record have been transmitted to the archive).



This sequence allows the archive to abort the transfer session or to cut it short if the archive decides that some records will not be transferred after the transfer has commenced.

Note that it is not possible to simply abort a transfer session and roll back to the original state with the records at the producer. This is because once custody has been accepted by the archive, the records must be formally transferred back to the producer.

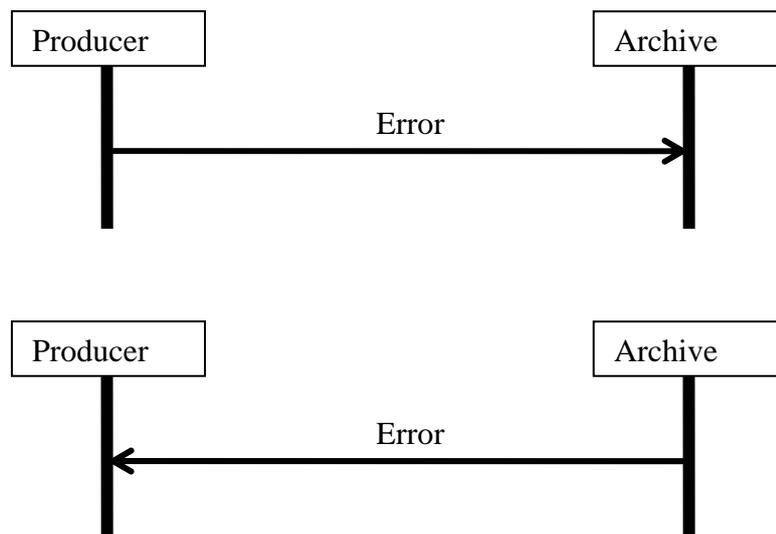
The archive may send the Final Status message at any time after receiving or transmitting the Manifest Agreement message, irrespective of the transfer status of the records being transferred. The Final Status message may be sent before the producer has sent all of the records in the agreement, and it may be sent before the archive has accepted custody of the records that it has received.

If the archive receives any messages after transmitting the Final Status message, the archive must send a duplicate 'Final Status' message.

If the producer does not respond within a reasonable period to a 'Final Status' message, the archive must resend the 'Final Status' message. A 'reasonable period' is not defined in this specification.

#### 5.2.1.9 Error sequence diagram

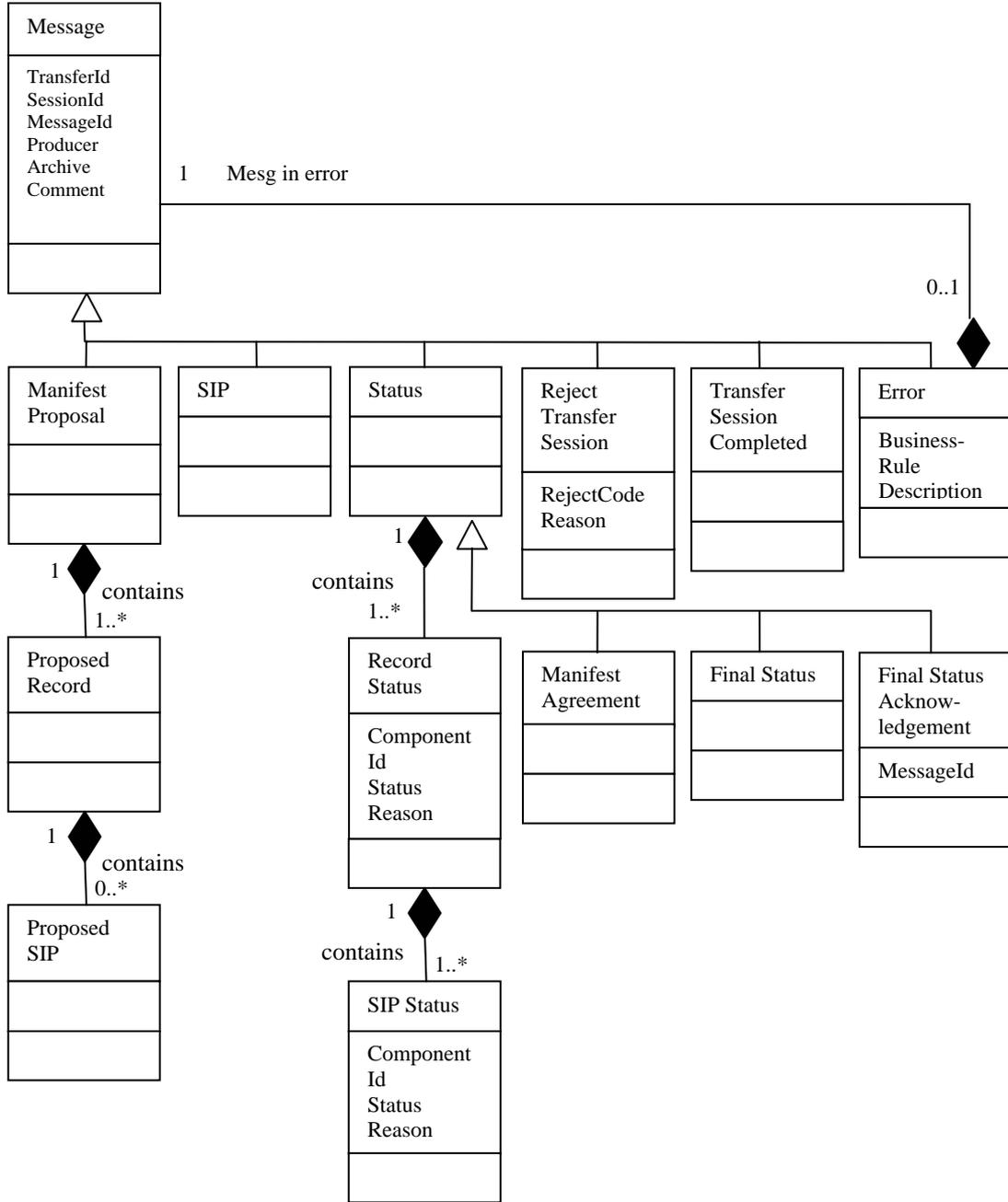
Either the producer or the archive sends an Error message when it receives a message that it does not expect, or a message which contains an error. A producer or an archive cannot send an Error message in response to an Error message. The circumstances under which Error messages are returned is given in the Business Rules section.

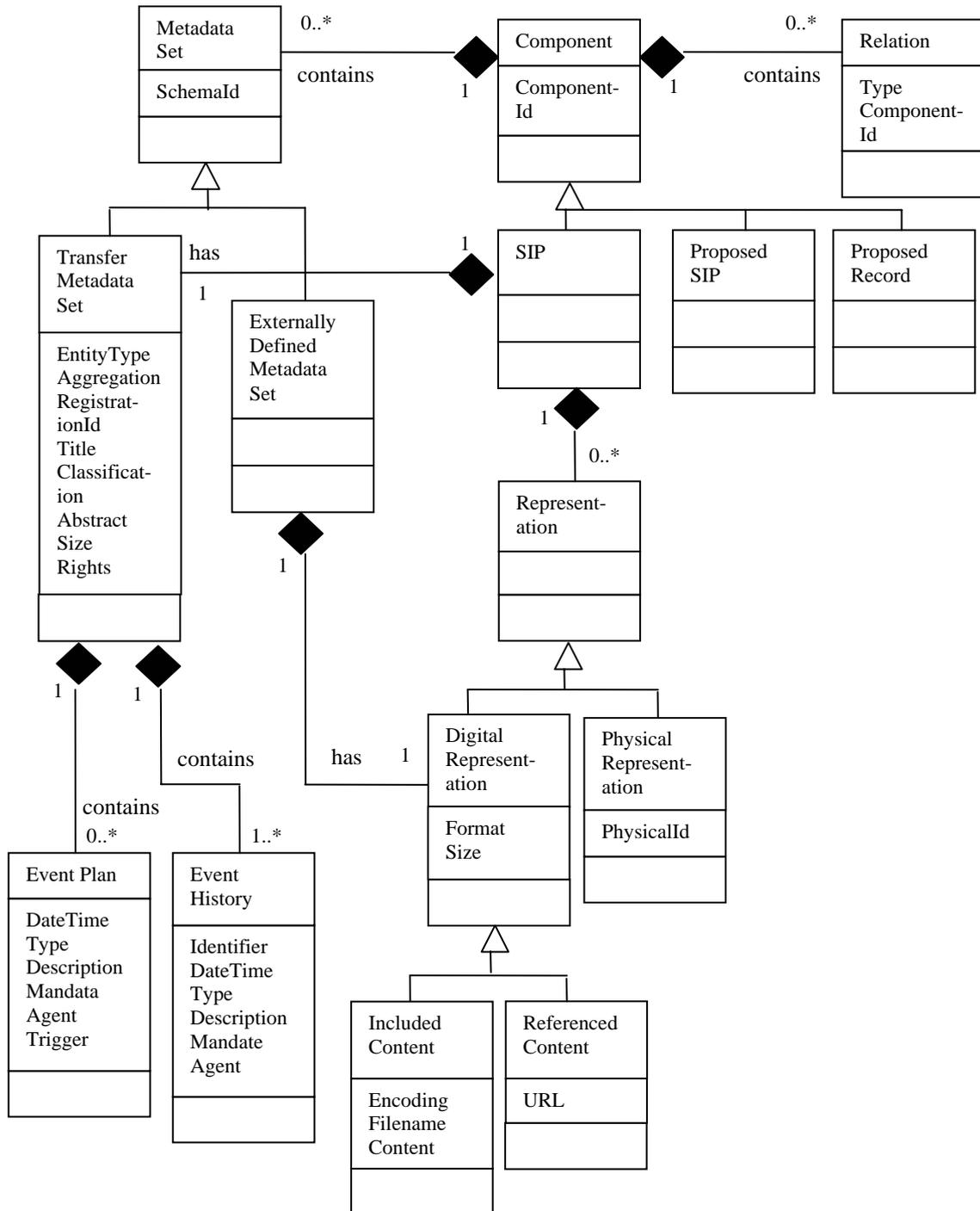


An Error message contains a brief description of the error that occurred, together with a copy of message that was in error.

### 5.3 Information model definition

The information model definition for the content of each flow of information (message) is shown in the following class diagrams.





The definition of each of these classes and attributes is as follows:

### 5.3.1 Message class

Each message exchanged in this business application is a subclass of the ‘Message’ class. This ensures that all messages have a consistent structure and top level data.

Subclass of			
Superclass of	Manifest Proposal, SIP, Status, Reject Transfer Session, Transfer Session Completed, Error		
Attributes			
Name	Number	Type	Description
Transfer Id	1..1	Str	(Transfer Agreement Identifier). This identifies the transfer agreement at the organizational level (i.e. the documentation of the organizational agreement to transfer records between the producer and the archive). This identifier will be assigned outside this protocol, but will first appear in the initial Manifest Proposal message and is thereafter used in every message relating to this transfer agreement. The combination of the TransferId and the SessionId form a unique identifier for this transfer session within the archive.
SessionId	1..1	Str	(Transfer Session Identifier). This identifies the transfer session within the broader transfer agreement. This identifier is assigned in the initial Manifest Proposal message, and is thereafter used in every message relating to this session.
MessageId	1...1	Str	(Message Identifier). This identifies the message within the transfer session. Message identifiers issued by the producer and the archive must be distinct. The message identifiers used must be monotonically increasing.
Producer	1..1	Str	A textual string identifying the producer transferring the records. The string may contain a code or a descriptive piece of text. It is envisaged that this attribute would be used to, for example, label reports or alerts for operators.
Archive	1..1	Str	A textual string identifying the archive receiving the records. The string may contain a code or a descriptive piece of text. It is envisaged that this attribute would be used to, for example, label reports or alerts for operators.
Comment	0..1	Str	A textual string containing a comment about the message.

The contents of the Message can be protected by an integrity mechanism<sup>17</sup>. The requirements of this integrity mechanism are as follows:

- Support for either a checksum (hash) or digital signature (i.e. encrypted hash value).
- Support for a range of algorithms.
- A mechanism for identifying the integrity algorithms used must be contained in the message.
- The integrity value (i.e. digital signature or checksum value) must be contained in the message.
- Any parameters required by the integrity algorithm must be contained in the message.
- In the case of a digital signature, the message should optionally include the certificates forming the certificate chain of the signer.
- All of the information in the Message must be covered by the integrity mechanism.

<sup>17</sup> These requirements are expected to be filled by general UN/CEFACT mechanisms.

### 5.3.2 Manifest Proposal class

The Manifest Proposal class represents the message proposing a set of records to be transferred from the producer to the archive in this transfer session.

<i>Subclass of</i>	Message		
<i>Superclass of</i>			
<i>Attributes</i>			
<i>Name</i>	<i>Number</i>	<i>Type</i>	<i>Description</i>
(None)			

A Manifest Proposal contains one or more Proposed Records, each of which contains zero or more Proposed SIPs. (The Proposed Records must contain Proposed SIPs if the Manifest Proposal is generated by the producer, but will not contain Proposed SIPs if the Manifest Proposal is generated by the archive.)

### 5.3.3 SIP class

The SIP class represents a message transferring zero or one Representations from the producer to the archive. A SIP may not contain any Representation if it contains part of a record that does not contain content (it may contain metadata and relationships).

<i>Subclass of</i>	Message, Component		
<i>Superclass of</i>			
<i>Attributes</i>			
<i>Name</i>	<i>Number</i>	<i>Type</i>	<i>Description</i>
(None)			

Each SIP has an associated TransferMetadataSet, and may contain zero or more Representations.

### 5.3.4 Transfer Session Completed class

The Transfer Session Completed class represents the message from the producer signalling that the producer considers this transfer session to be completed.

<i>Subclass of</i>	Message		
<i>Superclass of</i>			
<i>Attributes</i>			
<i>Name</i>	<i>Number</i>	<i>Type</i>	<i>Description</i>
(None)			

The Transfer Session Completed message purely exists to distinguish this message from the other types of message.

### 5.3.5 Status class

The Status class represents a message from the archive giving the current status of all records in the original proposal.

<i>Subclass of</i>	Message		
<i>Superclass of</i>	Manifest Agreement, Final Status, Final Status Acknowledgement		
<i>Attributes</i>			
<i>Name</i>	<i>Number</i>	<i>Type</i>	<i>Description</i>
(None)			

A Status message contains one or more Record Status.

### 5.3.6 Manifest Agreement class

The Manifest Agreement class is a subclass of Status and represents a Manifest Agreement message sent in response to a Manifest Proposal. The Agreement message responds to each record and SIP proposed in the Manifest Proposal and indicates whether the other party agrees to that record being transferred.

<i>Subclass of</i>	Status		
<i>Superclass of</i>			
<i>Attributes</i>			
<i>Name</i>	<i>Number</i>	<i>Type</i>	<i>Description</i>
(None)			

The Manifest Agreement class purely exists to distinguish the Manifest Agreement message from a generic Status message.

### 5.3.7 Final Status class

The Final Status class is a subclass of Status and is the acknowledgement by the archive of the Transfer Session Completed message. It may also be sent by the archive to signal that it is terminating the transfer session. The Final Status message gives the final status of all records and SIPs originally proposed for transfer in this transfer session. Once the archive generates the Final Status message it may no longer perform any processing on records in this transfer session.

<i>Subclass of</i>	Status		
<i>Superclass of</i>			
<i>Attributes</i>			
<i>Name</i>	<i>Number</i>	<i>Type</i>	<i>Description</i>
(None)			

The Final Status class purely exists to distinguish the Manifest Agreement message from a generic Status message. In particular, the Final Status message is necessary because otherwise the producer cannot distinguish between an ordinary (possibly

delayed) Status, and the acknowledgement that the archive has finished processing this transfer session.

### 5.3.8 Final Status Acknowledgement class

The Final Status Acknowledgement class is a subclass of Status and is the acknowledgement by the producer of the receipt of the Final Status message. It can be retained by the archive as evidence that the archive informed the producer of the custody status of every record transferred in this transfer session.

<i>Subclass of</i>	Status		
<i>Superclass of</i>			
<i>Attributes</i>			
<i>Name</i>	<i>Number</i>	<i>Type</i>	<i>Description</i>
MessageId	1..1	Str	The message identifier identifies the Final Status message being responded to

### 5.3.9 Reject Transfer Session class

The Reject Transfer Session represents a message rejecting the proposed transfer session entirely. It can be sent in response to a Manifest Proposal message.

<i>Subclass of</i>	Message		
<i>Superclass of</i>			
<i>Attributes</i>			
<i>Name</i>	<i>Number</i>	<i>Type</i>	<i>Description</i>
RejectCode	1..1	Str	An identifier indicating the reason the transfer session was rejected. The standard values are: <ul style="list-style-type: none"> <li>NoSuchTransfer. The transfer id does not identify a known transfer agreement.</li> <li>NoSuchTransferSession. The SessionId does not identify a known transfer session within this transfer agreement.</li> </ul> Other values may be used as required.
Reason	1..1	Str	A textual string describing the reason that the relocation was rejected.

### 5.3.10 Error class

The Error represents a message describing an error in the protocol that has been detected. It may be sent by either the producer or archive.

<i>Subclass of</i>	Message		
<i>Superclass of</i>			
<i>Attributes</i>			
<i>Name</i>	<i>Number</i>	<i>Type</i>	<i>Description</i>
BusinessRule	1..1	Str	The number of the business rule that was violated.
Description	1..1	Str	A textual description of the error that occurred. See the business rules (Section 6) for the allowed values

Each Error contains a copy of the message that was in error. This allows the receiver to identify precisely the error that occurred and resolve it.

### 5.3.11 Record Status class

The Record Status class contains the status of a record being transferred to the archive.

<i>Subclass of</i>			
<i>Superclass of</i>			
<i>Attributes</i>			
<i>Name</i>	<i>Number</i>	<i>Type</i>	<i>Description</i>
ComponentId	1..1	Str	(Component identifier). The identifier of the Component. This identifier must be identical to that transferred in the Manifest Proposal.
Status	1..1	Str	The current status of the Record at the archive. This may be: <ul style="list-style-type: none"> <li>• Rejected for transfer</li> <li>• Agreed to be transferred</li> <li>• Received by archive</li> <li>• Rejected, resubmit</li> <li>• Rejected, correct and resubmit</li> <li>• Rejected, do not resubmit</li> <li>• Custody accepted</li> </ul>
Reason	0..1	Str	A free text description of the reason for the current status. For example, if the status was 'Rejected, do not resubmit', the reason might be 'Representation format '.dln' not supported'.

### 5.3.12 SIP Status class

The SIP Status class contains the status of a component within a record being transferred to the archive.

<i>Subclass of</i>			
<i>Superclass of</i>			
<i>Attributes</i>			
<i>Name</i>	<i>Number</i>	<i>Type</i>	<i>Description</i>
ComponentId	1..1	Str	(Component identifier). The identifier of the Component. This identifier must be identical to that transferred in the Manifest Proposal.
Status	1..1	Str	The current status of the SIP at the archive. This may be: <ul style="list-style-type: none"> <li>• Not yet received</li> <li>• Received by archive</li> <li>• Rejected, resubmit</li> <li>• Rejected, correct and resubmit</li> <li>• Rejected, not included in Transfer Agreement</li> <li>• Finalized</li> </ul>
Reason	0..1	Str	A free text description of the reason for the current status. For example, if the status was 'Rejected, do not resubmit', the reason might be 'Not listed in Manifest Agreement'.

### 5.3.13 Component class

A Component class is a superclass representing information about components of records.

<i>Subclass of</i>			
<i>Superclass of</i>		Proposed Record, Proposed SIP, SIP	
<i>Attributes</i>			
<i>Name</i>	<i>Number</i>	<i>Type</i>	<i>Description</i>
ComponentId	1..1	Str	(component identifier). The component identifier uniquely identifies this component in the context of the producer's records system. The syntax of this component identifier is defined by the producer's system. The value may be textual or binary, it may be of fixed or arbitrary length, and it may be a structured XML data. Note that this identifier is unlikely to be globally unique. To be globally unique, this identifier would normally need to be qualified by an identifier for the producer and records system within the producer.

Each component may have one or more Metadata Sets associated with it to contain information about the component and one or more Relations (which link this component with other components).

### 5.3.14 Metadata Set class

A Metadata Set contains a collection of metadata about a component. A component may contain many independent Metadata Sets. Each Metadata Set contains metadata elements drawn from one metadata standard. The intention is to allow the component to be described by metadata drawn from many different standards, for example, Dublin Core, AGLS, EAD, PREMIS, etc. The metadata drawn from each standard is collected together into one Metadata Set. Note that an individual piece of information may appear multiple times; for example most metadata schemes include a mandatory title.

<i>Subclass of</i>			
<i>Superclass of</i>		Transfer Metadata Set, Externally Defined Metadata Set	
<i>Attributes</i>			
<i>Name</i>	<i>Number</i>	<i>Type</i>	<i>Description</i>
Schema Identifier	1..1	Str	This attribute identifies the standard that controls the metadata elements that may be contained within the metadata set (e.g. AGLS). The value of this element could be a textual name, or the URL/URI to a published scheme.

There are two subclasses of Metadata Set. The Transfer Metadata Set is a small set of metadata intended to be required by the transfer process itself. The Externally Defined metadata set is a container for any other Metadata Set.

### 5.3.15 Transfer Metadata Set class

The Transfer Metadata Set is a subclass of Metadata Set. It contains a minimal set of metadata elements that are intended to be used during the transfer session. It does not contain metadata that is intended to be used in other record activities (e.g. preservation, or access). It is not expected that all components will have an associated Transfer Metadata Set. The metadata is based on that listed in Section 9 of Metadata for records – Part 2: Conceptual and implementation issues (ISO/TS 23081-2:2007).

The metadata contained in a Transfer Metadata Set may be duplicated in External Metadata Sets included in the SIP, but because these are not standardized in this specification, the metadata could be located anywhere and be named anything.

<i>Subclass of</i>	Metadata Set		
<i>Superclass of</i>			
<i>Attributes</i>			
<i>Name</i>	<i>Number</i>	<i>Type</i>	<i>Description</i>
Entity Type	0..1	Str	This identifies the type of the component (e.g. record, agent) as defined in Section 6 of ISO/TS 23081-2:2007
Aggregation	0..1	Str	This identifies the aggregation level of the component as defined in Section 7.1.2 of ISO/TS 23081-2:2007
Registration Identifier	1..1	Str	This uniquely identifies the component within the producer's records system
Title	0..1	Str	The title of the component
Classification	0..*	Str	Information about the classification of the entity in accordance with an authorized source
Abstract	0..*	Str	Information that describes the component
Rights	0..*	Str	The rights applied to the component. These must be expressed in such a fashion that the rights could be applied by the archive. The ability to transfer access control information is required by MoReq2 section 5.3. It is conditional in this standard because not all applications will use this functionality.
Size	1..1	Str	The nominal size of the component. Note that the size of a component is purely nominal. It is intended as a piece of descriptive information that the archive could use in planning to receive the component.

A Transfer Metadata Set may have one or more Event Plans associated with it to contain information about plans for the component, and will have one or more Event Histories which document the events that occurred during the transfer session.

### 5.3.16 Event History class

The Event History class describes an event in the history of this component. The collection of events documents the history of the component during (at least) the transfer. The history must consist of at least one event (the inclusion of the record in the SIP message). The metadata is based on that listed in Section 9 of Metadata for records – Part 2: Conceptual and implementation issues (ISO/TS 23081-2:2007).

<i>Subclass of</i>			
<i>Superclass of</i>			
<i>Attributes</i>			
<i>Name</i>	<i>Number</i>	<i>Type</i>	<i>Description</i>
Identifier	1..1	Str	Unique identifier for the event or the event transaction number
DateTime	1..1	Str	The date and time of the event expressed using the W3C 'Date and Time Formats' profile of ISO8601
Type	1..1	Str	The type of event that occurred
Description	0..1	Str	A textual description of the event
Mandate	0..*	Str	Information about the mandate or instrument that provides the legal or administrative basis for the action taken.
Agent	1..1	Str	Information about the person responsible for undertaking or authorizing the event

### 5.3.17 Event Plan class

The Event Plan class describes a management action that is planned to occur in the future. The particular management action expected to be described is a Disposition Authority as the ability to transfer Disposition Authorities is required by MoReq2 section 5.3.

This class is conditional in this standard because not all components have disposition authorities (e.g. files might be covered by a disposition authority, but a record within the file might have the same disposition as the file), and not all applications will use this functionality.

The metadata is based on that listed in Section 9 of Metadata for records – Part 2: Conceptual and implementation issues (ISO/TS 23081-2:2007).

<i>Subclass of</i>			
<i>Superclass of</i>			
<i>Attributes</i>			
<i>Name</i>	<i>Number</i>	<i>Type</i>	<i>Description</i>
DateTime	1..1	Str	The date and (optionally) time that the action is intended to occur expressed using the W3C 'Date and Time Formats' profile of ISO8601
Type	1..1	Str	The type of action to perform
Description	0..1	Str	Information required by the agent to carry out the planned action
Mandate	0..*	Str	Information about the mandate or instrument that provides the legal or administrative basis for the action
Agent	1..1	Str	Information about the person responsible for undertaking or authorizing the action
Trigger	1..1	Str	Description of the trigger event.

### 5.3.18 Externally Defined Metadata class

The Externally Defined Metadata Set is a subclass of Metadata Set. The Externally Defined metadata set is a container for any other Metadata Set.

<i>Subclass of</i>	Metadata Set		
<i>Superclass of</i>			
<i>Attributes</i>			
<i>Name</i>	<i>Number</i>	<i>Type</i>	<i>Description</i>
(None)			

The Externally Defined Class refers to a Digital Representation which contains the encoded metadata.

### 5.3.19 Relation class

A Relation is a link between two components in a records system (links outside the producer's records system are not allowed). Relations may of any type. A link from one component to another implies the reverse link<sup>18</sup>, however it is not an error for both links to be described. Related components need not be contained within the one transfer session<sup>19</sup>.

<i>Subclass of</i>			
<i>Superclass of</i>			
<i>Attributes</i>			
<i>Name</i>	<i>Number</i>	<i>Type</i>	<i>Description</i>
Type	1..1	Str	The type of the relation. The key relation is 'contained'/'containing' which is used to link between hierarchically arranged components. Any other relationship type may be defined by the producer and archive. The type is a textual string describing the type.
ComponentId	1..1	Str	The identifier of the related component. See the definition of 'ComponentId' in the Component class for the possible values of this attribute. The ComponentId is explicitly included in the Relation to allow relationships with components that are not transferred within this transfer session.

### 5.3.20 Proposed Record class

A Proposed Record class represents a record proposed to be transfer to the archive. A record proposal contains zero or more Proposed SIPs.

<sup>18</sup> That is, a link from A to B implies a link from B to A. This is intended to minimize the overhead of managing links. For example, Files and their contained Items can be linked by a 'Contained in' relation from the Item to the File. This means that Files do not need to explicitly list their contents, and so the addition of a new item to a file does not require the File information to be changed. A similar argument applies for files and series. The bi-directional links can be constructed by an index of links at the archive.

<sup>19</sup> For example, a File (and its contained Records) could be transferred in one transfer session. The next transfer session could transfer additional Records in that File. It is for this reason that a Relation explicitly includes the Component Id of the linked component rather than directly linking components using the UML model.

<i>Subclass of</i>	Component		
<i>Superclass of</i>			
<i>Attributes</i>			
<i>Name</i>	<i>Number</i>	<i>Type</i>	<i>Description</i>
(None)			

A Proposed Record contains zero or more Proposed SIPs. The Proposed Records must contain Proposed SIPs if the Manifest Proposal is generated by the producer, but will not contain Proposed SIPs if the Manifest Proposal is generated by the archive.

### 5.3.21 Proposed SIP class

A Proposed SIP class represents a SIP that will hold a component of a record during transmission to the archive.

<i>Subclass of</i>	Component		
<i>Superclass of</i>			
<i>Attributes</i>			
<i>Name</i>	<i>Number</i>	<i>Type</i>	<i>Description</i>
(None)			

### 5.3.22 Representation class

This class represents the content<sup>20</sup> of a SIP.

A SIP may contain zero or more representations. For example, a SIP might contain two different representations of the same content, an Open Document Format version and a PDF version. Equally, a SIP may not contain a representation at all if it does not contain digital content (e.g. if it represents a compound object such as a fonds, or it represents a record with physical content).

Representation content may be included in the SIP, or referenced by a URL/URI.

There are two subclasses of Representation: Digital Representation, and Physical Representation. A Digital Representation contains a digital object, while a 'Physical' Representation is a surrogate for a physical (e.g. paper) representation.

<i>Subclass of</i>			
<i>Superclass of</i>	Digital Representation, Physical Representation		
<i>Attributes</i>			
<i>Name</i>	<i>Number</i>	<i>Type</i>	<i>Description</i>
(None)			

### 5.3.23 Digital Representation class

This class is a subclass of Representation. It represents a digital object.

<sup>20</sup> The content of a record can be represented as a digital object constructed according to another specification (e.g. METS). In this case, the representation could include metadata as well as actual content.

There are two subclasses of Digital Representation: Included Content, and Referenced Content. An Included Content representation is included within the SIP message. A Referenced Content representation is not included within the SIP message, but is accessed by a URL/URI.

<i>Subclass of</i>	Representation		
<i>Superclass of</i>	Included Content, Referenced Content		
<i>Attributes</i>			
<i>Name</i>	<i>Number</i>	<i>Type</i>	<i>Description</i>
Format	1..1	Str	The format of the representation (e.g. Open Document Format, PDF, etc). The encoding of the representation is indicated by a schema attribute as follows: <ul style="list-style-type: none"> <li>• MIME: The value of this attribute is a MIME type (see the IANA list <a href="http://www.iana.org/assignments/media-types/">http://www.iana.org/assignments/media-types/</a> for a list of defined object types).</li> <li>• The identifier of the format in any format registry (e.g. a PRONOM unique identifier issued by The National Archives UK).</li> </ul>
Size	1..1	Int	The size in bytes (of 8 bits) of the representation before any encoding (e.g. Base64) was applied

### 5.3.24 Physical Representation class

This class is a subclass of Representation. It represents physical content (e.g. paper records).

<i>Subclass of</i>	Representation		
<i>Superclass of</i>			
<i>Attributes</i>			
<i>Name</i>	<i>Number</i>	<i>Type</i>	<i>Description</i>
Physical Id	1..1	Str	This is a physical identifier (e.g. a barcode) identifying the physical representation of the record.

### 5.3.25 Included Content class

This class is a subclass of Digital Representation. It represents digital content that is included verbatim within the SIP Message being sent from the producer to the archive. The included content may be binary data or textual data.

<i>Subclass of</i>	Digital Representation		
<i>Superclass of</i>			
<i>Attributes</i>			
<i>Name</i>	<i>Number</i>	<i>Type</i>	<i>Description</i>
Encoding	1..1	Str	The mechanism of encoding the raw content for inclusion as textual content within the message. Valid values are <ul style="list-style-type: none"> <li>Base64: the binary content has been encoded into text using the Base64 encoding defined in Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies, section 6.8, Base64 Content-Transfer-Encoding, IETF RFC 2045</li> <li>XMLescaped: the textual content has been encoded to eliminate the XML reserved characters (primarily '&lt;', '&gt;' and '&amp;')</li> <li>None: the textual content is already XML (or XML safe)</li> <li>(other values may be added in the future).</li> </ul>
Filename	0..1	Str	The name of the computer file that contained the content
Content	1..1	Str	This value of this attribute is the included content itself (e.g. the Base64 encoded binary, or the textual information)

### 5.3.26 Referenced Content Class

This class is a subclass of Digital Representation. It represents digital content that is NOT included verbatim within the SIP Message being sent from the producer to the archive. Instead the Representation contains a URL from which the content can be retrieved. It is expected that the archive will retrieve the referenced content using the URL before signalling that it has accepted custody of the record. This behaviour, however, is not required by the standard.

<i>Subclass of</i>	Digital Representation		
<i>Superclass of</i>			
<i>Attributes</i>			
<i>Name</i>	<i>Number</i>	<i>Type</i>	<i>Description</i>
URL	1..1	Str	The URL/URI of the referenced content

## 6 Business rules

Rule	Description
1.	TransferId element All messages in a transfer session will contain the same TransferId element. Several transfer sessions within the same transfer agreement may have the same TransferId
2.	TransferId element Any messages received with an invalid TransferId element will result in an Error message being sent with the description 'Invalid TransferId'.
3.	SessionId element All messages in a transfer session will contain the same SessionId element. The SessionId element will be different for different transfer sessions within a transfer agreement. The combination of the TransferId and the SessionId will uniquely identify this session within the Producer/Archive context.

4.	<p>SessionId element</p> <p>Any messages received with an invalid SessionId element will result in an Error message being sent with the description 'Invalid SessionId'.</p>
5.	<p>ComponentId element</p> <p>A component will be represented by the same component identifier in the Manifest Proposal, SIP, and Status (including Manifest Agreement, Final Status, and Final Status Acknowledgement) messages</p>
6.	<p>Manifest Proposal</p> <p>If a duplicate Manifest Proposal message is received, a duplicate response (a Manifest Agreement or Reject Transfer Session as the case may be) must be returned.</p>
7.	<p>Manifest Proposal</p> <p>If a non-duplicate Manifest Proposal messages is received, an Error is sent with the description 'A Manifest Proposal has already been received. This Manifest Proposal is different to that originally received.'</p>
8.	<p>Manifest Proposal – SIP – Transfer Session Completed</p> <p>If the producer receives a duplicate Manifest Proposal message after sending SIP or Transfer Session Completed messages, it must resend the SIP or Transfer Session Completed messages.</p>
9.	<p>Manifest Proposal – Manifest Agreement – Reject Transfer Session</p> <p>The response to a Manifest Proposal message must be either a Manifest Agreement or Reject Transfer Session message.</p> <p>If, after sending a Manifest Proposal, any message is received other than a 'Manifest Agreement' or 'Reject Proposal', an Error is sent with the description 'A Manifest Proposal has been sent, awaiting 'Manifest Agreement or Reject Proposal, received this message instead'</p>
10.	<p>Manifest Proposal – Manifest Agreement – Reject Transfer Session</p> <p>If a Manifest Agreement or Reject Transfer Session message is not received within a time frame agreed upon in the transfer agreement, the Manifest Proposal message must be retransmitted exactly as originally sent</p>
11.	<p>Manifest Agreement</p> <p>If a duplicate Manifest Agreement message is received it must be discarded without further processing.</p>
12.	<p>Manifest Agreement</p> <p>If a non-duplicate Manifest Agreement message is received, an Error is sent with the description 'A Manifest Agreement has already been received. This Manifest Agreement is different to that originally received.'</p>
13.	<p>Reject Transfer Session</p> <p>If a duplicate Reject Transfer Session message is received it must be discarded without further processing.</p>
14.	<p>Reject Transfer Session</p> <p>If a non-duplicate Reject Transfer Session message is received, an Error is sent with the description 'A Reject Transfer Session has already been received. This Reject Transfer Session is different to that originally received.'</p>
15.	<p>Manifest Agreement – SIP</p> <p>A SIP message must not be sent prior to the receipt or transmission of a</p>

	Manifest Agreement message
16.	SIP If a SIP is received that is not in the manifest agreement, an Error is sent with the description 'This SIP is not listed in the Manifest Agreement'
17.	SIP If a non-duplicate SIP message is received, an Error is sent with the description 'This SIP has already been received. This SIP is different to that originally received.'
18.	Status element Once a record has been marked as 'Custody accepted' in a Status message, its status cannot change
19.	Status If a status message is received with a MessageId element that has a value less than the last Status message received, it is discarded without any further processing.
20.	SIP – Transfer Session Completed A SIP message must not be transmitted after a Transfer Session Completed message has been transmitted. If a SIP message is received after a Transfer Session Completed message has been received, an Error must be sent with the description 'This SIP was received after receipt of a Transfer Session Completed'.
21.	Status – Transfer Session Completed A Status message, except a Final Status message, must not be transmitted after the receipt of a Transfer Session Completed message
22.	Transfer Session Completed – Final Status If a Final Status message is not received within a time frame agreed in the transfer agreement, the Transfer Session Completed message must be resent.
23.	Transfer Session Completed If a duplicate Transfer Session Completed message is received before a Final Status message has been sent, the duplicate Transfer Session Completed must be discarded without further processing.
24.	Transfer Session Completed If a duplicate Transfer Session Completed message is received after a Final Status message has been sent, a duplicate Final Status message must be sent.
25.	Transfer Session Completed If a non-duplicate Transfer Session Completed message is received, an error must be sent with the description 'A Transfer Session Completed has already been received. This Transfer Session Completed is different to that originally received.'
26.	Transfer Session Completed – Final Status The response to every Transfer Session Completed message must be a Final Status message
27.	Final Status – Final Status Acknowledgement If a Final Status Acknowledgement message containing a Message Id matching that of the Final Status is not received within a time frame agreed in the transfer agreement, the Final Status message must be resent.

28.	Final Status – Final Status Acknowledgement If the MessageId in a Final Status Acknowledgement message does not match that of the Final Status, an error must be sent with the description ‘The MessageId in this Final Status Acknowledgement does not match that in the Final Status message sent.’
29.	Final Status If a duplicate Final Status message is received, a duplicate Final Status Acknowledgement message must be sent.
30.	Final Status If a non-duplicate Final Status message is received, an error must be sent with the description ‘A Final Status has already been received. This Final Status is different to that originally received.’
31.	Final Status Acknowledgement If a duplicate Final Status Acknowledgement is received, it must be discarded without any further processing
32.	Final Status Acknowledgement If a non-duplicate Final Status Acknowledgement message is received, an error must be sent with the description ‘A Final Status Acknowledgement has already been received. This Final Status Acknowledgement is different to that originally received.’

## 6.1 Definition of terms

*Archive:* Role played by those organizations, persons, or systems, who receive information from a producer in order to take custody of it. Such role is typically played by an archival body or a secondary storage supplier, but may also be played by an agency to which records are transferred back from a secondary storage supplier or from another agency whose mandate has expired. (based on OAIS Model ISO 14721:2003)

*Custody:* Charge of and responsibility for on-going preservation and access.

*Digital Records:* Records in which the content and metadata are represented in digital form.

*Export:* Process of producing a copy of records, along with their metadata, extracted from a records system to be included in one or more Submission Information Packages.

*Import:* Process of producing a copy of records, along with their metadata, extracted from one or more Submission Information Packages to be included in a records system.

*Manifest:* List of records that are to be transferred in a transfer session.

*Metadata:* Data describing context, content and structure of records and their management through time. (ISO 15489-1)

*Producer:* Role played by those organizations, persons, or systems, who provide information to an archive for this archive to take custody of it. Such role is typically played by a creating body, but may also be played by an archival body or a secondary storage supplier which transfers back records to a creating body.

*Records*: Information created, received, and maintained as evidence and information by an organization or person, in pursuance of legal obligations or in the transaction of business. (ISO 15489-1)

*Records system*: Computer system which captures, manages, and provides access to records through time. (based on ISO 15489-1)

*Submission*: see Transfer.

*Submission Information Package (SIP)*: Information package, including data to be preserved and associated metadata, which is delivered by a producer to an archive within a transfer session (based on OAIS Model ISO 14721:2003)

*Transfer*: Process of changing custody from a producer to an archive. This includes tasks ranging from the extremely high level (e.g. transfer agreement), to the extremely low level (e.g. physical relocation of the records).

*Transfer agreement*: agreement reached between an archive and a producer, which specifies what is to be transferred and how the transfer will be carried out.

*Transfer session*: The activities involved in administering and performing an actual transfer within a transfer agreement.