MODEL REQUIREMENTS FOR
THE MANAGEMENT OF ELECTRONIC RECORDS

UPDATE AND EXTENSION, 2008

MoReq2 SPECIFICATION

This specification has been prepared for the European Commission by Serco Consulting with funding from the IDABC programme

MODEL REQUIREMENTS FOR
THE MANAGEMENT OF ELECTRONIC RECORDS

UPDATE AND EXTENSION, 2008

MoReq2 SPECIFICATION

This specification is available in electronic form at the following urls:

www.dlm-network.org/moreq2

http://ec.europa.eu/transparency/archival_policy

and other websites. Translations into selected languages are expected to be available at these sites.

It is also available in paper form from the Office for Official Publications of the European Communities as INSAR Supplement VIII.

© CECA-CEE-CEEA, Bruxelles- Luxembourg, 2008

Reproduction autorisée, sauf à des fins commerciales, moyennant mention de la source.

Reproduction is authorised, except for commercial purposes, provided the source is acknowledged.

Legal notice: The copyright of this publication is owned by the European Communities. The European Commission does not guarantee the accuracy of the information included in this report, nor does it accept any responsibility for any use made thereof. Neither the European Communities and/or their institutions nor any person acting on their behalf shall be held responsible for any loss or damage resulting from the use of this publication.

Contents

PREFACE: MOREQ2

Update and extension of the
Model Requirements for the management of electronic records

Since it was first published in 2001, the original MoReq – Model Requirements for the management of electronic records – has been used widely throughout Europe and beyond. Throughout the European Union, prospective users of electronic records management have recognised the value of using a model specification such as MoReq as the basis for procuring Electronic Records Management Systems and software suppliers have responded by using MoReq to guide their development process.

MoReq is now regarded as an unqualified success. It has been cited many times on many continents and it has a central role on the electronic records management scene.

However, information technology has changed since 2001. There has been growth and evolutionary change in many technology areas that affect the creation, capture and management of electronic records. This new version of MoReq, called MoReq2, addresses the impacts of that technological change. It also takes account of new standards and best practice that have been developed over the last several years. Accordingly, it is written as an evolutionary update of the original MoReq.

MoReq2 for the first time also allows for a software testing regime to be implemented. It is written specifically to support the execution of independent compliance testing and a suite of compliance tests has been developed and published in parallel with the model requirements themselves. The need for rigorously-worded, testable, requirements has led to many changes of wording and expression in MoReq2.

Finally, the years of experience in using and applying MoReq has pointed out the need for national variations, to take into account different national languages, legislation, regulations, and record keeping traditions. For this reason, MoReq2 introduces for the first time a moderated mechanism – called ”chapter zero” – to allow member states to add their unique national requirements.

MoReq2 was prepared for the European Commission by Serco Consulting with financing from the European Union's IDABC programme. The development process was overseen by the European Commission working closely with the DLM Forum and drafts were reviewed by DLM Forum experts at key stages in the development. These reviews were in addition to input and review by dozens of users, consultants, suppliers, academics and professional bodies from around the globe, giving MoReq2 an unprecedented level of authority. As such MoReq2 will be of great value to all those involved in the management of electronic records in Europe and around the world.

1. INTRODUCTION

1.1 Background

The need for a comprehensive specification of requirements for electronic records management was first articulated by the DLM-Forum[1] in 1996 as one of its ten action points. Subsequently, the European Commission's IDA (Interchange of Data between Administrations) programme commissioned the development of a model specification for electronic records management systems (ERMSs). The result, MoReq, the Model Requirements for the management of electronic records[2], was published in 2001.

MoReq was widely used in throughout the European Union and beyond. However, there was no maintenance regime for MoReq; and there was no scheme to test software compliance against the MoReq specification.

Demand for both updates to MoReq and a compliance testing scheme grew. The DLM Forum entered into discussions with the European Commission. This culminated in the Commission's Secretariat-General (Directorate B e-Domec and archives) launching an open competition for the development of this document, MoReq2, in 2006. Development was carried out during 2007 by a small team of specialist consultants from Serco Consulting (formerly Cornwell Management Consultants plc), supported by an Editorial Board of experts drawn from several countries, and numerous volunteer reviewers from both the private and public sectors.

Appendix 2 contains further detail on the methodology used, and appendix 4 acknowledges the contributions of the review panel members who kindly volunteered their time, intellect, and experience.

1.2 Relationship between MoReq and MoReq2

MoReq2 is intended to replace MoReq.

The specification for MoReq2 is contained in the "Scoping Report[3]" for MoReq2. It describes the aims of MoReq2 as follows:

"The overall aims for the MoReq2 development are to develop extended functional requirements within a European context, and to support a compliance scheme by:

- Strengthening from MoReq what have in the interim become key areas and covering important new areas of requirements with clarity;
- Ensuring that the functional requirements are testable and developing test materials to enable products to be tested for compliance with the requirements;

---

1 DLM is an acronym for "Document Lifecycle Management" (it formerly was an acronym for the French "Données Lisibles par Machine," in English: "machine-readable data.") The DLM-Forum is based on the conclusions of the European Council (94/C 235/03) of 17 June 1994 concerning greater cooperation in the field of archives.

2 MoReq is available from http://www.DLM-Network.org. It is also published in paper form, with ISBN 92-894-1290-9.

3 "The Scoping Report for MoReq2" is available from http://www.DLM-Network.org.

- Making the requirements modular to assist application in the various environments in which they will be used."

"To provide compatibility, MoReq2 is to be an evolutionary update to the original MoReq, not a radically different product."

The concept of "evolutionary upgrade" is key. MoReq2 is almost entirely compatible with MoReq (minor incompatibilities are clearly indicated); it is based on the same concepts, and as a document it uses a similar structure.

1.3 Purpose and Scope of this Specification

This specification is the second version of the Model Requirements for the management of electronic records. (MoReq2). It focuses mainly on the functional requirements for the management of electronic records by an Electronic Records Management System (ERMS).

This specification is written to be equally applicable to public and private sector organisations which wish to introduce an ERMS, or which wish to assess the ERMS capability they currently have in place.

While the specification focuses on functional requirements, it recognises that non-functional attributes are central to the success of an ERMS, as with any information system. However, these non-functional attributes vary enormously between environments. Accordingly, they are identified but described only in outline.

Other closely-related requirements, such as document management and the electronic management of physical records (such as paper files and microfilm) are also addressed, but in less detail. Related issues such as digitisation and other means of creating electronic records are outside the scope of this specification. Similarly, it makes no attempt to cover the practical implementation of an ERMS.

This specification is written with the assumption that ERMS users include not only administrators, records managers or archivists, but also general office and operational staff who use ERMSs as part of their everyday work while creating, receiving and retrieving records.

As this specification contains "model" requirements, it is designed to be entirely generic. It does not consider any platform-specific or sector-specific issues. Because it is modular, user communities can add to it additional functionality specific to their own business requirements (see section 1.6 and appendix 3 for guidance on using and customising this specification).

1.4 What is an ERMS?

An ERMS is primarily an application for managing electronic records, though it may also be used to manage physical records. The emphasis of this specification is firmly on the management of electronic records.

The management of electronic records is complex, requiring a large range of functionality, meeting business needs, to be implemented well. Typically, a system to meet these needs – an ERMS – requires specialised software, though increasingly records management functionality is being built into operating system software and other applications. Specialist software may consist of a single package, a number of integrated packages, custom-designed software or some combination; and in all cases, there will be a need for complementary manual procedures and management policies. The nature of an ERMS will vary from organisation to organisation. This specification makes no

assumption about the nature of individual ERMS solutions. Users of this specification will need to determine how the functionality of an ERMS can be implemented to meet their requirements.

ERMSs are expected to be used over considerable periods and increasingly to interact with other applications. There are therefore many ways in which an implementer may want to connect an ERMS with other software applications. It may be necessary to create interfaces for the capture of individual records from other business applications (see section 6.1) and for the applications to access records in the ERMS (see section 4.1). This applies particularly with business applications such as CRM (Customer Relationship Management) and line of business applications.

Chapter 10 includes specific coverage of interfaces with CMSs (Content Management Systems), Workflow and Casework systems and fax integration. Chapter 6 covers interfaces with e-mail applications in section 6.3 (e-mail management) and scanning and imaging in section 6.5. An interface for validation of metadata is covered in section 6.1 (Capture) and with report generators in 8.3 (Printing).

MoReq2 is written primarily to describe application software that is designed expressly to manage records. However, it may also be used as a statement of outcomes together constituting electronic records management. Thus the statements in MoReq2 saying "The ERMS must or should…" may also be read as shorthand for "The using organisation's application system and/or the supplier platform must or should…" Readers of MoReq2 need to decide which requirements are necessary in their environment.

The full set of MoReq2 requirements may be appropriate for integrated application systems. However, a subset may be more appropriate in the situation, for example, where records management features are needed as part of a case management or line of business application.

The optional modules 10.4 Workflow and 10.5 Casework, specifically apply to line of business applications. However much of the functionality described in the requirements throughout MoReq2 can also be applicable and should be considered when implementing these business systems.

1.5 For what can this Specification be used?

The MoReq2 specification is intended to be used:

- **by potential ERMS users:** as a basis for preparing an invitation to tender;
- **by ERMS users:** as a basis for auditing or checking an existing ERMS;
- **by training organisations:** as a reference document for preparing records management training, and as course material;
- **by academic institutions:** as a teaching resource;
- **by ERMS suppliers and developers:** to guide product development by highlighting functionality required;
- **by record management service providers:** to guide the nature of the services to be provided;
- **by potential users of outsourced record management services:** as an aid in specifying the services to be procured.

In addition, when used with the testing framework documentation developed in parallel with MoReq2, it is intended to be used:

- **by ERMS suppliers and developers:** to test ERMS solutions for MoReq2 compliance;
- **by ERMS users:** to test ERMS implementations for MoReq2 compliance.

The specification is written with an emphasis on usability. Throughout, the intention has been to develop a specification which is useful in practice.

1.6 Intellectual Property Rights

All intellectual property rights in MoReq2, including use of the name MoReq2, lie with the European Commission. Accordingly, permission must be given before any translation of MoReq2 or chapter zero to MoReq2 is published – see the formal notice on the title page. To apply for permission, refer to the DLM Forum website at http://www.DLM-Network.org.

1.7 Emphasis and Limitations of this Specification

The MoReq2 specification is designed explicitly with pragmatism and usability in mind. It is primarily intended to serve as a practical tool in helping organisations meet their business needs for the management of both computer-based and paper-based records. While its development has taken traditional archival science and records management disciplines into account, these have been interpreted in a manner appropriate to electronic environments. Thus, MoReq was developed with the needs of managers of both electronic and physical records in mind.

The requirements in MoReq2 should, if implemented, result in a system which will manage electronic records with the desired levels of confidence and integrity, by combining both the advantages of electronic ways of working with classical records management theory. Examples of this pragmatic approach include the incorporation of requirements for document management, workflow, metadata and other related technologies.

Although MoReq2 covers a wide range of types of records, it is important to understand that ERMS solutions address mainly records that are often referred to as "unstructured" records[4]. . In simple terms, unstructured records are those that contain information presented in a form primarily intended to be used by human users. Examples of unstructured records are letters, memoranda, e-mail messages, pictures, photocopies, scanned images, audio recordings and video recordings. Structured records by contrast contain information in a form intended to be used primarily by computer applications (examples include accounting system records, manufacturing scheduling system records, and air traffic control system records). While an ERMS can, in principle be used to store such structured records, it rarely is. In most situations, structured data is stored under the management of a data processing application (in the examples above these might be a general ledger system, a manufacturing scheduling system, and an air traffic control system). ERMS solutions are used almost universally to store and manage unstructured records. The instances in which an ERMS is used for structured records occur often in case management environments – see section 10.5.

MoReq2 does not cover the practical aspects of the management of records. Intentionally, the specification addresses only the capabilities required for the management of electronic records by software. The specification avoids discussion of records management philosophy, archival theory, decision taking, management control etc.; these issues are well covered in other literature, some of

[4] It can be held that all properly managed electronic records are structured, as they all are linked to metadata, audit trail data etc. in a structured manner. On this basis it would be more accurate to refer to unstructured records as "records containing unstructured content"; however, this usage is not common and so is not adopted in MoReq2.

which is listed in appendix 1. As a particular example, the specification mentions in several places that certain functions must be limited to administrative roles. This is not to say that administrative roles have to take policy decisions, merely that they must be the only users empowered by the organisation to execute them through the ERMS.

It is important to note that records management policy must be integrated with the organisation's business and technical requirements and that an administrative role can only implement, from a records management and system perspective, decisions taken by more senior management.

Finally, this specification is intentionally user-centric; it uses, as far as possible, the type of terminology commonly used by those working with electronic records. For example, the specification describes electronic files as "containing" records, for ease of understanding, even though electronic files strictly do not contain anything. See section 2.2 for further details.

1.8 Considerations for Individual Member States

As explained in the section on scope, section 1.3, this specification attempts to cover a wide range of requirements – for different countries, in different industries and with different types of records. The wide scope is intentional; but it leads to a significant limitation, namely that this single specification cannot represent a requirement which precisely maps onto existing requirements without modification. Different countries have their differing traditions, views and regulatory demands for managing records. In some cases these will have to be taken into account when applying this Model Requirements Specification, especially when using it to specify a new system. For this reason, MoReq2 allows for individual European Union countries to add a "national chapter", or "chapter zero," that sets out national requirements such as:

- Translations of key terminology and key concepts;
- National legislative and regulatory requirements;
- National standards and guidance on accessibility;
- Potentially, other national requirements;
- National resources for further information.

1.9 Customising this Specification

The requirements in this specification are intended to serve only as a model. They are not prescriptive for all possible ERMS implementations; some requirements will not apply in some environments. Different business sectors, different sizes of implementation, different organisation types and other factors will also introduce additional specific requirements.

As a result, this specification must be customised before use for procurement purposes. The customisation for procurement should:

- add or remove requirements as specifically required by the organisation;
- adjust requirements that can be made more specific. For example:
  - requirements that specify one of several possible outcomes can be changed to specify a single required outcome;
  - requirements for volumes and performance.

- include details specific to the organisation, such as the software environment;
- indicate clearly which requirements are:
  - unchanged from MoReq2,
  - new,
  - deleted,
  - adjusted.

This specification has been prepared so that it can be used in paper or electronic form. It has been prepared using Microsoft Word 2003, and is published in the following formats:

- Microsoft Word 97-2003 (Version 11);
- Microsoft Word 2007 (Version 12);
- Adobe PDF (Version 1.4).

Use in electronic form has a number of benefits; details are given in appendix 3.

1.10 Organisation of this Specification

The specification is organised into chapters which are divided into sections.

The next chapter (chapter 2) provides an overview of some of the key requirements, starting with terminology which is central to this specification.

Chapters 3 to 9 contain the core ERMS functional requirements in detail. Each chapter contains a logical grouping of functional requirements. However, given the nature of the subject matter there is inevitably some overlap between chapters.

Chapter 10 is divided into several sections, each of which represents requirements for an optional module of an ERMS. Some of these sections (e.g. the section on distributed systems) will be essential for some organisations, but unnecessary for others.

Chapter 11 contains non-functional requirements.

Chapter 12 identifies requirements for managing metadata; definitions of the metadata elements needed to support MoReq2 are in appendix 9.

Chapter 13 contains a formal reference model of ERMS as understood in this specification. This model can be used to understand key aspects of the specification, such as formal definitions of terms (e.g. class, sub-file, volume) and the relationships which exist between them (for instance “what can be stored in an electronic file?”).

The appendices contain details of reference documents, administrative and other information. Appendix 9 contains the MoReq2 metadata model. It is published separately from the rest of MoReq2 to ease cross referencing and because of its length.

In response to demand from many sources, testing materials have been developed to complement these requirements. The testing materials are published alongside the electronic copies of the requirements. The structure of MoReq2 is designed to support testing of compliance with the

requirements, e.g. each section of chapter 10 represents one optional test module. For more detail on MoReq2 testing see http://www.DLM-Network.org. .

The requirements are presented in the form of tables, with one requirement per table row. This is illustrated in figure 1.1.

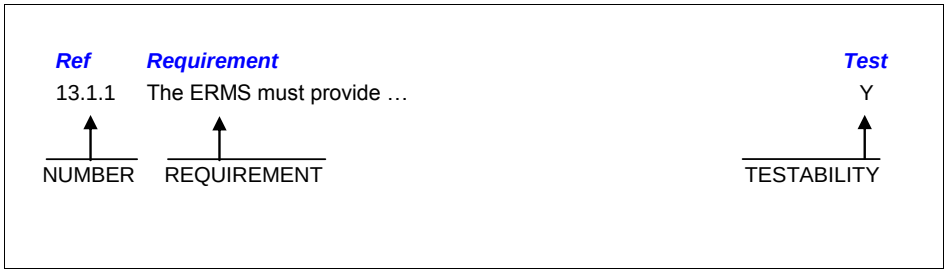

Figure 1.1

Each requirement bears a number, and each is expressed in natural language.

1.11 Compliance Testing

Testability

Each requirement is followed by an attribute labelled "Test". This indicates whether it will be possible to test compliance with the requirement. Possible values of this "testability" attribute are described below, with examples:

- **Y – The requirement can be tested formally.** An example is *"The ERMS must allow at least three hierarchical levels in the classification scheme"*. This can be tested by attempting to set up a hierarchy with three levels.

- **N – The requirement cannot be tested formally.** An example is *"The ERMS must support the organisation's business classification scheme"*. There is no way to test this in the general case.

- **P – The requirement can be tested but the coverage of the test is partial, and/or it is possible that lack of compliance can be discovered**. An example is *"the ERMS should not limit the number of levels in the hierarchy."* There is no way, formally, to test for the absence of a limit. However, the requirement is considered testable with partial coverage, for example by testing for a large number of levels; and during the testing it is possible that a limitation on the number of levels might be noticed, indicating that the ERMS does not comply with the requirement.

Systems beyond the ERMS

This specification is accompanied by the MoReq2 Testing Framework. The framework provides documentation that allows the compliance of an ERMS against MoReq2 to be tested.

Several MoReq2 requirements rely on hardware and software that is beyond the boundaries of the ERMS. For example, MoReq2 includes:

- requirements about e-mail integration that rely on features of e-mail software;

- scalability and integrity requirements that rely on features of database management software;

- scanning requirements that rely on scanning hardware.

Clearly it is not possible to test any ERMS with all possible hardware and software that might be used. Therefore, and as a matter of definition, such requirements will be tested with a combination of software and hardware specified by the ERMS supplier. The resulting compliance test certificate will specify the software and hardware that has been used for the test; compliance will extend to that environment only. Potential users of the ERMS wishing to know the compliance with any other software and/or hardware will need to assess it on a case by case basis.

1.12 Mandatory and Desirable Requirements

MoReq2 contains both mandatory and desirable requirements. This level of mandation is indicated as follows:

- the word “must” indicates that a requirement is mandatory;
- the word “should” indicates that a requirement is desirable.

In all cases, the level of mandation is dependent on its context. So, for example, a mandatory requirement in an optional module is mandatory only in the context of that optional module.

In some cases, a requirement is mandatory only if a desirable requirement is met. This is always clear from the context; for example the following:

- 3.1.17: The ERMS should support the export of all or part of a classification scheme.
- 3.1.18: Where the ERMS supports the export of all or part of a classification scheme (as in 3.1.17) this must include associated metadata […]

means that the functionality required by 3.1.18 is mandatory if, and only if, the desirable functionality required by 3.1.17 is provided.

1.13 Comments on this Specification

Information on how to submit comments and observations can be found on the DLM Forum website: http://www.DLM-Network.org.

2. OVERVIEW OF ERMS REQUIREMENTS

This chapter starts by defining some key terms (section 2.1). This is followed by a narrative description of some key concepts (section 2.2), and an entity-relationship diagram of the model on which this specification is based (section 2.3).

2.1 Key Terminology

MoReq2 requires certain terms to have precise meanings. Wherever possible, the meanings align with common usage, or usage generally agreed within the records management community. However, in some cases the usage is specific to MoReq2. All the terms are defined in the glossary (section 13.1). Key definitions – that is, the definitions that are crucial for an understanding of MoReq2 – from the glossary are reproduced here for ease of reference. The definitions reproduced here are identical to those in the full glossary.

In the definitions below, terms in *italics* are defined in the glossary, section 13.1.

capture (verb)

(1) The act of recording or saving a particular instantiation of a digital object (source: InterPARES 2 Project Terminology Database).

(2) Saving information in a computer system.

Note: in the context of MoReq2, capturing *records* is used to mean all of the processes involved in getting a record into an ERMS, namely registration, classification, addition of metadata, and freezing the contents of the source document. The term is used more generally to mean inputting to the ERMS and storing other information such as metadata values.

case file

A file relating to one or more transactions performed totally or partly in a structured or partly-structured way, as a result of a concrete process or activity.

Note: there is no universally-accepted definition of these terms, nor of the distinction between case files and the other kinds of files often managed by an ERMS. This definition is therefore developed for, and intended to facilitate the understanding of, MoReq2; its applicability in other situations is not guaranteed.

Note: the records in a case file may be structured or unstructured. The key distinguishing characteristic of case files is that they result from processes which are at least partly structured and repeatable. Examples include files about:

- applications for permits;
- enquiries about a routine service;
- investigation of an incident;
- regulatory monitoring.

Note: typically, other characteristics of case files are that they often:

- feature a predictable structure for their content;
- are numerous;
- are structured or partly structured;
- are used and managed within a known and predetermined process;
- need to be retained for specific periods, as a result of legislation or regulation;
- can be opened and closed by practitioners, end-users or data processing systems without the need for management approval.

class (noun)

(in MoReq2 only) The portion of a hierarchy represented by a line running from any point in the *classification scheme* hierarchy to all the files below it.

Note: this can correspond, in classical terminology, to a “primary class”, “group” or “series” (or sub-class, sub-group, sub-series etc.) at any level in the classification scheme.

Note: in MoReq2 class is also used to mean all the records allocated to a class.

classification

In records management, the systematic identification and arrangement of business activities and/or *records* into categories according to logically structured conventions, methods, and procedural rules represented in a classification system.

Source: ISO 15489 (see appendix 7).

classification scheme

(In MoReq2) A hierarchic arrangement of classes, files, sub-files, volumes and records.

component

A distinct bit stream that, alone or with other bit streams, makes up a *record* or *document*.

Note: this term is not in general use.

Note: the phrase “distinct bit stream” is used to describe what is usually called a “file” in information technology; the word “file” is avoided here to prevent confusion with the records management meaning of “file”. The key concept is that a “component” is an integral part of the content of a record, despite the fact that it can be handled and managed separately.

Note: examples of components include:

- An HTML document and JPEG images that make up a web page;
- A word processing document and a spreadsheet, where the record consists of the word processing document that contains an embedded link (a hyperlink) to the spreadsheet.

Note: components have to be distinct, that is separate from each other. If a word processed document contains an embedded spreadsheet (as opposed to an embedded link to a spreadsheet) then the spreadsheet is not considered to be a component; in this case, the word processed document complete with its embedded spreadsheet is a record made up of one component.

Note: an e-mail message with attachments may be one component, as several components, or as several records, depending on the format in which it is stored.

- If the message is stored in a format that includes the body and all its attachments, then there is only one component.
- If the attachments are stored separately from, and linked internally to, the body of the e-mail message, then each attachment and the body of the message is a component.
- If the attachments are stored separately from the body of the e-mail message but they are not linked internally, then each attachment and the body of the message is a separate record; good practice suggests that these records should be linked to each other manually.

document (noun)

Recorded information or object which can be treated as a unit.

Source: ISO 15489 (see appendix 7).

Note: a document may be on paper, microform, magnetic or any other electronic medium. It may include any combination of text, data, graphics, sound, moving pictures or any other forms of information. A single document may consist of one or several *components*.

Note: documents differ from *records* in several important respects. MoReq2 uses the term document to mean information that has not been captured as a record, i.e. classified, registered and locked against change. The word "recorded" in the definition does not imply the characteristics of a *record*. However, note that some documents become *records*.

electronic record

A *record* which is in *electronic* form.

Note: it can be in electronic form as a result of having been created by application software or as a result of digitisation, e.g. by scanning.

ERMS

Electronic Records Management System.

Note: ERMSs differ from *EDMSs* in several important respects. See section 10.3 for more details.

file (noun)

An organised unit of *records* grouped together because they relate to the same subject, activity or transaction.

Source: shortened and adapted from ISAD(G) (see appendix 7).

Note: this is the Records Management usage of the term *file*. It differs from the IT usage, for which MoReq2 uses the term *component.*

metadata

(In the context of records management) Data describing context, content and structure of records and their management through time.

Source: ISO 15489 (see appendix 7).

Note: some models are based on a different conceptual view of metadata. For example, they may treat audit trail information as being entirely metadata. These alternative views are valid and valuable in their contexts, but are not helpful in specifying the functionality of systems, and so are not considered here.

record (noun)

Information created, received, and maintained as evidence and information by an organisation or person, in pursuance of legal obligations or in the transaction of business.

Source: ISO 15489 (see appendix 7).

Note: local national definitions may also apply.

Note: a record may incorporate one or several *documents* (for instance when one document has attachments), and may be on any medium in any format. As a consequence, it may be made up of one or more *components.* In addition to the content of the document(s), a record should include contextual information and, if applicable, structural information (for instance information which describes the components of the record). A key feature of a record is that it cannot be changed.

Note: both electronic records and physical records can be managed by an ERMS.

sub-file

Intellectual subdivision of a file.

Note: sub-files are often used in case file management environments. Typically, each sub-file is named, and each sub-file is used to store a specified kind or kinds of records for one instance of a case, such as “invoices”, “assessments” or “correspondence”. They can, however, also be used, in a similar fashion, in non-case file environments.

volume

A subdivision of a *sub-file*.

Note: the subdivisions are created to improve manageability of the sub-file contents by creating units which are not too large to manage successfully. The subdivisions are mechanical (for instance, based on number of records or ranges of numbers or time spans) rather than intellectual.

2.2 Key Concepts

The key concepts required to understand this specification are:

- record and electronic record;

- authoritative record;
- electronic file, sub-file and volume;
- classification scheme;
- class;
- ERMS;
- capturing records;
- user roles.

Record and electronic record

As explained in section 2.4 of the DLM Forum Guidelines (appendix 1), records can be viewed as consisting of:

- content;
- structure;
- context;
- presentation.

The content is present in one or more physical and/or electronic documents that convey the message (the informational content) of the record. These are stored in such a way as to allow future users to understand them and their context. This view implies that a well-managed record consists of, in addition to the content of its document(s), information about its structure and metadata that provides information on its context, and its presentation to users. However in MoReq2, the term record is used to refer to the informational content – the document(s) from which the record is made, without the metadata. The presentation depends on a combination of the record's contents, structure and (in the case of electronic records) the software used to present it (see glossary).

In the world of physical records, the vast majority of records are on paper and are included in files, physically constituted of one or more volumes of records inserted within paper folders. Procedural controls should prevent users from changing the records, or their positions within the file.

Similar concepts apply to electronic records. A record is made from one or more electronic documents. These documents can be word processing documents, e-mail messages, spreadsheets, moving or still images, audio files or any other type of digital object. The documents become records when they are set aside, that is, "captured" into the ERMS. Upon capture, the records are "classified", that is they are assigned codes corresponding to the classification scheme class to which they belong, allowing the ERMS to manage them. The records usually are assigned to a file – though not always, see below.

For preservation purposes, it is necessary to appreciate that electronic records are often made up of several components (the word "component" is used in MoReq2 to avoid the IT word "file", so as to reduce the likelihood of confusion with records management "files"). Each component is an object managed by a computer operating system, and they may be in different formats; but they are all needed together to make up a record. Not all records have more than one component; for

example, most word processing documents are made of only one component. An example of a record with several components is a web page with text, graphics and style sheets; it is not unusual for a web page to contain one HTML component, dozens of JPEG image components, and a handful of CSS (cascading style sheet) components.

An essential quality of records is that their informational content is fixed. One consequence of this is that no action carried out on electronic records can be allowed to interfere with the relationships between its components; in other words, all actions carried out on any record must preserve the correct relationships between all its components. So, for example, whenever any record is moved or copied, it must be moved or copied in a way that keeps all its components and all their relationships.

Authoritative Records

ISO 15489 describes an "authoritative record" as being a record that has the characteristics of:

- authenticity;
- reliability;
- integrity;
- usability.

As explained in ISO 15489, the aim of all records management systems should be to ensure that records stored within them are authoritative. Summarising, an authoritative record:

- can be proven to be what it purports to be;
- can be proven to have been created or sent by the person purported to have created or sent it;
- can be proven to have been created or sent at the time purported;
- can be depended on because its contents can be trusted as a full and accurate representation of the transactions, activities or facts to which it attests;
- is complete and unaltered;
- can be located, retrieved, presented and interpreted.

The requirements in MoReq2 are designed to ensure that records stored in a MoReq2-compliant ERMS are authoritative. However, compliance with these requirements alone is not sufficient; the existence of, and compliance with, corporate policies is also required.

Electronic File, Sub-file and Volume

Paper records generally are accumulated in physical files, contained in paper folders. The paper files are aggregated into a structure, or classification scheme. In an ERMS electronic records can be managed as if they are accumulated in electronic files and stored in electronic folders. Strictly, electronic files and folders need not have a real existence; they are virtual, in the sense that they do not really "contain" anything; in fact they consist of the metadata elements of the records assigned to them. Further, in many cases, there need be no real distinction in the electronic system between file and folder. However, these details are not generally visible to ERMS users; ERMS application software allows users to view and manage folders as if they physically contained

the documents logically assigned to the files. This user-centred view is carried forward into this specification. The rest of this specification therefore describes electronic files as "containing" records, for ease of understanding. Note however, that while this specification provides functional requirements for the management of electronic files, it does not prescribe the manner in which the concept of electronic files is implemented.

In some environments it is useful to divide files into sub-files. The division into sub-files is an "intellectual" one; that is it (generally) requires human input to decide into which sub-file a record should be stored. Sub-files are most often used in case processing environment. An example would be a file for the sale of land, with sub-files for each of the business activities involved in the sale (such as advertising; contracts; dealing with lawyers, etc.).

A sub-file is therefore a division of a file by type of content. As a result, a sub-file can be used to permit the application of a different retention and disposition schedule to a set of records within the file.

Regardless of whether sub-files are used or not, files are sometimes divided "mechanically" into file volumes, according to predetermined conventions. The term "mechanically" implies simple adherence to such conventions, which are not based on the intellectual content of the files, but on size, number of records contained in them, or time spans. This practice originated with paper files, in order to restrict them to a manageable size and weight. It can be continued with electronic files, to limit them to a manageable length for appraisal, transfer, or other management purpose. It is especially appropriate for the management of files which are open for long periods and/or which grow to contain a large number of records.

While the distinction between files and file volumes is clear, the implications are less clear. This is because the implications of choosing to divide files into volumes vary according to implementation needs. The variation arises as:

- some files are closed within a limited time, and so the unit used for management purposes is the file (even though a file may consist of several volumes). Examples are a file of a specific small procurement, or a file of one project;

- some files have an unlimited life span (or nearly unlimited life span), and so the unit used for management purposes is the volume. Examples are a file of records about a geographic region, or a file dealing with a subject which is not sensitive to time, such as some policies, or an invoice file where a new volume is started every year.

In relatively rare cases, records may be stored outside of files – by being assigned to a class. This is explained in 3.2.17.

Classification scheme

Records management aggregates files in a structured manner, and good practice dictates that this structure should reflect business functions. The representation of this aggregation is referred to as a "classification scheme". The classification scheme is commonly a hierarchy. The remainder of MoReq2 focuses on the hierarchical view; other approaches are outside the scope of MoReq2, and a hierarchical arrangement is a prerequisite for MoReq2 compliance.

Just as files appear to exist even though they are really no more than aggregations of records, so higher levels of the classification scheme hierarchy seem to exist, though they are no more than aggregations of files and/or lower levels. As with files, this specification states requirements for the hierarchy without mandating the manner in which it is implemented.

Files can appear at any level of the hierarchy. This is illustrated in figure 2.1, which represents a fictitious classification scheme, showing its classes and the files allocated to the lowest level classes. This fictitious scheme is much simpler than would be a real classification scheme.

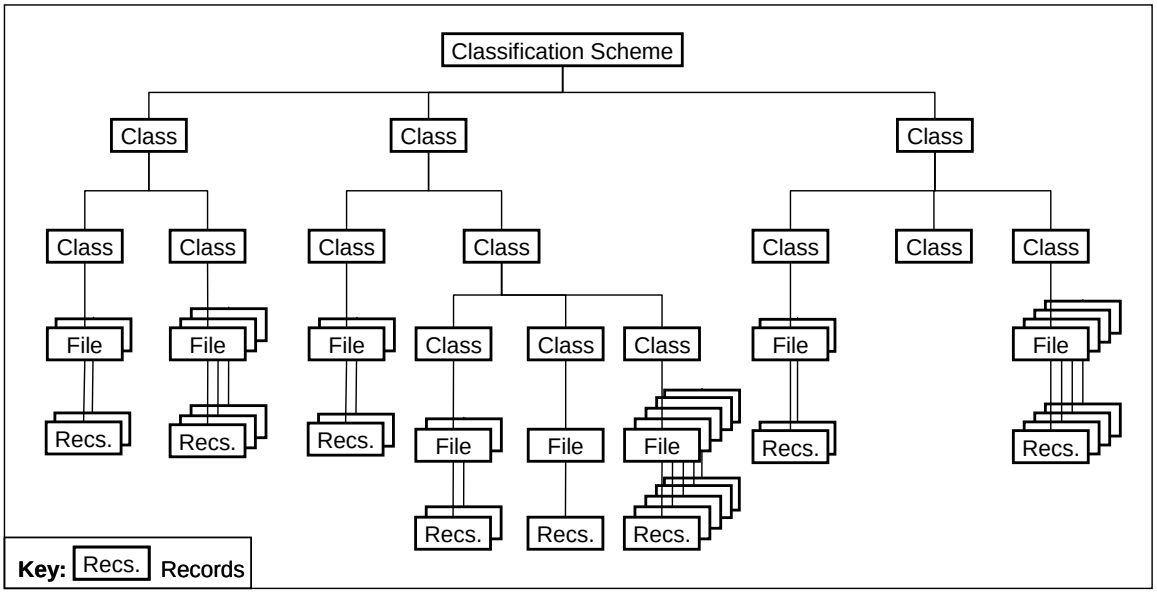

Figure 2.1

Note that this figure is intended only to show selected possible relationships between levels, files and records. It does not show all possible levels or all possible arrangements.

Class

MoReq2 uses the term “class” to describe the portion of a hierarchy represented by a line running from any point of the hierarchy to all the files below it. The term class therefore corresponds to a “group” or “series” (or sub-group, sub-series etc.) in some texts.

Visually, a class of a hierarchy corresponds to a branch of a tree. A class may thus contain other classes, just as a series contains sub-series and sub-sub-series. Continuing the above example, the shaded boxes and thick lines in figure 2.2 are one example of a class.

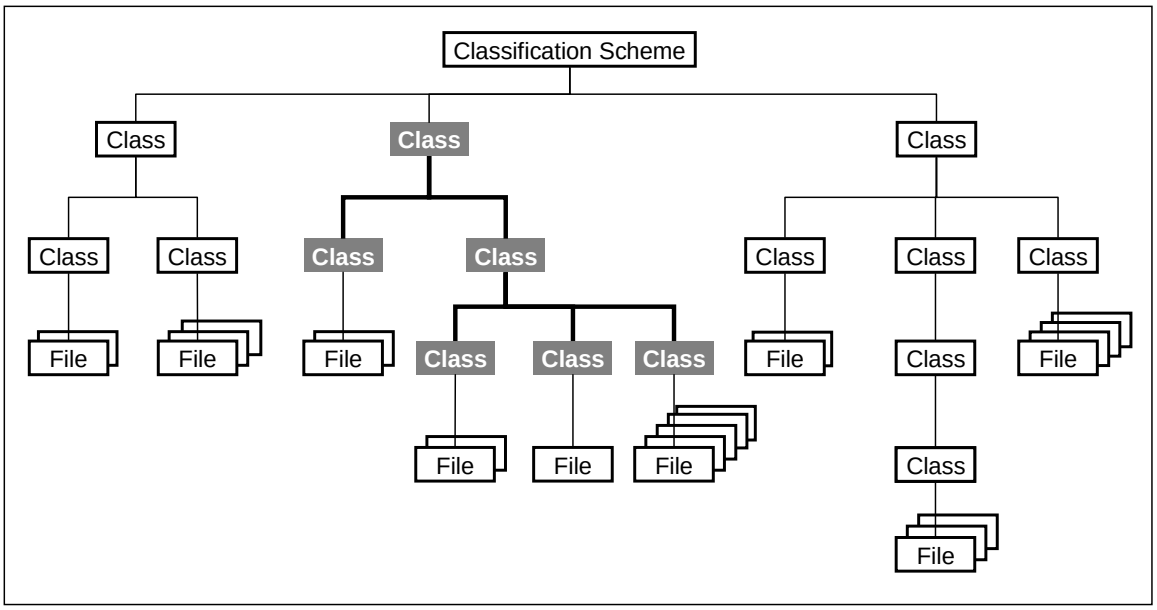

Figure 2.2

MoReq2 also uses the term "class" to mean all the files, records etc. assigned to a class – much as the word "bottle" can be used to describe both a container and that container full of a liquid. This double usage is intentional, and the appropriate interpretation of the term is always clear from the context.

MoReq2 uses the terms "child" and "parent" to describe the relationships between entities. A "child" of one entity is an entity that is below it in the hierarchy (in other words, is a descendant entity). A "parent" of one entity is an entity that is above it in the hierarchy. So for example, the children of classes can be other classes, files, or (in rare cases) records.

MoReq2 allows for records to be assigned to, or stored directly in, a class without being in a file. This is intended for relatively rare circumstances, as described in the body of MoReq2.

Electronic Records Management System (ERMS)

An ERMS is primarily an application for managing electronic records, though it may also be used to manage physical records.

An ERMS is often closely integrated with an Electronic Document Management System (EDMS) or a business application. Technically, an ERMS manages records, while an EDMS manages documents (which are not records). However, especially when used to support day-to-day working, it can be difficult to separate their functionality. This is explored further in section 10.3 which deals with Document Management.

Capturing Records

Documents made or received in the course of business become records when they are set aside, that is, "captured" into the ERMS. During capture, the records are "classified", that is they are assigned codes corresponding to the class to which they belong, allowing the ERMS to manage them; and they are also assigned a unique identifier.

In many cases, documents that are set aside, or captured, become records by being bound to a business process, as often happens in a workflow. For example, when an invoice is raised it

should automatically cause a record to be captured. In other cases there may be a policy that every document relating to a business matter must become a record, even if it does not formally participate in a business process. In yet other circumstances however, the process of capture will be initiated selectively by a user. Determination of which documents should be captured into a records system should be based on an analysis of the regulatory environment, business and accountability requirements and the risk of not capturing the records. An example is a memorandum in an organisation which deals with policy issues; the organisation may define that only memoranda deemed to be significant will become records (i.e. insignificant memoranda, such as those relating to meeting arrangements, will generally not form records). In some situations, the drafts will be deemed to be significant and will become records, whereas in other situations drafts will not become records. MoReq2 is intended to cater for any of these scenarios. In other words, MoReq2 describes an office system for general use, not simply a records management system for particular kinds of application or for the exclusive use of archivists or administrators.

User and Administrative Roles

MoReq2 uses the concept of "user" to mean any person with valid permissions to work using the ERMS. Therefore anyone who is allowed to log on to the ERMS is a user, including administrators. However, the distinction between administrators and other users can be complex and is sometimes unclear. MoReq2 therefore uses the concepts of "roles" in defining many requirements.

Different organisations will implement an ERMS differently. For example, a small organisation may implement an ERMS with a single administrator, while a large organisation may need several different administrative positions, each with different access permissions. For this reason, it is not helpful to identify specific access profiles in this generic specification; instead, MoReq2 uses the concept of "roles".

MoReq2 identifies two kinds of roles: "user roles" and "administrative roles". In practice, most organisations will have more than one person in these roles; and many organisations will define further roles. Example roles with possible access permissions are outlined in the matrix at section 13.4.

In brief, however, a "role" in MoReq2 is something like a user profile – it is not a job or a position, but a set of responsibilities and functional permissions shared by several users. MoReq2 recognises examples of two administrative and two user roles.

Administrative roles take actions related to the management of records themselves; their interest is in managing records as entities rather than their content or business context. They also manage the ERMS hardware, software and storage, ensure backups are taken and manage the performance of the ERMS.

Unlike administrative roles, user roles have access to facilities which an office worker or researcher needs when using records. This includes adding documents, searching for and retrieving records; their interest is primarily in the contents of records rather than their management – in other words, they are interested in the business processes evidenced by the records.

2.3 Entity-Relationship Model

This section contains an entity-relationship model at figure 2.5 which can be used as an aid to understanding the specification. Section 13.3 contains a narrative explanation.

An important aspect of this model is that it need not represent actual structures stored in the ERMS. It represents a theoretical view of the entities associated with records. An ERMS uses

these relationships to produce behaviour equivalent to the structures in the model. See section 2.2 for further explanation of this point.

The relationships between the following key entities are depicted in the following entity-relationship model:

- Class;
- File;
- Sub-file;
- Volume;
- Record;
- Component.

Other entities are also included.

In the diagram, entities – files, records and so on – are represented by rectangles. The lines connecting them represent the relationships between the entities. Each relationship is described by text in the middle of the line; this text should be read in the direction of the arrow. Each end of the relationship has a number which represents the number of occurrences (strictly, the cardinality); the numbers are explained in the key. So, for example, figure2.3 means "one record is made up of one or more components" (note the direction of the relationship arrow).

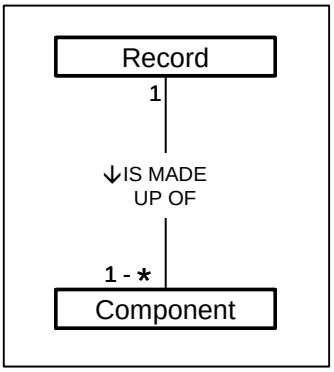

Figure 2.3

A curved line crossing two or more relationships indicates that the relationships are mutually exclusive, for any given instance. So, for example, the curved line in figure 2.4 means "each record is stored in either a volume or in a sub-file but not in both".

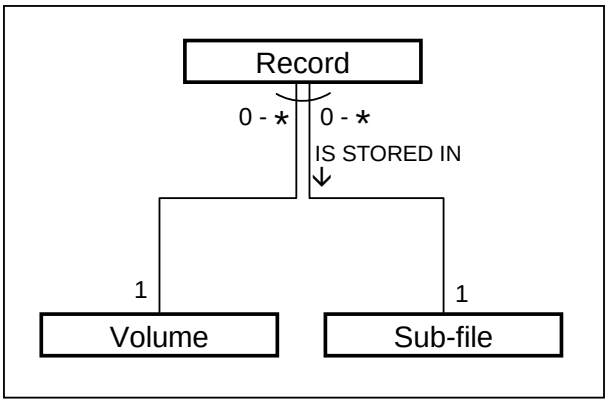

Figure 2.4

Note that the entity class is related to itself by the relationship “is made up of”. This relationship describes, in formal terms, the relationship between classes in a hierarchical classification scheme, where a class may be made up of one or more other classes. If this relationship (sometimes called a recursive relationship) is removed, the model applies equally to non-hierarchical relationships.

In the remainder of MoReq2, terms printed in **blue bold text** indicate the first usage of a term defined in the glossary. In the electronic version this is a hyperlink to the definition, so pressing CTRL + click on the term navigates to the glossary definition, and pressing CTRL + click on the glossary definition navigates back to the term.

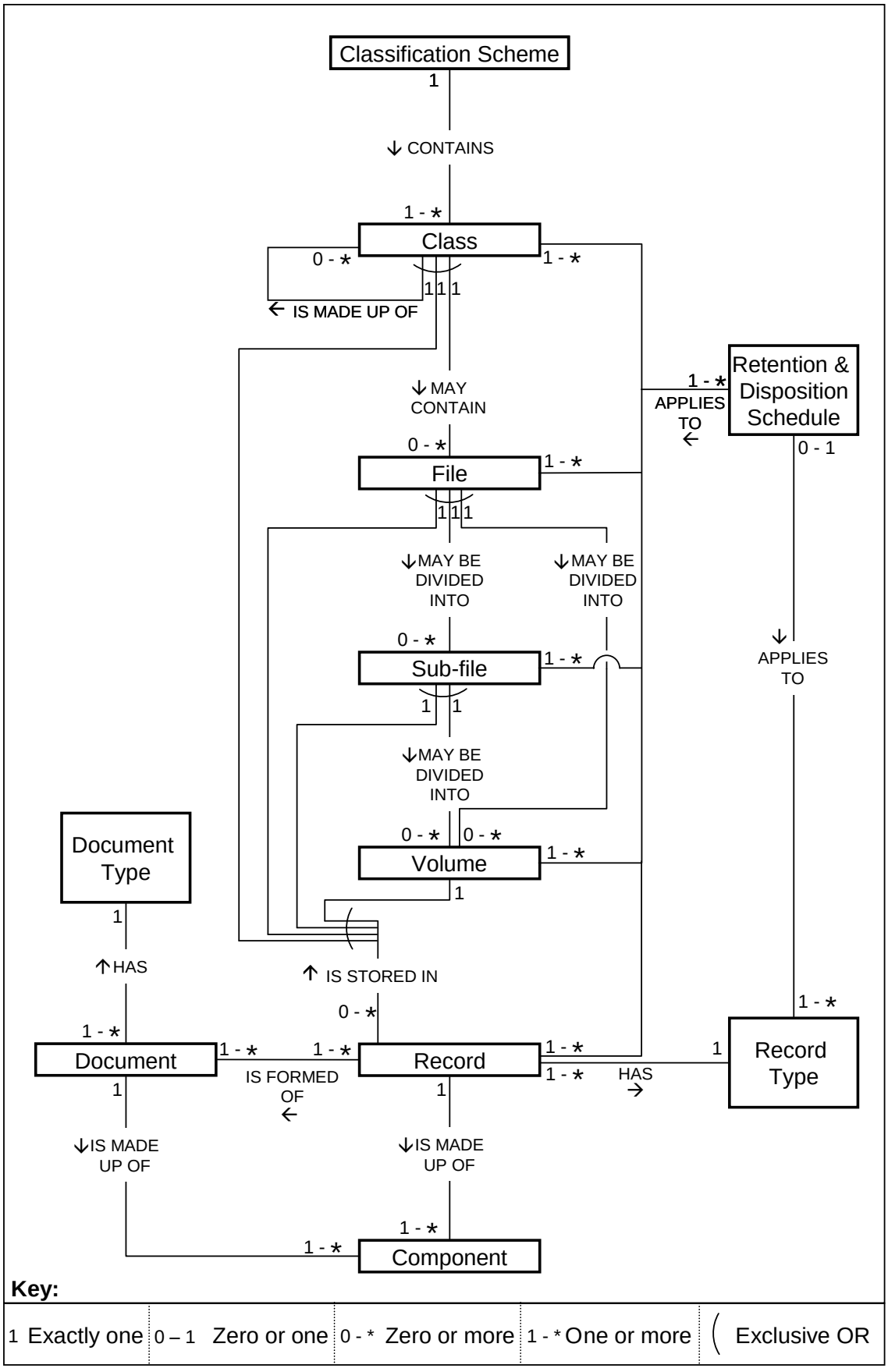
Classification Scheme
1
CONTAINS
1 - *
Class
0 - *
1 - *
1 1 1
IS MADE UP OF
MAY CONTAIN
0 - *
File
1 - *
1 1 1
MAY BE DIVIDED INTO
MAY BE DIVIDED INTO
0 - *
Sub-file
1 - *
1 1
MAY BE DIVIDED INTO
0 - *
0 - *
Volume
1 - *
1
Retention & Disposition Schedule
1 - *
APPLIES TO
0 - 1
APPLIES TO
Document Type
1
HAS
1 - *
Document
1
IS STORED IN
0 - *
1 - *
1 - *
IS FORMED OF
Record
1 - *
1 - *
HAS
1
Record Type
1 - *
IS MADE UP OF
IS MADE UP OF
1
1
1 - *
1 - *
Component
Key:
1 Exactly one
0 – 1 Zero or one
0 - * Zero or more
1 - * One or more
Exclusive OR

Figure 2.5

3. CLASSIFICATION SCHEME AND FILE ORGANISATION

This chapter lists requirements for management of the **classification scheme** and of the organisation of **files**. It first lists requirements for setting up the classification scheme in section 3.1. It then lists requirements relating to **classes** and files (section 3.2) and **volumes** and **sub-files** (section 3.3). Section 3.4 lists requirements associated with maintenance of the classification scheme.

A classification scheme is the foundation of any **ERMS**. It allows an **electronic record** to be stored together with other **records** that provide its context, by defining the way in which the electronic records will be organised into electronic files, and the relationships between the files.

A significant difference between MoReq2 and its predecessor is that MoReq2 allows the declaring of a record directly into a class, as well as into a file. The original MoReq did not allow declaration directly into a class; it allowed only declaration into a file.

MoReq2 thus allows a record to be **captured** into any of the following:

- Class;
- File;
- Sub-File;
- Volume.

Records will most commonly be captured into volumes; for the rationale that requires capture in files and sub-files see 3.3.1, 3.3.2 and 3.3.3.

Capture of records into classes is illustrated in figure 3.1, which adds such records (shaded in grey) to figure 2.1.

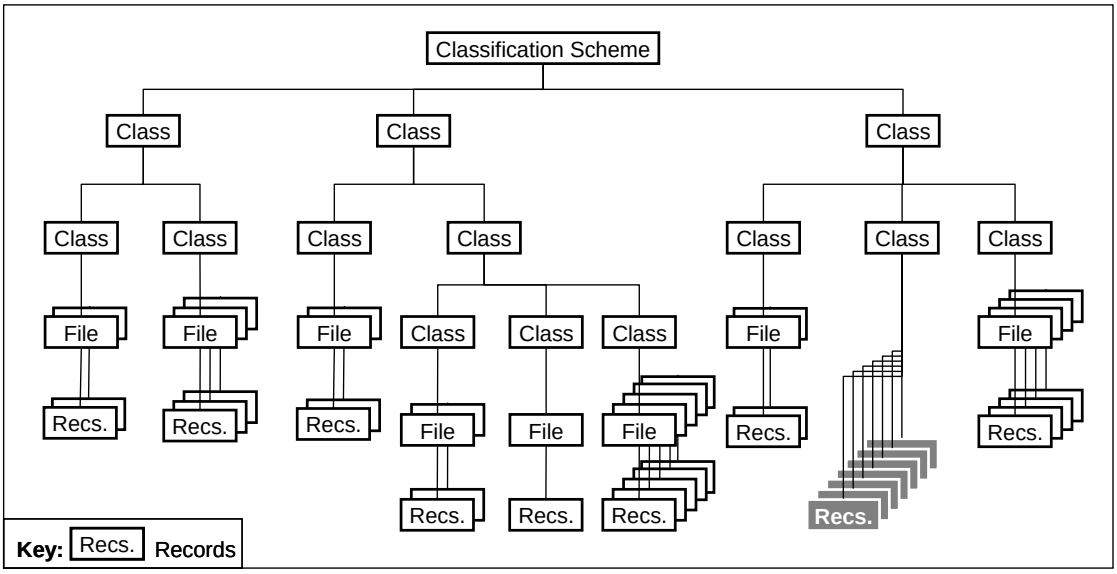

Figure 3.1

This change has been introduced to reflect the requirements of high-volume case management systems. It is, however, not meant to remove the necessity for a hierarchical classification scheme, or for the existence of files. Inappropriate use of this feature will introduce the risk of later difficulties in managing records, and users of MoReq2 are advised to use this functionality only after careful analysis. Most users of MoReq2 are unlikely to require this functionality, and so MoReq2 includes the requirement that this functionality can be disabled.

MoReq2 compliance requires support for hierarchical **classification**. This is because:

- hierarchical schemes are able to ensure an effective, stable and clear organisation of records;
- hierarchical schemes are the most widely used in Europe.

It also maintains compatibility with the previous version of MoReq. Many requirements use the concept of class. In many cases, it may be possible to apply the requirement to non-hierarchical classification schemes; but this may not always possible.

It is essential that the classification scheme (technically, a records classification scheme) is closely aligned with the business needs of the organisation. Good practice suggests that the organisation first identifies a business classification scheme before designing a records classification scheme.

3.1 Configuring the Classification Scheme

Ref	*Requirement*	*Test*
3.1.1	The ERMS must support and be compatible with the organisation's business classification scheme. *This requirement is not testable in the general case; it is included as a reminder to users of MoReq2 of the need to align the classification scheme used by an ERMS with the business needs of the organisation. These needs should be reflected by the arrangement of records external to the ERMS.*	N
3.1.2	The ERMS must maintain internal integrity (relational integrity or otherwise) at all times, regardless of: ♦ maintenance activities; ♦ other **user** actions; ♦ failure of system components. *In other words, it must be impossible for a situation to arise where any user action or any software failure results in an inconsistency within the ERMS or its database.*	P
3.1.3	The ERMS should allow **administrative roles** to label each classification scheme with a Title, and Description, and must automatically label each classification scheme with an Identifier. *This **metadata** will support functions such as **export** of the classification scheme and of records.*	Y

Ref	Requirement	Test
3.1.4	The ERMS must be able to support a classification scheme which can represent files and records as being organised in a hierarchy of classes. *The use of a hierarchical classification scheme is mandatory for compliance with MoReq2. This is in order to enable the inheritance of retention and disposition schedules and other metadata and also to facilitate navigability.* *Support for at least three levels is the minimum requirement; more levels will be needed in many environments.*	Y
3.1.5	The ERMS must allow management of the classification scheme by an administrative role only, subject to requirement 3.1.6. *In this requirement, "management" refers to the operations described in section 3.1 and section 3.4.*	Y
3.1.6	The ERMS should allow management of individual classes by specified **user roles** and/or by a specified **group** of users. *In this requirement, "management" has the same meaning as in requirement 3.1.5. This is intended for two settings:* ♦ *large classification schemes which are too large to be maintained centrally (and which therefore have central management for the higher levels and distributed management for the lower levels);* ♦ *classification schemes that include classes for the management of **case files**, which need to be managed in the business unit dealing with the cases on allocation of **authorised user** privileges.*	Y
3.1.7	The ERMS should not limit the number of levels in the classification scheme hierarchy. *In most settings, it is unlikely that the number of levels needed could be more than ten.*	P
3.1.8	The ERMS must support the creation of a classification scheme at **configuration time** in readiness for the capture and/or importation of electronic records. *This requirement is intended to allow a classification scheme to be created while the ERMS is being configured, and before it is used for the management of records.*	Y
3.1.9	The ERMS must allow the titling mechanism(s) to be defined at configuration time by an administrative role**.**	Y

Ref	Requirement	Test
3.1.10	The ERMS should allow the input of textual scope notes (also known as descriptions) to all classes, files, sub-files and volumes. *Scope notes are narrative intended to clarify the intended contents and/or exclusions of classes, files sub-files and volumes for the benefit of users.*	Y
3.1.11	If a formal MoReq2 XML schema has been published, the ERMS must be able to **import** and export records etc. in a form compliant with that schema.	Y
3.1.12	The ERMS must support the importing of all or parts of a classification scheme, at configuration time or at any other time. *This requirement is intended to allow a classification scheme to be created while the ERMS is being configured, and before it is used for the management of records. Where any part(s) is (are) imported, this may be to add to an existing scheme, or to create a new classification scheme if none exists.*	Y
3.1.13	When the ERMS imports all or part of a classification scheme it must allow the import of the associated metadata, **retention and disposition schedules** and **audit trails** if these exist. *In ideal cases, the classification scheme that is being imported will have class metadata and retention and disposition schedules. In other cases, these may be absent or incomplete.*	Y
3.1.14	Where the ERMS imports the metadata of a classification scheme, it must reject any class that does not have a title, and create an exception report for an administrative role listing the classes that were rejected. *In an ERMS that is not MoReq2 compliant it may be possible for a class to have no title (a null value); but such a class would be impossible to use within a MoReq2-compliant ERMS.*	Y
3.1.15	Where the ERMS imports the metadata of a classification scheme, the ERMS must assign to each imported class a hierarchical code in one of the following ways, according to an option set by an administrative role: ♦ following the same rules as would be used for the manual creation of the classification scheme; ♦ keeping the original codes in their entirety (only possible if the structures are compatible); ♦ appending the original codes to the codes in the receiving scheme. *If a hierarchy that is being imported already includes hierarchical class codes (for example 4/6/4) it may not be possible to use these as codes in the ERMS, as consistency and uniqueness cannot be guaranteed.*	P

Ref	Requirement	Test
	There are many possible scenarios for such an import, with different kinds of incompatibility between hierarchical numbering schemes. MoReq2 does not prescribe the outcome of an attempt to select an option that is logically impossible because the schemes are incompatible.	
	If the existing codes cannot be used, they can be treated as appropriate to the situation, e.g. copied to a metadata element called "old class code".	
3.1.16	Where the ERMS imports the metadata and retention and disposition schedules of a classification scheme, it must validate them using the same rules as would be used for the manual creation of the classification scheme (see chapter 12). Where this validation process finds errors (for example the absence of mandatory metadata, or format errors) it must bring these to the attention of the administrative role performing the importation, identifying the metadata involved. *In ideal cases, the classification scheme that is being imported will have metadata (e.g. metadata for its classes) that complies fully with the MoReq2 metadata model. In other cases, the metadata may be non-compliant. In these cases, several outcomes are possible; MoReq2 does not mandate any one outcome. Possible outcomes include:* ♦ *The entire importation is cancelled and the administrative role is informed of the reason for the cancellation;* ♦ *Importation of the class that has non-compliant metadata is cancelled and the administrative role is informed of the reason for the cancellation;* ♦ *The administrative role is required to choose between correcting the error and cancelling importation of the affected class;* ♦ *Importation continues even though part of the metadata is non-compliant, with non-compliant data being replaced by default values specified for the affected elements and an error report produced.* *Informing the administrative role does not require that the importation process be a foreground, or real-time process; it will be acceptable for the process to be a background, or batch, process.*	Y
3.1.17	The ERMS should support the export of all or part of a classification scheme.	Y
3.1.18	Where the ERMS supports the export of all or part of a classification scheme this must include associated metadata, an administrative role being able to select which metadata is exported.	Y
3.1.19	Where the ERMS supports the export of all or part of a classification scheme this must include all associated retention and disposition schedules at the option of an administrative role.	Y

Ref	Requirement	Test
3.1.20	Where the ERMS supports the export of all or part of a classification scheme, this must include all or selected audit trail data, the selection to be made by an administrative role.	Y
3.1.21	Where the ERMS supports export (for any of the above requirements) it must use a fully-documented method to relate the entities to each other. *The documentation of the method must define how the records, files, classes etc., and their relationships to each other, are expressed. See also 3.1.22.*	Y
3.1.22	Where the ERMS supports export (for any of the above requirements) it should export the information in XML or equivalent open standardised **format**.	Y
3.1.23	Where the ERMS supports the copying of all or part of a classification scheme this must include all associated metadata.	Y
3.1.24	Where the ERMS supports the copying of all or part of a classification scheme this must include all associated retention and disposition schedules.	Y
3.1.25	The ERMS must allow administrative roles to add new classes at any point within any class, so long as files or records are not stored at that point. *MoReq2 does not allow files and classes to exist at the same level within a class (in other words, files and classes cannot be mixed at a single node in the classification scheme hierarchy). This is for reasons of good records management practice.*	Y
3.1.26	The ERMS should support the definition and simultaneous use of multiple classification schemes. *Most organisations will mandate that a single classification scheme be used for the primary classification of all the files in the ERMS. This requirement allows some of the files in the ERMS to belong to one classification scheme while other files belong to another. This may be required, for example, following the merger of two organisations, or when different collections of records in a single organisation require different management regimes.*	Y

3.2 Classes and Files

This section lists requirements which apply to classes and files.

Classes and files are different kinds of construct. Classes provide a framework for classification, while files aggregate records; classes are building blocks of classification schemes, while files are not. Despite these major differences, it is helpful to list some requirements together, as they are common to both.

Ref	Requirement	Test
3.2.1	The ERMS must support the capture, maintenance and **presentation** of metadata for files and classes in the classification scheme, compliant with the MoReq2 metadata model.	Y
3.2.2	The ERMS must restrict the ability to add to file and class metadata as set out in the MoReq2 metadata model.	N
3.2.3	The ERMS must provide a mechanism for allocating automatically a hierarchical **classification code** (where such a code does not already exist – see 3.1.15) to each class, file, sub-file and volume in the classification scheme. *See also 7.1.1.*	Y
3.2.4	The ERMS must allow user roles to allocate a title for each **electronic** class, file, sub-file and volume. *This requirement applies to **non-case file** environments. Where case file management is needed, an alternative naming approach is needed. This is specified in section 10.5.*	Y
3.2.5	It must be possible to use both the classification code and textual file title separately or together.	Y
3.2.6	The ERMS must allow an administrative role to configure the classification code at configuration time or later.	Y
3.2.7	The ERMS should allow configuration of the classification code to include: ♦ the format of the identifier associated with each level of the hierarchy, e.g. numeric, alphabetic; ♦ the first value of this identifier at each class, e.g. 1, 1000; ♦ the interval to be used between successive classes, e.g. 1, 10; ♦ the presence or absence of leading zeroes; ♦ any global prefix, e.g. “corporate/”; ♦ any global extension, e.g. country suffix; ♦ the separator between each identifier, e.g. “/”, “-”.	Y
3.2.8	The ERMS must store the date of opening and the date of closing of a class or file within the class’ or file’s metadata. *The date of opening and closing of a class or file provide important context for the records classified within it. See also 3.3.9.*	Y

Ref	Requirement	Test
	*When a class or file is **open**, it is possible to capture records into it. When a class or file is **closed**, it is not possible to capture records into it.*	
3.2.9	The ERMS must store the date of creation of a new class, file, sub-file or volume in the metadata of the class or file. *In the case of **physical files**, it is possible for the date of opening to be earlier than the date of creation stored in the ERMS. This can arise if a physical file is created and opened, in physical form only, before it is created in the ERMS.* *In the case of electronic files, it is possible for the date of opening to be earlier than the date of creation stored in the ERMS. This can arise when an electronic file is imported into the ERMS from another system.*	Y
3.2.10	Whenever a new class or file is opened, the ERMS must automatically include in its metadata those attributes which are inherited due to its position in the classification scheme. *For example, if a file titled "Public meetings" is in a hierarchical path titled:* *Regional plan development : Public consultation : Public meetings* *and an administrative role adds a new file titled "Written consultations" at the same level as the "Public meetings" file then the new file must automatically inherit the prefix Regional plan development : Public consultation.* *Note that inherited metadata does not have to be stored explicitly; it can be inherited implicitly. See appendix 9.3 for details.*	Y
3.2.11	The ERMS must allow an administrative role to modify inherited metadata values, to the extent permitted by the MoReq2 metadata model. *Inherited values often provide a default, or starting position. This can be changed, so long as the change is compatible with the metadata model.*	Y
3.2.12	Any addition to the inherited metadata of a class should be inherited by default by all its child classes and files.	Y
3.2.13	The ERMS should support the allocation of controlled vocabulary terms compliant to ISO 2788 as descriptive class or file metadata subject terms, in addition to the other requirements in this section.	Y
3.2.14	The ERMS should support the allocation of controlled vocabulary terms compliant to ISO 5964 as descriptive class or file metadata subject terms, in addition to the other requirements in this section. *Requirements 3.2.13 and 3.2.14 are identical save that the former specifies a monolingual thesaurus and the latter a multilingual thesaurus.*	Y

Ref	Requirement	Test
3.2.15	The ERMS must not impose any practical limit on the number of classes or files which can be defined.	P
3.2.16	The ERMS should be able to export a list, or **repertory**, of all files or of files classified against a specific class (and its child classes) in XML format and/or in a human-readable format.	P
3.2.17	The ERMS must allow an administrative role to configure a class so that it can, or so that it cannot, store records directly. *In other words the system must be able to be configured so that records do not have to be held in files, sub-files or volumes.*	Y

3.3 Volumes and Sub-Files

In a system that keeps paper records, subdivision of large files is essential for reasons of ergonomics and the physical survival of folders, binders, jackets etc. Typically, **paper files** are limited to 2cm in thickness, by the establishment of volumes. When the file (in reality the first volume of the file, despite being referred to as a "file") reaches the size limit – 2cm thick in this example – it is considered to be a closed volume and a new volume is opened. This is not true of electronic files – an electronic file can usually grow to almost any size without such difficulties.

However, in practice, there can be benefits in splitting large electronic files into volumes. These benefits are, for example:

- when users need to work remotely (that is, over low-bandwidth connections, or after downloading records to a portable PC, or onto a storage device with limited capacity);
- when files are never closed, because they are (for example) geographically linked.

Similarly, paper files are often divided into sub-files – especially in case management environments. The sub-files are used to organise the file contents, often according to **document type**.

Correspondingly, there are sometimes benefits in dividing electronic files into sub-files, for example:

- improving the ease of navigation through a file;
- providing a means to manage records that have retention requirements that differ from others in the file, such as those covered by privacy legislation.

This section includes requirements relating to the use of volumes and sub-files, both of which are typically used to subdivide files which might otherwise be unmanageably large. However, Moreq2 does not mandate that these subdivisions be implemented; it merely requires that MoReq2 compliant software must be able to provide them when needed.

Sub-files were not recognised in the previous version of MoReq.

In summary:

- Each file may contain one or many sub-files;
- Each sub-file may contain one or many volumes;
- Volumes of different sub-files are created independently;
- All the sub-files of an open file can be open or closed by users as required;
- Only one volume can be open in each sub-file.

For more detail about sub-files and volumes, see section 2.2.

Ref	Requirement	Test
3.3.1	It must be possible for an administrative role to configure the ERMS at configuration time or later to remove the ability to create sub-files and/or volumes within files across the classification scheme.	Y
3.3.2	It must be possible for an administrative role to configure the ERMS at configuration time or later to allow only sub-files to be created within files within specified classes of the classification scheme.	Y
3.3.3	It must be possible for an administrative role to configure the ERMS at configuration time or later to allow only volumes to be created within files within specified classes of the classification scheme. *The intention of the three requirements above is to allow organisations to allow or prevent the use of sub-files and/or volumes in different parts of the classification scheme. The use of both brings the maximum flexibility, but this flexibility brings complexity and possible confusion for users.* *Where a part of a classification scheme is configured to allow sub-files, then all files in it must contain at least one sub-file. Where a part of a classification scheme is configured to allow volumes, then all files (or sub-files if allowed) must contain at least one volume.* *Therefore the system should remain transparent to users, for example:* ♦ *when a sub-file contains only one volume, it is acceptable for the sub-file and volume to be indistinguishable to end users;* ♦ *when a file contains only one sub-file which itself contains only one volume, it is acceptable for all three to be indistinguishable to end users.* *The intention of this is to emphasise that the ERMS need not impose on users the structure of "file, sub-file, volume". The ERMS must allow the use of sub-files and volumes, while allowing users to think in terms of files only if this suits them.*	Y

Ref	Requirement	Test
	The essence of this is that the user only sees what is essential from a business process point of view and is not encumbered by potentially confusing choices.	
3.3.4	The ERMS must support the concept of open and closed electronic volumes, as follows: ♦ only the most recently created volume within a sub-file can be open; ♦ all other volumes within that sub-file must be closed.	Y
3.3.5	The ERMS must prevent the user from adding electronic records to a closed volume.	Y
3.3.6	The ERMS must allow administrative roles to add an electronic volume to any electronic sub-file which is not closed. *The process of adding a new volume consists of closing the volume that is currently open and creating a new open volume.*	Y
3.3.7	The ERMS must allow administrative roles to add sub-files to any electronic file which is not closed.	Y
3.3.8	The ERMS must allow users to **close** a sub-file at any time.	Y
3.3.9	The ERMS must store the date of opening of a new volume or sub-file in its metadata.	Y
3.3.10	Whenever a new volume or sub-file is opened, the ERMS must automatically store in its metadata those values of its parent file's metadata which are common (as defined in the MoReq2 metadata model). *Records in a volume can be accessed regardless of whether the volume is open or closed.*	Y
3.3.11	Whenever a new volume is opened, the ERMS must automatically assign to it an identifier that is unique within its parent sub-file. *The identifier could be a simple sequence number, starting at 1 for each sub-file.*	P
3.3.12	The ERMS must store the date of closing of volumes and sub-files in their metadata.	Y
3.3.13	When classifying a record the user must be presented with the most recently created volume in the chosen sub-file by default.	Y

Ref	Requirement	Test
3.3.14	The ERMS must allow the creation of multiple concurrent open sub-files within any file.	Y
3.3.15	The ERMS must allow an administrative role to delete an empty volume.	Y
3.3.16	The ERMS must allow an administrative role to delete an empty volume and re-open the previous volume in the sub-file, in a single action, logging the event in the audit trail. *This is intended to correct an error which has resulted in the incorrect closure of a volume.*	Y
3.3.17	The ERMS should allow a "template" of sub-files to be created by an administrative role for a specified class, such that the template specifies the sub-files to be created automatically for each new file that is subsequently created in that class. *This is intended primarily for case management environments. As an example, a template in an insurance company might specify, for the class dealing with client insurance policies, the following sub-files: policy and amendments, internal correspondence, correspondence with medical specialists, billing, other client correspondence. Thereafter, every new file created in that class would automatically be created with these sub-files.*	Y
3.3.18	The ERMS must automatically close any open sub-files in a file whenever their parent file is closed.	Y
3.3.19	The ERMS must allow users to close volumes individually.	Y

3.4 Maintaining the Classification Scheme

This section starts with requirements for reclassifying, combining, splitting and copying classes (3.4.1 to 3.4.4). All these facilities are intended for exceptional circumstances only, such as organisational mergers or other re-organisation, or to correct clerical errors, or when the classification scheme is not well suited to the business. These facilities are not intended for routine use with a well-designed classification scheme. The requirements should be read together with 9.3.3 and 9.3.4. The section concludes with other requirements related to classification scheme maintenance (3.4.17 onwards).

Ref	Requirement	Test
3.4.1	The ERMS must allow an administrative role to relocate a class within the classification scheme in a single transaction. *In this context, "relocation" means reclassifying the class or file, that is moving it to another point in the classification scheme. The relocation can be to the same level in the classification scheme, or to any other level. Relocation implies several additional requirements that are described later in this section.*	Y

Ref	Requirement	Test
3.4.2	The ERMS must allow an administrative role to combine two classes in a single transaction. *In this requirement, "combine" is to be understood as follows: if a class is combined with another class,* ♦ *all the children and contents of the former class are relocated so that they become children and contents of the latter class;* ♦ *the former class is closed.*	Y
3.4.3	The ERMS must allow an administrative role to divide a class into two in a single transaction. *In this requirement, "divide" is to be understood as follows: if a class is divided:* ♦ *a new class is created as a child of the same parent class as the class being divided (this assumes all the requirements of creating a new class, such as metadata capture and inheritance);* ♦ *the user specifies a point in the contents of the class that is to be divided;* ♦ *all the contents of that class beyond that point (that is, with a higher classification code) are relocated to the newly created class.* *The contents of the class being divided can be any of the kinds of content allowed, namely classes, files, or records.*	Y
3.4.4	The ERMS should allow an administrative role to copy any class within the classification scheme in a single transaction. *In this requirement, "copy" is to be understood to mean creating a copy of the class and all its contents at another point in the classification scheme leaving the original in place. The copy can be to the same level in the classification scheme, or to any other level. Copying implies several additional requirements that are described later in this section.* *This facility is intended for use when replicating branches of a classification scheme, an act that is sometimes required (for example) when designing a part of the scheme that is not designed on a functional basis.* *Use of export followed by import will not be considered sufficiently easy to meet this requirement.*	Y

Ref	Requirement	Test
3.4.5	When any classes are relocated or copied the ERMS must ensure that the newly relocated or newly copied files and all their contents are reclassified with the classification codes for their new location in the classification scheme. *This means that every class, file, sub-file, volume, record and **component** that is relocated or copied acquires a new classification code and fully-qualified classification code.* *The rules for the allocation of the new codes are the same as the rules that would be followed when creating new classes, files, records etc.*	Y
3.4.6	The ERMS must not require an administrative role who is relocating, dividing, combining or copying classes to perform separate export and import actions. *The essence of this requirement is ease of use; the users must not be forced to execute a series of unrelated actions to achieve the desired outcome.*	Y
3.4.7	The ERMS must not allow any relocation or copying that would result in a data structure that is contrary to the rules implicit in the MoReq2 Entity-Relationship model (see section 13.2) or explicit in other requirements. Specifically, it must not allow any relocation that would result in: ♦ storing any sub-file(s) or volume(s) in a class of the classification scheme that has been configured not to allow sub-files or volumes (see 3.3.1, 3.3.2, 3.3.3); ♦ storing any record(s) directly in a class that already contains any file(s) or vice versa; ♦ storing any file(s) in a class that already contains any class(es) or vice versa.	Y
3.4.8	The ERMS must ensure that during relocation all electronic records remain correctly allocated to the class(es) and/or and file(s) being relocated; and that any sub-file(s), volume(s) and file(s) remain correctly related.	P
3.4.9	The ERMS must ensure that during copying all copies of electronic records remain correctly allocated to the new copies of the class(es) and/or file(s) and that copies of any sub-file(s), volume(s) and file(s) remain correctly related.	P
3.4.10	When any classes, files, volumes, sub-files or records are relocated or reclassified any closed files must remain closed, retaining their references to the classification scheme (classification codes) before the change.	Y

Ref	Requirement	Test
3.4.11	When any classes, files, volumes, sub-files or records are relocated or reclassified any open files must either: ♦ be closed, retaining their references to the classification scheme before the change, and cross-referenced to a new file in the changed scheme in metadata; ♦ be referenced to the changed scheme, but clearly retaining all prior references to the classification scheme before the change in metadata; according to the choice of the administrative role performing the relocation.	Y
3.4.12	When any classes are relocated or copied the ERMS must enable the optional inheritance of metadata by the classes and their contents (or the copies) from the new parent class. *This includes such elements as access permissions and security classifications.*	Y
3.4.13	When any classes are relocated or copied, the ERMS must be able to apply any inheritable retention and disposition schedules from the new parent class to the relocated or copy classes and their contents, in addition to the existing retention and disposition schedules. *This is the minimum functionality required; the ERMS may offer additional ways to treat the retention and disposition schedules.* *This may result in conflicts between schedules; if any conflicts arise these should be dealt with as in section 5.1 (especially 5.1.18 and 5.1.33).*	Y
3.4.14	When any classes are relocated or copied the ERMS must require an administrative role to enter as metadata the reason for the relocation or copying. *Entry of a reason is mandatory, as relocation and copying are exceptional, potentially endangering the integrity of the records if not carefully managed.*	Y
3.4.15	When any classes, files, or records are relocated or copied the ERMS must log their status prior to the relocation or copying in the audit trail.	Y
3.4.16	When any classes are relocated the ERMS must log the values of their metadata prior to the relocation. *Both of the above requirements are in support of the need to be able to determine the history of records that have been relocated.*	Y
3.4.17	The ERMS should enable an administrative role to mark a class or file as inactive to prevent any new files being added to that class or records being added to that file.	Y
3.4.18	The ERMS should allow an administrative role to delete an empty class.	Y

Ref	Requirement	Test
3.4.19	The ERMS must prevent the deletion of an electronic file or any part of its contents at all times. *This requirement is subject to the exceptions of:* ♦ ***destruction*** *in accordance with a retention and disposition schedule – as explained in 5.1.25;* *or* ♦ *deletion by an administrative role as part of an audited procedure – as explained in section 9.3.*	Y
3.4.20	The ERMS must allow an electronic file to be closed by user roles. *This is different than the corresponding requirement in MoReq, which limited this function to* ***administrators****.*	Y
3.4.21	The ERMS should be able to close an electronic file volume automatically on fulfilment of specified criteria to be defined at configuration, including at least: ♦ volumes delineated by an annual cut-off date; for example, the end of the calendar year, financial year or other defined annual cycle; ♦ the passage of time since a specified event; for example, the most recent addition of an electronic record to the volume; ♦ the number of electronic records which the volume contains. *Other criteria may be desirable in particular circumstances, for example when the size of the volume reaches the storage capacity of a removable disc.*	Y
3.4.22	The ERMS must make the contents of closed classes, files, sub-files and volumes as accessible for viewing as those that are open, without making any differentiation between open and closed. *In other words, users who are searching for or browsing through information using the ERMS must not have to be aware of whether files etc. are closed or open; and the same search facilities and access rules must apply.*	Y
3.4.23	The ERMS should allow users to create cross-references (that is, “see also” type links) between related files.	Y
3.4.24	The ERMS should support the ability to create multiple entries for an electronic record, in several electronic classes, files, sub-files or volumes, without duplication of the record or of the **document** on which it is based. *MoReq2 does not specify how this is achieved. One way of supporting this requirement would be to use pointers when capturing more than one record based on the same document.*	Y

Ref	Requirement	Test
3.4.25	The ERMS must provide reporting tools for the provision of statistics to administrative roles on aspects of activity within the classification scheme, including the numbers and sizes of classes, files, volumes, sub-files or records created, closed or deleted within a given period. *Reporting should be both overall and by any specified user or class.*	Y
3.4.26	The ERMS should provide ad hoc reporting capabilities on aspects of activity within the classification scheme.	P
3.4.27	Any user working with a class, file or record must be able to discover the context of that class, file or record, or in other words, the metadata and parent file or class(es); and must be able to navigate to these parents from the class, file or record. *It must be possible to discover the context without having to leave the class or file, in a way that allows work with the file to be continued without interruption.*	Y
3.4.28	Whenever any **keyword** of any file is changed, the ERMS must require an administrative role to enter the reason for the change.	Y
3.4.29	Whenever any keyword of any file is changed, the ERMS must keep a clear trace of its status prior to the change so that its history can be determined easily. *These controls over keyword changes are required to mitigate the risk of records being concealed by changes to keywords. Because keywords are used to find records, it is necessary to track any changes to keywords to avoid the possibility that a user would attempt to hide a record by changing its keywords.*	Y

4. CONTROLS AND SECURITY

This chapter brings together requirements for a wide range of controls which relate to the security of records. These requirements provide features needed to protect the characteristics of records defined in section 7.2 of ISO 15489.

It is essential that organisations are able to control who is permitted to access records and in what circumstances, as records may contain personal, commercial or operationally sensitive data.

Restrictions on access may also need to be applied to external users. For example, in some countries where freedom of information legislation permits access to selected public records, customers may wish to view records. Also some organisations may wish to share parts of their ERMS repository with partner organisations. Requirements for these controls are listed in section 4.1.

Any access to records and all other activities involving them and related documents or data also need to be logged in the audit trail to ensure legal admissibility and to assist in data recovery. Requirements for these audit trail controls are listed in section 4.2; these requirements address principally the record characteristics of **authenticity** and integrity defined in section 7.2 of ISO 15489.

Security of records also includes the ability to protect them from system failure by means of backup, and the ability to recover the records from backups. These requirements are listed in section 4.3; these requirements are related to the record characteristic of usability defined in section 7.2 of ISO 15489

Vital records are mission-critical records that need to be recovered rapidly after a disaster. These are addressed in section 4.4.

4.1 Access

Organisations need to be able to control access to their records and typically this is achieved by the specification and implementation of security policies, i.e. access to records is granted based on the business role an individual plays in the organisation. Users are usually managed centrally and simultaneously granted access rights to a number of corporate systems, including but not restricted to the ERMS.

It is not considered best practice to manage permissions in an ERMS simply by allocating individual permissions on individual entities to individual users. Access rights will therefore normally be granted to **roles** and/or groups to allow them to save and refer to records in specified classes or files within the classification scheme.

In addition to the entitlement to access specific parts of the classification scheme, permissions also restrict the actions that a user, role or group can perform on entities within the ERMS, such as inspecting their metadata or their contents, modifying or deleting them and creating or viewing entities of a particular type.

For example, a user role can search for and read records, but role-based security organisation may restrict the capability to search and read to particular sub-sets of the classification scheme.

Permissions can be applied to groups and be inherited by the group members. Applying permissions at the group level, rather than the user level improves the management of the ERMS over time as new users arrive, and existing users change and leave.

Through the assignment of roles in the ERMS multiple permissions can be granted to a user or group automatically. Later, when the user or group is removed from the role, all the permissions are automatically rescinded.

The ERMS must be able to limit the setting of these access rights to certain roles. In the table in 13.4, this is shown as belonging to administrative roles.

Note, however, that administrative roles are only implementing, from a system perspective, policy decisions taken by more senior management. The security polices and their allocation to individual end users, are typically based on the business needs of users to access information, the organisation's records policy and laws and regulations, such as information laws, data security laws, archival laws and industry regulations (see section 11.5).

In some environments, ERMS access permissions are managed entirely within the ERMS. In others, some permissions are managed using separate software, such as a network operating system utility. Either is acceptable for compliance with the following requirements.

The roles identified are "indicative" only. It should be the organisation that sets the number and the make-up of the roles that it uses and even whether it uses roles at all, according to its own requirements.

Ref	***Requirement***	***Test***
4.1.1	The ERMS must not allow any person to carry out any action in the ERMS unless the person is an authorised user who is successfully identified and authenticated. *MoReq2 does not specify the nature of the authentication mechanism. In many situations, a user-id and password mechanism is considered to provide sufficient authentication. Organisations using MoReq2 for procurement purposes need to ensure that an appropriate level of authentication is included.*	Y
4.1.2	The ERMS must allow administrative roles to allocate access to records, sub-files, files, classes and metadata to specified users and/or user roles and/or **user groups** and for specified periods of time.	Y
4.1.3	The ERMS must not limit the number of roles or groups that can be configured.	P
4.1.4	The ERMS must allow administrative roles to maintain permissions for all roles and groups. These determine the functionality, metadata elements, records or files to which the roles and groups have access, and the kinds of access allowed.	Y

Ref	Requirement	Test
4.1.5	The ERMS must allow administrative roles to use permissions to: ♦ restrict access to specific files or records; ♦ restrict access to specific classes of the classification scheme; ♦ restrict access according to the user's **security clearance** (where applicable); ♦ restrict access to particular features and functions (e.g. read, update and/or delete specific metadata elements); ♦ deny access after a specified date; ♦ allow access after a specified date. *The permissions should be used to allocate access according to the organisation's security policies.* *The level of granularity required is indicated in section 13.4.*	P
4.1.6	The ERMS should allow configuration to enable access by means of an integrated network log-on.	Y
4.1.7	The ERMS must allow administrative roles to add and remove users to and from roles and groups at any time. *It is acceptable for administrative roles to manage groups by means of separate directory management software.*	Y
4.1.8	The ERMS must allow the allocation of administration rights over different sections of the classification scheme to different administrative roles. *For example see the access control model in section 13.4.*	Y
4.1.9	The ERMS must allow administrative roles to mark an individual user as inactive, without deleting the user from the system. *It is acceptable for administrative roles to manage users by means of separate directory management software.*	Y
4.1.10	The ERMS must allow administrative roles to define the same access rights for user roles as for users. *This feature allows administrative roles to manage and maintain a limited set of role access rights rather than a larger number of individual users. Examples of roles might include Manager, Claims Processing Clerk, Security Analyst, Database Administrator.*	Y

Ref	Requirement	Test
4.1.11	The ERMS must be able to apply selections of access requirements across roles. *For examples see section 13.4.*	Y
4.1.12	The ERMS must allow an administrative role to set up and maintain groups of users. *Examples of groups might be Human Resources, Northern sales team.*	Y
4.1.13	The ERMS must allow a user to be a member of one group, more than one group or no group. *It is likely that some users will have different access requirements for different parts of the classification scheme. In all cases, users are assigned to groups by administrative roles in response to business needs and policies.*	Y
4.1.14	The ERMS must allow administrative roles to set up ad hoc lists of individual users in order to control access to specified parts of the classification scheme or records.	Y
4.1.15	The ERMS must restrict systems functions and related events to administrative roles only. *This is needed to protect the authoritativeness of electronic records.*	Y
4.1.16	The ERMS must allow only administrative roles to set up **user profiles** and allocate users to groups and roles. *See also section 13.4.*	Y
4.1.17	The ERMS must allow roles with ownership of records to specify which other users or groups can access those records. *See glossary for the MoReq2 usage of the term "**owner**". If the organisational policy allows, ownership should be with administrative roles.*	Y
4.1.18	The ERMS must restrict the ability to make changes, such as adding, amending and deleting **profiles** for groups, roles or users to administrative roles. *This includes attributes such as access rights, privileges, password allocation and management.*	Y

Ref	Requirement	Test
4.1.19	The ERMS must allow administrative roles to set up and manage rules to govern users' access to ERMS functions, so that different roles have access to different combinations of functions. The ERMS must allow such rules to be set up with at least the level of granularity (i.e. the amount of breakdown) shown in the illustrative access rights table in section 13.4. *Different organisations have different functional access control requirements. It is therefore not appropriate to attempt to define a generic model. Accordingly, this requirement specifies instead the level of detail of control that an ERMS must offer.*	Y
4.1.20	The ERMS must allow administrative roles to create roles additional to those shown in 13.4. *An organisation could define roles with specific access rights such as: **case worker**, manager etc.*	Y
4.1.21	The ERMS should provide an application programming interface to provide access to records by initiation from another application system.	N
4.1.22	If a user performs any search that includes content searching (typically, but not necessarily, a full text search or free text search), the ERMS must not include in the result list any record for which the user does not have the permissions to access. *This requirement is needed to prevent users employing text searches to investigate the contents of documents to which they are not allowed access.*	Y
4.1.23	If a user requests access to, navigates to, or searches for, without searching for content, any object such as a record, volume, sub-file, file or class which the user does not have the permission to access, the ERMS must provide one of the following responses (the response to be selected at system configuration or at a later time): ♦ provide no information about the object, thus providing no indication of whether the object does or does not exist; ♦ confirm the existence and (optionally) the owner of the object (display its file or record identifier) but not its title or other metadata. ♦ display title, type of entity (class, record etc.), date of creation and owner only; ♦ display title and other metadata of the object. *The option in the first bullet of this requirement specifies the same outcome as for content searches (see 4.1.22). The other three options intentionally offer other possibilities, appropriate in some organisations; they are shown here in order of decreasing security. They should be configured by administrative roles.*	Y

Ref	Requirement	Test
	This requirement applies only to access attempts that do not involve searching on record content. Searches on record content are addressed in 4.1.22, with which this requirement should be read.	
4.1.24	The ERMS should allow the responses specified in 4.1.23 to be selected for a class as an alternative to a system-wide setting at configuration time or later.	Y

4.2 Audit Trails

An audit trail is a record of actions taken which involve the ERMS. This includes actions taken by users or administrative roles, or actions initiated automatically by the ERMS as a result of system parameters. See the glossary at section 13.1 for a formal definition.

The audit trail shows whether business rules are being followed and ensures that unauthorised activity can be identified and traced.

In order to support accountability it is essential that the ERMS is able to log in the audit trail any action where any degree of automated or machine assisted processing is implemented within the system. Section 10.5 Casework provides examples of such an interface.

The audit trail is a key factor in enabling the ERMS to fulfil these requirements by maintaining a complete log of all the actions on every record (subject to the constraint of the level of security of the technical environment).

The volume of audit trail information can become large if all actions are audited. Consequently, in some implementations, management may decide that selected actions need not be included in the audit trail (after the date of the decision).

In many implementations, the on-line audit trail is periodically moved to off-line storage, the off-line copy being subject to deletion if and when the relevant records are disposed of, or if and when policies and legislation permit.

These are matters of management policy and/or legal/regulatory requirements. MoReq2 therefore includes system requirements to allow these actions, but does not establish the extent to which they are used.

Ref	Requirement	Test
4.2.1	The ERMS must keep an unalterable audit trail capable of automatically capturing and storing information about: ♦ any action taken on any record, any aggregate or the classification scheme; ♦ the user undertaking the action; ♦ the date and time of the action. *By way of illustration, the actions logged in the audit trail must include, but need not be limited to:* ♦ *capture of all electronic records;* ♦ *re-classification of an electronic file within the classification scheme (see 3.4.1);* ♦ *any change to any retention and disposition schedule;* ♦ *any **disposition** review actions carried out by administrative roles;* ♦ *the placing or removal of a **disposal hold** on an electronic file;* ♦ *any change made to any metadata associated with classes, electronic files or electronic records;* ♦ *amendment and deletion of metadata by a user;* ♦ *changes made to the access permissions;* ♦ *creation, amendment or deletion of a user or group;* ♦ *export or **transfer**;* ♦ *creation of a presentation;* ♦ *deletion/destruction of records.* *The term "unalterable" in this requirement means that it must be impossible for any user or administrator to change or delete any part of the audit trail. The level of assurance needed will depend on the organisation; the level of assurance that can be achieved will depend on the level of security of the underlying operating system and system software.* *The audit trail may, however, be subject to re-organisation and/or copying to off-line storage if required by, for example, database software, so long as its integrity remains intact.*	Y

Ref	Requirement	Test
4.2.2	Where the ERMS supports the transfer of audit trail data to off-line storage, the ERMS must support secure processes for managing the off-line data and demonstrate how off-line data can be brought back on-line as and when required; and the ERMS must ensure it is not possible for this mechanism to be used as a means of by-passing the controls imposed by the ERMS (for example, by simply moving audit trail data out of the ERMS and changing or deleting it externally to the system).	P
4.2.3	The ERMS should be able to log automatically in the audit trail any access to any record or **aggregation** and whether the access was to read, print or otherwise present it. *This is normally only required in highly secure environments.*	Y
4.2.4	The ERMS audit trail parameters must be configurable so that administrative roles can configure which actions are automatically logged.	Y
4.2.5	All changes to audit trail parameters must be audited in the audit trail. *It should never be possible to turn off the auditing of changes to audit trail parameters so that the ERMS does not record in the audit trail who changed them and when.*	Y
4.2.6	Once the audit trail parameters have been set, the ERMS must track actions automatically and must log information about them within the audit trail.	Y
4.2.7	The ERMS must maintain the audit trail for as long as is required by the organisation's records policy. *This often will be at least for the life of the records to which the audit trail refers. However, there may be situations in which other policies apply, for example periodic scrutiny of the audit trail followed by its destruction and replacement by a certificate of scrutiny.*	N
4.2.8	The ERMS must log in an audit trail all actions performed on records, volumes, sub-files, files, classes and retention and disposition schedules, regardless of whether the action affects one or more of them.	P
4.2.9	The ERMS must log in an audit trail all changes to metadata values that apply to the metadata elements listed in the MoReq2 metadata model.	P
4.2.10	Any annotation of or amendment to a record must be logged within the record's audit trail.	Y
4.2.11	The ERMS must automatically log in an audit trail all changes made to administrative parameters. *For example, if an administrative role changes a user's access permissions or reconfigures the audit trail.*	Y

Ref	Requirement	Test
4.2.12	The ERMS must ensure that audit trail data is available for inspection on request, so that a specific event can be identified and all related data made accessible.	Y
4.2.13	The ERMS must include features that allow all authorised users, including those who have little or no familiarity with the system, to search for information in the audit trail. *This is an ease of use requirement. The users may be external to the organisation, such as external auditors. Nonetheless, from the ERMS perspective, they will be users.*	P
4.2.14	The ERMS must allow users to search audit trails for specified events, objects (classes, records etc.), users, groups, roles, times, or time intervals.	Y
4.2.15	The ERMS must be able to export audit trail data for specified records, volumes, sub-files, files and classes without affecting the audit trail stored by the ERMS in any way save for the addition of an audit trail of the export process. *This functionality is to enable, for example, external auditors to examine or analyse system activity.*	Y
4.2.16	The ERMS must be able to capture and store, where applicable, any attempted violations of access control mechanisms (i.e. a user's attempts to access a record, volume, sub-file or file to which he is denied access). *For an illustration of circumstances which can allow attempts at violation, see 4.1.23. This cannot apply when the system is configured to hide from a user all knowledge of information to which the user does not have access permissions.*	Y

4.3 Backup and Recovery

Business and regulatory demands require that an ERMS be provided with comprehensive controls for regular backup of the records and metadata. It must also be able to recover records if any are lost because of, for example, system failure, accident or security breach.

Regular automated backup and recovery can be provided by the ERMS, by integration with the services of an Electronic Document Management System (**EDMS**), by a database management system operating with the ERMS, or by some other software. In this section, references to "the ERMS" can mean any of these, as appropriate to the setting.

In practice, backup and recovery functions may lie more with the organisation's IT operations area than by being divided between ERMS administrative roles.

Ref	Requirement	Test
4.3.1	The ERMS must provide or allow automated backup and recovery procedures that allow for regular backup of all or selected classes, files, records, metadata, administrative parameters, and the audit trail of the ERMS; and their recovery when needed.	Y
4.3.2	The ERMS must allow administrative roles to schedule backup routines by: ♦ specifying the frequency of backup; ♦ selecting classes, files or records to be backed up; ♦ allocating storage media, system or location for the backup (e.g. off line storage, separate system, remote site).	Y
4.3.3	The ERMS must allow only authorised administrative roles to restore from ERMS backups.	Y
4.3.4	When an ERMS restores from a backup, full integrity of the data including the audit trail must be maintained after the restore. *Records which have been correctly disposed and are present in the backup should not be restored except in exceptional circumstances,*	P
4.3.5	Where the ERMS features checkpoints and database roll-forward facilities, the ERMS must allow only authorised administrative roles to roll it forward.	P

4.4 Vital Records

Vital records are the records that are considered absolutely essential to the organisation's ability to carry out its business functions, in the short term, in the long term or both (see also the glossary). This can be either mission-critical in terms of its ability to cope with emergency/disaster conditions or to protect its long-term financial and legal interests.

The identification and protection of such records is of great importance to any organisation and it is likely that it is these records that will need to be recovered first in the event of a disaster.

Records may be considered as vital records either for the organisation as a whole or part of the organisation.

Ref	Requirement	Test
4.4.1	The ERMS must allow administrative roles to indicate that selected files or records contain, or are considered to be, "vital records". *This indication should be included as a metadata element.*	Y

Ref	Requirement	Test
4.4.2	The ERMS must provide two separate back-up operations: ♦ "full" backup, which backs up all (specified) ERMS data; ♦ "vital" backup, which backs up only the ERMS configuration and files and records identified as "vital". *Two back-up operations are used for the following reasons to allow:* ♦ *"vital" back-ups to be scheduled more often than "full" ERMS back-ups;* ♦ *"vital" back-ups to be taken onto different media and stored separately from (and possibly more securely than) "full" back-ups.* *It also provides for better managed ERMS restoration where restoring from "vital" back-ups can occur entirely independently of, and at a different time to, "full" restoration.* *As specified in section 4.3, backup can be performed either by the ERMS or by integration with some other software.*	Y
4.4.3	After recovering from a "vital" back-up the ERMS must be fully operational. *After restoring from a "vital" back-up many files and records will not be present. Other than this, however, the ERMS must not be in any way limited in its operation or the functionality that it provides to users.*	P
4.4.4	The ERMS should provide for two methods of restoring from a "full" back-up: ♦ restoration to a "clean" environment, in which the data from the "full" back-up overwrites and replaces the ERMS during the recovery operation; ♦ restoration over an existing environment, in which the data from the "full" back-up is merged back into an existing ERMS environment. *The first method of restoration will be common in organisations where "vital" back-ups are not taken. The second method of restoration will occur when an ERMS has previously been partially restored from a "vital" back-up and returned to normal operation; it then becomes necessary to merge in the "full" back-up without overwriting either the vital files and records that were previously restored or any new entities that have been added, or changes that have been made, to the ERMS in the interval since it was returned to full operation.* *If the ERMS supports two methods of restoring from a "full" back-up as outlined in 4.4.4, the "vital" back-up (if it exists) will always be restored first. There is no need to consider the restoration of a "vital" back-up over a "full" back-up.*	Y

Ref	Requirement	Test
	When undertaking a two-part system restoration in this way it may be necessary for administrative roles to resolve manually any conflicts that arise. For example, the classification scheme may be altered in one back-up when compared to the other.	
4.4.5	The ERMS must allow administrative roles to indicate that selected files or records are no longer considered vital. This action must be logged in the audit trail. *For example a lease agreement or contract might expire and therefore no longer be considered vital.*	Y

5. RETENTION AND DISPOSITION

This chapter lists requirements for the use of retention and disposition schedules to govern the retention and eventual fate of records from ongoing operations. Retention and disposition schedules define how long the records have to be kept by the ERMS, and how they may be disposed of. Requirements for retention and disposition schedules are listed in section 5.1; a formal definition is in the glossary.

The processes that can take place at the date specified by retention and disposition schedules are described in subsequent sections. Requirements for review processes are listed in section 5.2, and requirements for transfer, export and destruction are listed in section 5.3.

As explained in section 2.2 under the heading Electronic File, Sub-File and Volume, records can be managed in classes, files, sub-files and volumes, as appropriate to the business requirement. According to circumstances, retention and disposition schedules apply to classes, files and/or sub-files and/or volumes. Retention and disposition schedules can also be applied to **record types**, for example to apply short retention periods to sensitive personal data, or to apply long retention periods to engineering drawings. The resolution of conflicts between retention and disposition schedules is also allowed for.

MoReq2 includes the concept of “disposal holds”, which was not mentioned in the previous version of MoReq. Disposal holds are used in response to unexpected events to ensure that specified records are not destroyed. The common example is to ensure that records that are, or that may be, required as evidence in legal proceedings are not routinely destroyed as a result of a disposition decision.

5.1 Retention and Disposition Schedules

Ref	*Requirement*	*Test*
5.1.1	The ERMS must allow administrative roles, and only administrative roles, to create and maintain retention and disposition schedules.	Y
5.1.2	The ERMS must not limit the number of retention and disposition schedules.	P
5.1.3	The ERMS should be able to arrange retention and disposition schedules in a hierarchical structure resembling the structure of general and organisation-specific retention and disposition schedules authorised by appropriate mandates. *A hierarchical structure facilitates the management of numerous retention and disposition schedules.*	N
5.1.4	The ERMS must allocate a unique identifier to each retention and disposition schedule when it is created.	Y
5.1.5	The ERMS must allow a unique title to be entered for each retention and disposition schedule when it is created.	Y
5.1.6	The ERMS must maintain an unalterable history of changes and deletions (audit trail) that are made to retention and disposition schedules including the date of change or deletion, and user making the change.	Y

Ref	Requirement	Test
5.1.7	The ERMS must ensure that any amendment to a retention and disposition schedule is immediately applied to all entities to which the retention and disposition schedule is allocated.	Y
5.1.8	The ERMS must require an administrative role changing or deleting a retention and disposition schedule to enter a reason, and must store that reason in the audit trail. *Changes to, or deletions of, retention and disposition schedules must be controlled carefully to minimise the risk of records being destroyed inappropriately.*	Y
5.1.9	The ERMS must be capable of importing and exporting retention and disposition schedules.	P
5.1.10	The ERMS must ensure that every class, file, sub-file and volume always has at least one retention and disposition schedule. *This requirement is included to ensure that no entities are created without a retention and disposition schedule; and to improve usability.*	Y
5.1.11	The retention and disposition schedules applied by default to every new class, file, sub-file or volume should be inherited from their parent. *Where this is not possible (for classes at the top level of the classification scheme and if no inheritable retention and disposition schedule applies – see 5.1.18) a default retention and disposition schedule should be applied.*	Y
5.1.12	Every record stored directly in a class must always have at least one retention and disposition schedule assigned to it.	Y
5.1.13	The retention and disposition schedules applied by default to any new record stored directly in a class (see section 3.2 3.2.17) must be inherited from its parent class.	Y
5.1.14	The ERMS must allow an administrative role to apply a retention and disposition schedule to any class, file, sub-file, volume or record type at any time. *The phrase "at any time" means that an administrative role can replace a retention and disposition schedule or (if the system supports multiple retention and disposition schedules, see 5.1.16) apply an additional retention and disposition schedule to any class, file, sub-file, volume or record type. One example will be the replacement of a default retention and disposition schedule; another is the application of an additional retention and disposition schedule in response to a regulatory investigation. This may cause a conflict between retention and disposition schedules: see 5.1.23.*	Y

Ref	Requirement	Test
5.1.15	The ERMS should be able to apply a default retention and disposition schedule to record types. *This implies that record types can exist with no applied retention and disposition schedule. This is acceptable, as each individual record will have at least one retention and disposition schedule applied to it, because each record is held in a file or class and requirement 5.1.10 mandates that at least one retention and disposition schedule is applied to each file and class.*	Y
5.1.16	The ERMS must allow more than one retention and disposition schedule to be in force for any class, file, sub-file or volume. *This is required to manage real-life scenarios, which involve retention requirements arising from a range of mandates and business needs. This is illustrated by one example, chosen from many possible.* *In this example, a file has a single retention and disposition schedule, assigned for business reasons, as the records within it are not expected to be subject to legal or regulatory retention requirements. The retention and disposition schedule applying to this file also applies to many other files. At some point, it becomes apparent that it may be necessary to retain the file for a longer period than the current retention and disposition schedule allows, due to a business issue related to a safety case. At this point, it seems that the contents of the file may become subject to a regulatory control related to safety regulations; so a second retention and disposition schedule is applied to the file, taking this into account. At a later time, it may become apparent that the safety issue did not exist; in that event the second retention and disposition schedule can be removed, leaving the original one in place and active.*	Y
5.1.17	The retention and disposition of every record must be governed by the retention and disposition schedule(s) associated with the class, file, sub-file, volume and record type to which the record belongs; and by any applicable disposal hold(s) (see 5.1.34). *Once a retention and disposition schedule is applied, it governs the retention and disposition of records associated with the entity to which it is applied (unless it is overridden by a different retention and disposition schedule).*	Y

Ref	Requirement	Test
5.1.18	The ERMS must allow any retention and disposition schedule, and changes made to it, to be inherited down the hierarchy of the classification scheme, at the option of an administrative role. *Whether or not a retention and disposition schedule is inherited can be selected by an administrative role using any appropriate means. MoReq2 does not prescribe how this is achieved. Possibilities include:* ♦ *the option is selected when the retention and disposition schedule is created (in which case it applies whenever the retention and disposition schedule is applied);* ♦ *the option is selected whenever the retention and disposition schedule is applied (in which case it applies to all child entities);* ♦ *the option is selected when an entity is created for it to inherit the retention and disposition schedule(s) of its parent.*	Y
5.1.19	Each retention and disposition schedule must include either: ♦ a retention period (5.1.25) and a trigger event (5.1.25); or ♦ a disposition date.	Y
5.1.20	Each retention and disposition schedule must include: ♦ a disposition action (5.1.24); ♦ a reason.	Y
5.1.21	Each retention and disposition schedule should include a: ♦ a description; ♦ a mandate. *The mandate specifies the justification for the retention and disposition schedule. This is often a reference to a law, regulation or corporate policy.*	Y
5.1.22	When the retention period applicable to some record(s) because of a retention and disposition schedule reaches its end, the ERMS must automatically initiate the processing of the disposition decision. *This may mean that the decision is executed (subject to 5.2.4) or it may mean that action is required by an administrative role (see 5.1.23). Some organisations may prefer to implement the latter because of the risks involved with automatic execution.*	Y

Ref	Requirement	Test
5.1.23	When the ERMS is initiating a disposition decision (as in 5.1.22), if any other retention and disposition schedule applies with a different retention period end and/or with a different disposition decision then a conflict arises. It must be possible to configure the ERMS so that it automatically informs an administrative role if such a conflict arises, leaving the administrative role to resolve the conflict. *The phrase "must be possible to…" is included because it is not required that administrative roles intervene in all situations. It is acceptable for the ERMS to resolve a conflict automatically; but it must be possible to configure the ERMS to require administrative intervention in the event of conflict.* *A conflict can arise because* ♦ *some retention and disposition schedule(s) indicate that disposition is to be initiated while some other(s) indicate the opposite;* *and/or* ♦ *different retention and disposition schedules indicate different disposition decisions.* *In most cases it will be simple to determine which schedule will take precedence.* *These conflicts can arise in two scenarios:* ♦ *the conflicting schedules all apply to the entirety of an aggregation (such as a file);* ♦ *schedules apply to both an aggregation and to some records within it (because the latter apply to specified record types that occur within the aggregation).*	Y

Ref	Requirement	Test

Administrative intervention may be required where it is not practical to define rules that correctly resolve these conflicts. For example:

- *two retention and disposition schedules, derived from different legal mandates may specify different retention periods. Normally, the decision will be to retain the records until the end of the later of the two end dates;*

- *one retention and disposition schedule may specify a date by which certain records must be disposed of (typically because of data protection legislation). If this date is earlier than the retention date of a conflicting retention and disposition schedule, then the decision will depend on the relative weight of the two mandates and/or on business needs.*

These situations can arise when a document has a record type permitting the application and inheritance of a disposal rule to that record from the record type rather than from the aggregation in which it is contained.

The administrative role's resolution may include any of the following:

- *remove one or more of the conflicting schedules from the aggregation or records affected;*

- *change one or more of the conflicting schedules to remove a conflict;*

- *remove all the conflicting schedules and apply a new schedule;*

- *use the exceptional deletion features specified in section 9.3.*

All of these actions, if they are not carefully controlled, could raise concerns about good governance of the records. Therefore any of these actions – changing retention and disposition schedules or deleting records – must be the subject of written procedures. In some settings, further management controls such as the division of the tasks, will be appropriate.

If the resolution results in some record(s) remaining in an aggregation that otherwise would not have been retained, the organisation may also need to have guidelines for their storage. This may include leaving the aggregation in place, or the relocation (see section 3.4) of the remaining record(s).

Ref	Requirement	Test
5.1.24	The ERMS must allow at least the following disposition actions (as defined in 5.1.20) for each retention and disposition schedule: ♦ retain permanently; ♦ present for review; ♦ destroy automatically; ♦ destroy after authorisation from an administrative role; ♦ transfer to an archive or another repository (see glossary). *There are risks involved with implementing the "destroy automatically" option outlined in the above requirement; organisations will need to balance these risks against the benefits of automation.*	Y
5.1.25	The ERMS must allow at least the following combinations of trigger events and retention periods (as defined in 5.1.19) to be specified: ♦ passage of a specified period of time after the class, file, sub-file or volume is opened; ♦ passage of a specified period of time after the class, file, sub-file or volume is closed; ♦ passage of a specified period of time since the most recent record has been classified to the class, file, sub-file or volume; ♦ passage of a specified period of time since a record has been retrieved from the class, file, sub-file or volume; ♦ passage of a specified period of time after a specified external event (which event is described in the schedule, and will be notified to the ERMS by an administrative role rather than being detected automatically by the ERMS) (for example, "…after contract signature" or "…100 years after date of birth"); ♦ "permanent" to indicate long term preservation of the records. *While the above is generally inclusive, it is possible that some organisations will want to impose additional activating events and/or additional retention periods.* *Any number of external events can be linked to different retention and disposition schedules.*	Y
5.1.26	The ERMS should not limit the length of retention periods.	P

Ref	Requirement	Test
5.1.27	The ERMS must support retention periods of time up to at least one hundred years for requirement 5.1.24**.** *This maximum is suggested as an arbitrary period intended to avoid any practical limitation. While it is improbable that any ERMS will exist for one hundred years, a requirement of this nature will allow records to be transferred to future systems without the need to revise retention and disposition schedules.*	P
5.1.28	The ERMS must be able to restrict the management of the disposition process to administrative roles.	Y
5.1.29	The ERMS must log in the audit trail and notify to an administrative role all automatic disposition actions**.**	Y
5.1.30	The ERMS must automatically notify an administrative role when any review action becomes due.	Y
5.1.31	The ERMS must allow an administrative role to delegate any notified review action to a reviewer role for action.	Y
5.1.32	The ERMS must allow an administrative role to amend any retention and disposition schedule (apart from its unique identifier, see 5.1.6).	Y
5.1.33	When an administrative role moves electronic files or records between classes of the classification scheme, the ERMS must offer the option to: ♦ allow the retention and disposition schedule of the destination class to replace the existing retention and disposition schedule(s); or ♦ enable an administrative role to select the appropriate retention and disposition schedule(s). *This refers to moving records, as is permitted on an exception basis, in 9.3.3 and 9.3.4. On the rare occasions this functionality is used, administrative roles will need to take great care over the assigning or changing of retention and disposition schedules, especially for vital records.*	Y
5.1.34	The ERMS must enable a disposal hold to be placed on a class, file, sub-file, or volume by an authorised user.	Y
5.1.35	A disposal hold must not prevent any retention period from running and completing, *However, see 5.1.36.*	P

Ref	Requirement	Test
5.1.36	The ERMS must prevent any entity subject to a disposal hold, along with its contents (child entities) if they exist, from being deleted or being subject to any disposition decision. *Deletion is described in section 9.3.*	Y
5.1.37	The ERMS must restrict the removal of a disposal hold to an authorised user.	Y
5.1.38	When an authorised user applies or removes a disposal hold, the ERMS must capture and store the following information about it, at a minimum in the audit trail and preferably as metadata: ♦ the date the hold was applied or removed; ♦ the identity of the authorised user; ♦ the reason for the hold.	Y
5.1.39	The ERMS should allow an authorised user to apply several disposal holds, each specifying the same reason, to a group of classes, files, sub-files or volumes as a bulk operation. *This requirement allows the authorised user to apply holds for the same reason to several classes, files, etc.*	Y
5.1.40	The ERMS should allow the lifting of multiple disposal holds (citing the same reason) simultaneously, as a bulk operation, by an authorised user.	Y
5.1.41	The ERMS should allow a class, file, sub-file or volume to be subject to multiple disposal holds simultaneously, either because they are applied to the entity, and/or because they are applied to a higher-level entity. In either event the restrictions on disposition and other functionality imposed by disposal holds must remain in place until the last disposal hold affecting the entity is lifted.	Y
5.1.42	The ERMS should allow an authorised user to search and report on all entities subject to a specified disposal hold.	Y
5.1.43	The ERMS should allow an authorised user to set, change, and delete a “reminder” that notifies the user of the existence of a specified disposal hold on a specified date.	Y

5.2 Review of Disposition Actions

In some environments, the retention and disposition schedules are used to govern disposition without a review. In others, retention and disposition schedules trigger a review of the specified disposition action on an aggregation that has reached the date or event specified in the schedule. The review may consider metadata, contents or both in deciding on the disposition action (a further retention period, transfer to another system, destruction or combination of these).

The disposition of certain records is subject to laws and regulations. Reviews of disposition actions must be performed in a way which is consistent with these laws and regulations. Reviews must also take account of any appraisal policy and procedures set down for the organisation. Where appropriate, this must be done in co-operation with (and sometimes exclusively by) responsible archival authorities. Further discussion of these issues is beyond the scope of MoReq2.

Ref	*Requirement*	*Test*
5.2.1	The ERMS should automatically notify an administrative role of all retention and disposition schedules which will come into force in a specified period of time.	Y
5.2.2	The ERMS must support the review process by presenting classes, files, sub-files and volumes to be reviewed, together with their metadata and retention and disposition schedule information. *In practice, this implies features for navigating forward, back etc. within and between files, and from/to the metadata for files and records.*	Y
5.2.3	The ERMS must be able to maintain links between different **renditions** of the same records and enable disposition actions to be carried out on them simultaneously.	Y
5.2.4	The ERMS must allow the reviewer to take at least any of the following actions for each class, file, sub-file or volume during review: ♦ mark for destruction, immediately or at a future date (see section 5.3); ♦ mark for transfer (see section 5.3), immediately or at a future date; ♦ mark for a further review, immediately or at a future date; ♦ mark for indefinite retention. *This may be achieved by the application of different retention and disposition schedules, or by other means.*	Y
5.2.5	The ERMS must automatically log the date of a review.	Y
5.2.6	The ERMS must allow the reviewer to enter comments into the class, sub-file, volume, or file's metadata to log the reasons for the review decisions.	Y
5.2.7	The ERMS must keep an unalterable history of all decisions taken by the reviewer during reviews, including reasons. *The decisions should be stored as metadata and possibly also in the audit trail.*	Y

Ref	Requirement	Test
5.2.8	The ERMS should alert an administrative role if a conflict arises because a file that is due for destruction is referred to in a link from another file. It must pause the destruction process to allow the following remedial action to be taken: ♦ confirmation by the administrative role to proceed with or cancel the process; ♦ generation of a report detailing the files or record(s) concerned and all references or links for which it is a destination.	Y

5.3 Transfer, Export and Destruction

Organisations may need to move records from their ERMS to other locations or systems for archival or other purposes. This is referred to here as “transfer”.

Reasons for transfer may include:

- permanent preservation of the records for legal, administrative or research reasons;
- the use of devolved or external services for the medium term or long term management of the records.

This action often results in the records being transferred to a different ERMS environment.

The term transfer is used even though, initially, only a copy is sent to the other location or system. The records originally residing in the ERMS are retained and only destroyed upon verification that the transfer has been successful.

The term export, on the other hand, refers to the process of producing a copy of complete aggregations, files and records for another system, while the records remain on the originating system – the process does not delete them.

In effect the transfer process takes place in two stages – export of a copy with all associated metadata and audit trails, followed by destruction of the original.

In each case, the requirement is to execute the transfer, export or destruction in a controlled manner. Decisions must be taken on the metadata and audit trails at the same time as actions are carried out on the records to which they relate.

In this context “destruction” is different from “deletion”. Deletion of records under other circumstances is covered in section 9.3.

Ref	Requirement	Test
5.3.1	If a formal MoReq2 XML schema has been published[5] the ERMS must be able to export records in a form compliant with this schema. *See also requirement 6.2.1 regarding the bulk import of records. Taken together these two requirements address the interoperability of MoReq2 compliant ERMSs.*	P
5.3.2	Whenever an ERMS transfers or exports any record, it must transfer or export all its components and must preserve the correct relationships between them.	P
5.3.3	The ERMS must provide a well defined process to transfer records, together with their associated metadata and audit trail information, to another system or to another organisation.	P
5.3.4	The ERMS should be able to export records and their metadata in the form of a submission information package as defined by the OAIS standard (see appendix7). *See the similar requirement for dissemination information packages at 11.7.12.*	Y
5.3.5	Whenever the ERMS transfers or exports any class, file, sub-file or volume, the transfer or export must include: ♦ (for classes) all files and records in the class; ♦ (for files) all volumes and sub-files in the file; ♦ all records in all these files, sub-files or volumes; ♦ all or selected metadata associated with all of the above; ♦ all or selected audit trails for all of the above. *Although the ERMS must be capable of exporting all metadata and audit trails, not all of these are always required by every target system.*	P
5.3.6	Whenever the ERMS exports or transfers any records with their metadata, it must include any implicit metadata in explicit form. *In other words, all the metadata values that apply to any class, file, sub-file, volume or record must be shown explicitly, even if it has been stored only implicitly. See appendix 9.3 for examples.*	P

[5] At the time of writing, development of an XML schema for MoReq2 is about to start. See http://ec.europa.eu/transparency/archival_policy for details.

Ref	Requirement	Test
5.3.7	The ERMS must be able to do either or both of the following when exporting or transferring any set of records: ♦ export or transfer with the records the retention and disposition schedules applied to those records, in a manner which allows the schedules to be re-applied to the records in the destination system; ♦ print one or several reports showing the retention and disposition schedules to be applied to each set of records, and the characteristics of these schedules.	P
5.3.8	The ERMS must be able to do either or both of the following when exporting or transferring any set of records: ♦ export or transfer with the records the access controls for those records, in a manner which allows the controls to be re-applied to the records in the destination system; ♦ print one or several reports showing the access controls applicable to each set of records, and the characteristics of these controls.	P
5.3.9	The ERMS must be able to transfer or export a file or the contents of a class in one sequence of operations, such that: ♦ the content and structure of its electronic records are not changed; ♦ all components of an electronic record, (when the record consists of more than one component) are exported as one unit; ♦ all links between the record and its metadata and audit trails are retained; ♦ all links between classes, files, sub-files, volumes and records are retained so that they can be reconstituted in the receiving ERMS.	P
5.3.10	When the ERMS is transferring or exporting files and/or sub-files and/or volumes, if any of them include pointers to records stored in other files (see 3.4.24) then the ERMS must transfer or export the complete record, not a pointer. *This is required so as to make sure that there are no difficulties of pointer resolution between the transferring or exporting system and the receiving system.*	Y
5.3.11	The ERMS must be able to transfer and export records in the format in which they were captured.	Y
5.3.12	The ERMS must be able to transfer and export records in any format(s) into which records have been **rendered**.	Y

Ref	Requirement	Test
5.3.13	The ERMS must be able to migrate records marked for transfer or export into specified transfer format(s). *For example, an approved XML or other open format.* *This requirement is to cover long retention periods where records must be automatically rendered into approved long-term preservation formats after a defined period of time without affecting the integrity and authenticity of the records.*	P
5.3.14	The ERMS must retain all aggregations, records and other information that are being transferred, at least until confirmation of a successful transfer process. *This is a procedural safeguard, to ensure that records are not destroyed before successful transfer-in is reported from the recipient.* *See 9.2.30 and 9.2.31 for requirements regarding the reporting of any failure during the transfer process.*	Y
5.3.15	The ERMS must destroy aggregations, records and other information that are being transferred when it receives confirmation that the transfer process is successful, save for metadata that is retained as a **stub**. *See 5.3.19.*	Y
5.3.16	The ERMS should be able to export the entire contents of a class of the classification scheme in one sequence of operations, ensuring that: ♦ the relative location of each file in the classification scheme is maintained, so that the file structure can be reconstructed; ♦ sufficient metadata to rebuild the whole parent class branch is retained and moved with the contents of the class.	P
5.3.17	The ERMS should provide the ability to add user-defined metadata elements required for archival management purposes to electronic files selected for transfer.	Y
5.3.18	The ERMS must ensure that, when a record marked for destruction is destroyed, all its renditions are destroyed. *Where the same record appears in more than one file (3.4.24 in section 3.4) then the record and its renditions should be removed from the file when it is destroyed but should not be finally deleted until all occurrences of the record have been destroyed.*	Y

Ref	Requirement	Test
5.3.19	The ERMS must have the ability to retain a **metadata stub** for: ♦ classes; ♦ files; ♦ sub-files; ♦ volumes; ♦ records stored directly in a class; which have been destroyed or transferred. *In some environments it is desirable to retain information about records which have been destroyed. The metadata in question should include at least the date of acquisition and all the metadata relevant to identify uniquely each record and its relations to the classification scheme. See the MoReq2 metadata model.* *This is so that the organisation can still know what records it has held and the dates they were destroyed or disposed of, without incurring the overhead of keeping all the detailed metadata for files and records.*	Y
5.3.20	The metadata stub (see 5.3.19) must include at least the following: ♦ date destroyed or transferred; ♦ fully qualified classification code; ♦ title; ♦ description; ♦ user responsible for destruction or transfer; ♦ reason for destruction or transfer (this can be a reference to a retention and destruction schedule or a manually-entered reason); ♦ any reference given by the system to which the records have been transferred, to facilitate the retrieval of transferred records.	Y
5.3.21	The ERMS must allow an administrative role to specify a subset of additional metadata elements which will be retained as metadata stubs.	Y
5.3.22	The ERMS must be able to export metadata stubs when records are exported. *This is required to allow migration between ERMSs.*	Y
5.3.23	The ERMS must allow information to be exported more than once.	Y

Ref	Requirement	Test
5.3.24	Whenever the ERMS exports or transfers information, it should be able to produce on request a report listing the records exported or transferred according to their security categories.	Y

6. CAPTURING AND DECLARING RECORDS

Overview

This chapter covers requirements relating to the process of capturing records into an ERMS. The first section (6.1) covers the standard capture process. The following section (6.2) covers the **bulk import** of records from other systems and this is followed by a section devoted to e-mail because of its particular importance (6.3). Section 6.4 concerns record types and section 6.5 covers integration with scanning and imaging systems.

Terminology

The term "capture" is used with its natural English language sense, in an information management/ information technology context. Here, "capturing" information means saving it in a computer system. This is consistent with the archival meaning of "capture", ("the act of recording or saving a particular instantiation of a **digital** object") given in the InterPARES 2 Project Terminology Database[6].

It follows that ERMSs can capture a variety of information. An ERMS can capture records, metadata, and in some cases documents, among others.

The fact that an ERMS can (in some cases) capture documents as well as records suggests that the term "capture" is imprecise, because capturing a record involves more processes than capturing a document that is not a record. For example, capturing a record includes the processes of classification, **registration**, and locking against change whereas this is not necessarily the case for documents. Hence the term "declare" is sometimes used synonymously with "capture" in the case of records. However, "declare" can also apply to a document that starts outside the ERMS, or to a document that has already been captured by the ERMS.

This lack of precision should have no negative impact on the clarity of MoReq2.

More formal definitions are given in the glossary in section 13.1.

6.1 Capture

Electronic documents that are made or received in the course of business processes originate from both internal and external sources. The electronic documents will be in various formats, be produced by different authors and may be received either as single documents or as documents comprising several components (see glossary for the MoReq2 definition of "component").

Some records are created within the organisation, in the course of its business processes. Others are received through various communication channels, for instance electronic mail, facsimile, letter post (optionally to be scanned), by hand, and at variable arrival rates and volumes. A flexible capture system with good management controls is required to capture documents so that their diverse requirements are addressed.

[6] See http://www.interpares.org/ip2/ip2_terminology_db.cfm.

Ref	Requirement	Test
6.1.1	The ERMS capture process must provide the controls and functionality to allow users to: ♦ capture electronic records regardless of **file format**, method of encoding or other technological characteristics, with no alteration of their content; ♦ ensure that the records are associated with a classification scheme; ♦ ensure that the records are associated with one or more file(s) or class(es). *File format is defined in the glossary. The requirement is to be able to capture any file format.* *The requirement to capture records in any file format is not intended to be testable, and it does not imply that the ERMS needs to be able to make presentations (see glossary) of all possible formats. MoReq2 therefore does not list the kinds of formats that may be captured, as formats vary over time with the evolution of software. However, for the avoidance of doubt, the kinds of records to be included can be diverse; they might include, for example, the following kinds of records frequently used in office settings:* ♦ *output from desktop applications such as office suites;* ♦ *e-mails (see section 6.3);* ♦ *audio;* ♦ *databases;* ♦ *portable document formats;* ♦ *scanned images;* ♦ *video;* ♦ *web pages.* *In some situations, the ERMS may also need to capture other kinds of record such as:*	P

Ref	Requirement	Test
	♦ *blogs* ♦ *compressed files (sometimes referred to as "archives", applying an IT meaning of the term);* ♦ *electronic calendars;* ♦ *electronic forms;* ♦ *geographical information system data;* ♦ *information from other computer applications e.g., accounting, payroll, computer aided design;* ♦ *instant messaging systems;* ♦ *multimedia documents;* ♦ *records of web-based transactions;* ♦ *records which include links to other records;* ♦ *software source code and project documentation;* ♦ *structured data (e.g. EDI transactions);* ♦ *webcasts;* ♦ *wikis.* *These lists are not complete.*	
6.1.2	The ERMS must not impose any practical limit on the number of records which can be captured in any class, file, sub-file or volume, nor on the number of records which can be stored in the ERMS. *Large numbers of records in volumes etc. will tend to make the system difficult to use in some settings, and so is not generally advisable. This requirement is intended to allow for situations in which large numbers are unavoidable, such as some transactional environments.*	P
6.1.3	When capturing a record made up of several components, the ERMS must capture all of its components.	P

Ref	Requirement	Test
6.1.4	When capturing an electronic record that has more than one component, the ERMS must allow the record to be managed as a single unit, retaining the relationship between the components, and retaining the record's structural integrity. *Examples of such records are:* ♦ *web pages with embedded graphics;* ♦ *a word-processed document linked to a spreadsheet.* *In some cases, the components will be related by links that do not work if simply copied into the ERMS repository. For example, many web pages contain links to graphics and other objects with addresses (URLs) that are external to the repository; and linked spreadsheets typically contain links to addresses (operating system filenames) external to the repository. See next requirement.*	P
6.1.5	When capturing an electronic record that has more than one component the ERMS should modify the record, if necessary, to preserve the ability to present it. This is likely to mean that the ERMS changes the internal references (links) within some of the components. *This requirement applies only to file formats specified for the ERMS – it is not intended to apply to unspecified formats. Examples may include:* ♦ *HTML pages that include links to graphics and other objects;* ♦ *spreadsheets that include links to other spreadsheets.* *Making such changes is contrary to the general principle of not changing the content of records, but is unavoidable if records that include components etc. are to be stored in their original formats without losing all functionality and fidelity. The changes will generally be acceptable so long as the changes are logged in the ERMS audit trail (see next requirement). An alternative approach involves rendering the record into some other file format (such as **PDF/A**) that preserves the static appearance; see requirement 11.7.8; however, even though this avoids changing links, it is likely to result in losing them.*	P
6.1.6	When the ERMS changes references within records during capture, it must log automatically all details of the changes made in its audit trail.	Y
6.1.7	The ERMS must automatically capture the file format (see glossary), including the **version**, of each component when it is captured and must store it in the metadata of the component. *This is required to support the digital preservation of records – their accessibility over time. See section 11.7.*	P

Ref	Requirement	Test
	Some information about file format is usually implicit in the component's filename extension, e.g. ".htm" or ".pdf"; and on occasion it is ambiguous, e.g. ".doc" can specify several unrelated formats. However, the extension alone frequently does not indicate the file format version and sometimes not even the file format itself. This will be acceptable in many cases, though it may not suffice in cases where long-term preservation is needed, or where precision is needed (for example, precision of colour space). *File formats are numerous and are subject to frequent change. It therefore is not realistic to expect an ERMS to capture information for all file formats. It is therefore acceptable for the ERMS to:* ♦ *specify a list of file formats that can be recognised;* ♦ *rely on reference to an established file format registry – preferably one designed specifically to support digital preservation.* *In either case, the using organisation needs to satisfy itself that the range of file formats included is sufficient for its preservation requirements.*	
6.1.8	The ERMS record capture process must validate the values of metadata entered into the ERMS when records are being captured, at a minimum according to the rules in the MoReq2 metadata model. *See also 6.1.34 in this section.*	Y
6.1.9	The ERMS should support validation of metadata elements using check digit algorithms. *For example, files may be identified by a sixteen-digit credit card number, of which the last digit is a check digit computed from the other fifteen digits using the mod 10 algorithm.* *Provision of an application program interface for this feature, allowing organisations to introduce their chosen algorithm, should normally be considered acceptable.*	Y
6.1.10	The ERMS must allow users to capture an electronic record even if the application used to produce the record is not present. *For example a user may receive a project plan and a CAD/CAM drawing as attachments in an e-mail. If the user does not have access to the project plan or CAD/CAM applications, then the user may not be able to view the attachments. Despite this, the user should be able to capture the attachments as records in the ERMS. The ERMS may provide "viewer" software that allows the user to view these records; this is not required by MoReq2.*	Y
6.1.11	The ERMS must be able to capture metadata about records consistent with the MoReq2 metadata model.	Y

Ref	Requirement	Test
6.1.12	The ERMS should be able to capture automatically values from fields defined by an administrative role within the specified document types, using these values automatically to populate metadata elements as specified in the MoReq2 metadata model. *The functionality needed for this requirement applies only to specific kinds of electronic objects, for example letters produced using a specified template and a specified word processor.* *Many documents including some office documents and .**PDF** files include user-configurable metadata elements. It should be possible to configure the ERMS to capture automatically the values of these elements and retain them with the record.*	Y
6.1.13	The ERMS must allow the capture of all metadata elements specified at system configuration, and must retain them with the electronic record in a persistently-linked relationship at all times.	Y
6.1.14	The ERMS should allow users who wish to capture a record but who are unable to provide all the mandatory metadata values for it to store it temporarily in the ERMS. *In other words, the ERMS should provide a means of storing records without all their metadata, that is without completing the normal capture process. This implies exception reporting and progress monitoring; it does not imply any requirement to treat such records as normal records for the purpose of export, transfer, rendition etc. MoReq2 does not specify how this is achieved.* *Only the editable metadata values can be changed at a later stage, and fixed metadata (e.g. e-mail transmission data) must remain unchanged.*	Y
6.1.15	The ERMS must ensure that the values of some elements of the metadata of the electronic record can only be updated by authorised users and administrative roles, consistent with the rules in chapter 12.	Y
6.1.16	The ERMS must ensure that all records are assigned to at least one class, file (or its sub-file if appropriate), as appropriate, when captured.	Y

Ref	Requirement	Test
6.1.17	The ERMS should support automated assistance in capturing electronic documents, by automatically extracting as much metadata as possible, for as many kinds of document as possible. *The rationales for this requirement are to:* ♦ *minimise the amount of data entry performed by users (experience shows that in many environments, a requirement to enter metadata can cause users to reject the system);* ♦ *increase the accuracy of metadata.* *The metadata elements involved, and the kinds of documents for which this is possible, will depend on the environment. Some guidance is given in the metadata model.*	N
6.1.18	The ERMS must support automated assistance in the capture of outgoing and internal documents (e.g. memoranda or word-processed letters in a specified layout and file format) as records, by automatically extracting the following metadata from them: ♦ document date (as in the body of the document); ♦ recipient(s); ♦ any copy recipient(s); ♦ subject line (title); ♦ author(s); ♦ internal reference (typically shown as “our reference”); to the extent that these are present. *MoReq2 does not specify the software or formats for office documents or e-mail. The metadata extraction may be achieved by locating metadata within the record, by using a template to identify the metadata and populate a blank document, or by any other means.*	Y
6.1.19	The ERMS must log the capture date and time of a record both as metadata and in the audit trail. *If the date and time are part of the unique identifier of the record, and as long as they can be explicitly extracted from this number, it is not necessary to store the date and time separately.* *MoReq2 does not specify the accuracy of the time needed. Most ERMSs record time to an accuracy of one second or better.*	Y

Ref	Requirement	Test
	Some legislative frameworks call for time stamping to be performed against a certified device or authority. Where this is the case it should be accommodated in a chapter zero.	
6.1.20	For every captured record, the ERMS must be able to present on-screen the metadata, including that specified at configuration time. *The metadata specified at configuration time may consist of any or all elements from the relevant section of chapter 12.*	Y
6.1.21	The ERMS must ensure that all mandatory metadata is present for every captured record.	Y
6.1.22	During capture of a record the ERMS must prompt the user to enter any required metadata that has not automatically been captured.	Y
6.1.23	The ERMS must support the assignment of multiple keywords (or key terms) to each class, file, sub-file and record. *MoReq2 does not require the ability to assign keywords to volumes.*	Y
6.1.24	The ERMS should allow an administrative role to configure whether keywords are mandatory or optional, at configuration time, for each of classes, files and sub-files.	Y
6.1.25	The ERMS must allow more than one entity (class, file, etc.) to be created using the same combination of keywords.	Y
6.1.26	The ERMS should allow a user creating an entity to populate its keyword values by copying them all from another entity, in one action.	Y
6.1.27	The ERMS should allow a user to enter the identifier of one or more languages for any record.	Y
6.1.28	The ERMS must provide a capability for the keyword values and other metadata element values to be picked from, or validated against, controlled vocabularies (or lists of permitted terms). *For example, by means of a pick list or thesaurus. See also requirement 11.8.11.*	Y
6.1.29	The ERMS must allow entry of further descriptive and other metadata at the time of capture and/or at a later stage of processing.	Y
6.1.30	The ERMS must warn the user if an attempt is made to capture an object with a title which already exists in the same entity or to re-title an object with a title which already exists in the same entity. *See also 11.8.6.*	Y

Ref	Requirement	Test
6.1.31	The ERMS must be able to reserve the ability to amend the title of an electronic record for an administrative role or other authorised user. *This facility can be used or not used, at the option of the organisation.*	Y
6.1.32	When a user is capturing a document that has more than one version, the ERMS must allow the user to choose at least one of the following: ♦ declare all versions as one record; ♦ declare one specified version as a record; ♦ declare each version as an individual record.	Y
6.1.33	The ERMS should be able to provide automated support for decisions on the classification of electronic records by means of at least one of the following: ♦ making only a subset of a classification scheme accessible to a user or role; ♦ suggesting the classes or files used most recently by that user; ♦ suggesting the classes or files used most frequently by that user; ♦ suggesting classes or files by inferences drawn from record metadata elements (for example, significant words used in the title or e-mail subject line); ♦ suggesting classes or files by inferences drawn from the record contents.	P
6.1.34	The ERMS should allow the process of capturing a record to be completed by a more than one user. *The ERMS should allow the capture process to be divided between users; typically this will mean that one user enters some metadata then passes the electronic record to another user, who enters the remaining metadata and classifies the record.*	Y
6.1.35	The ERMS should provide simple workflow facilities to enable simple routing for checking and approving a document before capture, logging the decisions taken, who took them, and allowing a reason to be entered by each. *Note that this requires only basic workflow features. It intentionally stops short of the full workflow features described in chapter 10.*	Y

Ref	Requirement	Test
6.1.36	The ERMS should provide an application programming interface to receive and capture in real time individual records and transactions provided by another application or system. *As mentioned in section 1.4, ERM functionality may be required as part of a wider system This may require the ERMS to receive records from another system, for example, a corporate business application such as Customer Relationship Management (CRM) or a line of business application, via an Application Programming Interface (API) to enable the ERMS to capture individual records.*	N
6.1.37	Where possible, the ERMS should issue a warning if a user attempts to capture an e-mail record which has already been captured into the same file or (if classified directly into a class) the same class. *MoReq2 does not define how the e-mail is identified; however, the internet message ID may be suitable.* *There are several cases in which this is not logically possible, for example where the e-mail record has been captured into a file to which the user is denied access.*	Y
6.1.38	Where possible, the ERMS should issue a warning if a user attempts to capture a record (other than an e-mail, as this is dealt with by 6.1.37) that has the same content as another record which has already been registered in the same file or (if classified directly into a class) the same class.	Y
6.1.39	Where possible, the ERMS should issue a warning if a user attempts to capture a record (other than an e-mail, as this is dealt with by 6.1.37) that has the same values of identifying metadata as another record which has already been registered in the same file or (if classified directly into a class) the same class. *The identifying metadata for this requirement is:* ♦ *Title;* ♦ *Date;* ♦ *Author;* ♦ *Addressee.*	Y
6.1.40	Where possible and appropriate, the ERMS should be able to provide a warning if an attempt is made to capture a record which is incomplete or inconsistent in a way which will compromise its future apparent reliability. *For example, a purchase order without a valid electronic signature or an invoice from an unrecognised supplier.*	N

Ref	Requirement	Test
6.1.41	The ERMS must allow an administrative role (not user roles) to add a record to a previously-closed volume, provided that the date of the record is not later than the date of closure. When this takes place: ♦ the ERMS must require that an administrative role adds a reason to the metadata of both the volume and the record, to explain why this exception has taken place; ♦ the ERMS must automatically log this in the audit trail. This action must not update the date of closure stored in the metadata. *This facility is intended to be used to rectify user error, e.g. if a volume was closed unintentionally. For this reason, it is important that the exception causing this action is properly documented.* *MoReq2 does not mandate how this is achieved. It may be achieved by temporarily re-opening the closed volume, or by other means.*	Y

6.2 Bulk Importing

Records may reach the ERMS in bulk in a number of ways. For example:

- a bulk transfer from a compatible EDMS;
- a bulk transfer from a compatible ERMS;
- as a single compatible data file containing a series of records of the same type (e.g. daily invoices);
- from a compatible scanning or imaging system;
- records from a hierarchy of operating system folders.

The ERMS needs to be able to accept these, and must include features to manage the capture process and maintain the content and structure of the imported records.

During bulk import the ERMS needs to capture the same information as the normal capture process – namely the records themselves and their metadata. It also needs to classify the records – extending the classification scheme if necessary (see 3.1.12) – and possibly capturing audit trail information. Finally, bulk import needs to allow for the processing of exceptions and errors.

At the time of writing, the development of an XML schema for MoReq2 is planned. This schema is expected to implement the MoReq2 metadata model, and to provide an ideal protocol for the bulk import of electronic records from a MoReq2-compliant ERMS.

Ref	Requirement	Test
6.2.1	If a formal MoReq2 XML schema has been published, the ERMS must be able to perform a bulk import of records in a form compliant with this schema. *See also requirement 5.3.1 regarding the export of records. Taken together these two requirements address the interoperability of MoReq2 compliant systems.*	P
6.2.2	The ERMS must provide the ability to capture transactional records generated by other systems. This must include: ♦ supporting predefined batch file transaction imports; ♦ providing editable rules to customise the automatic capture of the records; ♦ validation to maintain data integrity. *MoReq2 does not specify how this ability is provided.*	P
6.2.3	The ERMS must be able to capture automatically the metadata associated with records during a bulk import (allowing for manual input of missing or incorrect metadata).	P
6.2.4	Where the ERMS captures the metadata of some record(s) during import, it must validate it using the same rules as would be used for the manual capture of the records(s). Where this validation process finds errors (such as absence of mandatory metadata, or format errors) it must bring these to the attention of the user performing the importation, in all cases identifying the metadata involved, and logging errors and actions in the audit trail. *In ideal cases, the record(s) being imported will have metadata that complies fully with the metadata model. In other cases, the metadata may be non-compliant. In these cases, several outcomes are possible; MoReq2 does not mandate any one outcome. Possible outcomes include:* ♦ *The entire importation is cancelled;* ♦ *Importation of the record that has non-compliant metadata is cancelled;* ♦ *The user is required to choose between correcting the error and cancelling importation of the affected class;* ♦ *The data is imported as a temporary incomplete record (this resembles the requirement that capture can be divided between users, see 6.1.34).*	Y
6.2.5	The ERMS must be able to import audit trail records that show the history of the imported record(s).	Y

6.2.6	The ERMS must not import audit trail records into its audit trail; it must store imported audit trail records separately. *The imported audit trail records must be maintained separately so as to avoid producing a mechanism that allows administrative roles to change or compromise the integrity of the audit trail. MoReq2 does not specify how this is achieved; it may involve storing the imported audit trail as a record alongside the imported records, or as a separate entity recognised as an audit trail imported from another system.*	Y
6.2.7	The ERMS must provide facilities to manage input queues. *Facilities such as the following are expected:* ♦ *view the queues;* ♦ *pause any or all queues;* ♦ *re-start any or all queues;* ♦ *delete a queue.*	Y
6.2.8	The ERMS must enable an administrative role to (optionally) set the ERMS to close classes, files and volumes automatically after they have been imported. *For example, on the merger of two organisations it may be necessary to close down branches of the structure so that records can no longer be added to them.*	Y

6.3 e-Mail Management

Definitions

As a verb, “e-mail” refers to a mechanism for transmitting messages between “agents” (in this context, the term” agent” has a precise technical meaning; more detail is not required for an understanding of MoReq2).

The standard protocol used for e-mailing is defined by the Network Working Group documents RFC 2821 and RFC 2822 (see appendix 7). MoReq2 uses RFC 2821/RFC 2822 as the basis of its working definition of “e-mail”.

As a noun, “e-mail” is usually used to refer to a document that contains the complete data of a single e-mail transmission. However, although RFC 2822 defines the syntax for e-mail transmissions, there are no standards that define the data format that should be used when e-mail transmissions are captured as documents.

In other words, even though e-mail applications from different suppliers can freely transmit messages (because they observe the e-mail protocols defined in RFC 2821/ RFC 2822) it is not possible to capture an e-mail from one application as a document and be sure that another e-mail application will be able to read it back. Each e-mail supplier uses its own proprietary format(s) for capturing e-mail. For this reason, accurate automated extraction of metadata from messages cannot be guaranteed.

Use and issues

e-Mail is used for sending documents (in the form of messages and as attachments) within and between organisations. The characteristics of e-mail management software (in particular the lack of standardisation for formats explained above), combined with user attitudes towards e-mail, can make it difficult to apply records management functionality to e-mail. Organisations need to be able to enforce procedures and management controls to:

- capture all inbound and outbound e-mails and attachments;

and/or to:

- capture e-mails and attachments according to pre-defined rules;

and/or to:

- provide users with the capability of capturing selected e-mails and attachments

In some countries the legal ownership of e-mail is unclear, and in some situations automatic capture of e-mails into an ERMS may be inappropriate. Where this is the case, the latter two options should be considered during configuration.

Furthermore, e-mail has become the default means of communication for many organisations and an important one for others. In some organisations, much e-mail traffic is ephemeral. Each organisation needs to decide which of the above alternatives represents the most appropriate compromise for its situation:

- The first option results in the capture of any ephemeral e-mails as well as those that are meaningful records;
- The second option relies on successfully configuring appropriate rules and filters;
- The third option requires the users to assess the relevance and importance of items and there is a risk that they will not all do so reliably.

MoReq2 allows for ERMS support for all three approaches. The procedures and management controls are beyond the scope of MoReq2.

Ref	*Requirement*	*Test*
6.3.1	Whenever an e-mail is captured, the ERMS must by default capture it in a format that retains its header information.	Y
6.3.2	The ERMS must support the capture of e-mails in an integrated way, such that the capture can be performed by a user from within the e-mail application, without the user needing to switch to the ERMS. *Close integration is essential for effective use of an ERMS. For example the user should be able to "drag and drop" from the e-mail client into the ERMS, choose a "capture" command from within the e-mail client or the e-mail client should indicate which e-mails have been captured into the ERMS. The essence of this requirement is that the user must not have to switch to the ERMS application to capture e-mails.*	Y

Ref	Requirement	Test
	MoReq2 also permits, but does not require, the capture of e-mails in other, less closely integrated, ways.	
6.3.3	It must be possible to configure the ERMS at configuration time so that it operates in one of the following ways when a user sends an e-mail: ♦ it automatically captures the e-mail; ♦ it determines whether to capture the e-mail according to pre-defined rules; ♦ it automatically prompts the user, giving the user an option to capture the e-mail; ♦ it takes no action (and thus relies on the user to initiate a capture if appropriate). *Regardless of which way is chosen, it is acceptable for the ERMS to require the user to classify records manually and enter some metadata manually.*	Y
6.3.4	It must be possible to configure the ERMS at configuration time so that it operates in one of the following ways when an ERMS user receives an e-mail: ♦ it automatically captures the message, unless it has already been captured; ♦ it determines whether to capture the e-mail according to pre-defined rules; ♦ if the e-mail has not already been captured it automatically prompts the user, giving the user an option to capture it; ♦ it takes no action (and thus relies on the user to initiate a capture if appropriate). *Regardless of which way is chosen, it is acceptable for the ERMS to require the user to classify the record manually and enter metadata manually.*	Y

Ref	Requirement	Test
6.3.5	The ERMS must support automated assistance in the capture of outgoing and incoming e-mails, with and without attachments, as records, by automatically extracting the following metadata from them: ♦ e-mail date sent (and in some settings, time); ♦ recipient(s); ♦ any copy recipient(s); ♦ subject line (title); ♦ sender; ♦ embedded electronic signature; ♦ certification service provider; to the extent that these are present. *This requirement specifies the capture of "sender" for e-mail messages. This is not always the same as the author, for example when a secretary sends a message on behalf of an executive. The capture of "sender" is specified here as a conscious compromise, it being impossible to reliably capture the author automatically. Organisations should consider the need for manual procedures to ensure the correctness of the author metadata.* *Appendix 9 provides guidance on the interpretation of e-mail metadata.*	P
6.3.6	Users should be able to capture an e-mail record to a sub-file, file or class by dragging it from an e-mail client (technically, a Mail User Agent) to a specified sub-file, file or class in the ERMS. *The sub-file, file or class can be represented in the e-mail client window or in a separate window.*	Y
6.3.7	The ERMS must allow a user to choose how to capture an e-mail message with attachment(s) as: ♦ the e-mail message only, without attachment(s); ♦ the e-mail with its attachment(s), as one record made of linked components; ♦ the attachment(s) only, each or any as individual records. *This applies to sent and received messages.* *The last of these three options results in attachment(s) being captured without the context of the e-mail with which they were transmitted.*	Y

Ref	Requirement	Test
6.3.8	Where an e-mail and its attachment(s) are captured at the same time but as separate records, the resultant records should be linked automatically by the ERMS. *The ERMS should allow a user to navigate the cross-reference link between the records so as to discover each of the attachment records from the e-mail record and the e-mail record from any of the attachment records.*	Y
6.3.9	Whenever an attachment is captured as a separate record, the ERMS must require appropriate record metadata values to be captured and/or entered for it.	Y
6.3.10	When capturing an e-mail message, the ERMS must by default populate the Title metadata with the "subject" field of the message. *Appendix 9 provides guidance on the interpretation of e-mail metadata.*	Y
6.3.11	The ERMS must allow a user who is capturing an e-mail message to edit the record title. *This is intended to allow users to correct inappropriate or to clarify imprecise e-mail titles, or to make the titles more meaningful.* *The e-mail title is separate from the subject line (title) of the e-mail; the latter will remain as part of the message regardless of the content of the e-mail title.*	Y
6.3.12	If a user captures an e-mail delivery status notification report (where these are supported) for an e-mail which has been captured as a record, the ERMS should be able to link the two automatically. *Examples of delivery status notifications are non-delivery reports and delivery confirmations. The link should allow a user to navigate between the records so as to discover each of the notifications from the e-mail record and the e-mail record from any of the notifications.*	Y
6.3.13	The ERMS must enable the automatic capturing of metadata belonging to e-mails and their attachments as outlined in the MoReq2 metadata model.	Y
6.3.14	The ERMS must allow "date sent" and "date received" metadata to be entered manually. *This is to allow for situations in which the dates held in the e-mail message are not appropriate for the business setting (see introduction to this section for explanation of how this may occur). A configuration option to disable this facility will be acceptable.*	Y

Ref	Requirement	Test
6.3.15	A user must be able to capture into the ERMS, in a single operation, several manually-selected e-mails as: ♦ one record; or as ♦ a set of records, one per e-mail; at the user's option.	Y
6.3.16	The ERMS should be able to identify automatically and capture all the e-mails related to an e-mail specified by a user, in a single operation, capturing them as: ♦ one record; or as ♦ a set of records, one per e-mail; at the user's option. *RFC 2822 Section 3.6.4. "Identification fields" describes how the optional SMTP header fields "References:" and "In-Reply-To:" can be used in conjunction with the "Message-ID:" field to identify related e-mail messages, sometimes referred to as the 'thread of the discussion.'*	Y
6.3.17	The ERMS must allow a user who is capturing an e-mail message in a proprietary format to save it in multiple, including open, formats *It may be useful for an ERMS to enforce e-mail saving criteria based on the retention and disposition schedule. The e-mail contents of files with a short retention period could be stored in a proprietary format, but those with longer schedules could be saved into an open format.*	Y
6.3.18	Whenever address fields captured from an e-mail header appear in the metadata of an e-mail record, the ERMS must ensure that it captures the optional "display name" (if present) of any mailbox listed as well as the "address-spec" address; for example, 'Jan Schmidt' rather than 'js97@xyz.int'.	Y

6.4 Record Types

Record Type describes characteristics of records that are not (and usually cannot be) defined in the classification scheme. This can include specific:

- metadata attributes;
- retention requirements;

- access controls;
- kind of document (e.g. contract, CV, disciplinary report).

A record's record type usually corresponds to the document type of the document from which the record was made.

Ref	*Requirement*	*Test*
6.4.1	The ERMS must support the definition and maintenance of record types.	Y
6.4.2	All records in the ERMS must have exactly one record type.	Y
6.4.3	The ERMS must restrict the definition and maintenance of record types to an administrative role.	Y
6.4.4	The ERMS must allow an administrative role to restrict the creation of records of specified record types to specified groups of users, based on their business needs.	Y
6.4.5	The ERMS must allow an administrative role to define one record type as the default record type, which can be used by all users who are allowed to capture records.	Y

6.5 Scanning and Imaging

When planning for the implementation of an ERMS, **physical records** in the form of paper or microform often need to be considered.

There are two main issues:

- existing records that are held on paper or microform and may need to be referred to in conjunction with electronic records;
- documents on paper that continue to be received or created by the organisation, but which the organisation wishes to hold as electronic records in the ERMS.

This section deals with the scanning (imaging) of paper-based and microform documents, so that they can be captured into the ERMS as electronic records. It includes several requirements that address details of the scanning process.

Scanning can be organised in the following ways:

- centralised;
- local or workgroup;
- outsourced or subcontracted;

or in any combination. These ways are described briefly below.

Centralised scanning is most appropriate for high-volume capture, typically using fast scanning equipment specifically designed for bulk input, together with specialist scanner operators.

Local or workgroup scanning takes place close to the receiving users and is appropriate for low-volume activity, where the person doing the scanning needs a knowledge of the business, or when dictated by the geographic distribution of the organisation. This typically uses scanners with lower capacity and speed; these are sometimes multi-functional devices.

Outsourced or subcontracted scanning – this can be considered for a number of reasons related to cost-effectiveness:

- where there is a large amount of scanning to be done as a one-off exercise;
- where sufficient human resources are not readily available in the organisation;
- where sufficient accommodation and/or equipment are not readily available in the organisation;
- where the scanning and/or storage are not site-dependent.

The rest of this section sets out key requirements to be considered in provision of an integrated ERMS and scanning solution. The requirements apply only where scanning facilities are part of the ERMS. Many of the requirements can also be interpreted for use when scanning is outsourced.

Ref	*Requirement*	*Test*
6.5.1	The ERMS must be capable of integration with at least one scanning solution. *The scanning solution provides the interface with the scanning equipment and allows the operator to perform several processes related to scanning such as rotating and de-speckling.*	Y
6.5.2	The ERMS scanning feature should support both monochrome and colour scanning. *Many applications do not require colour scanning.*	Y
6.5.3	The ERMS scanning feature must be capable of saving images in standard formats, including, but not limited to: ♦ TIFF (see TIFF 6.0 Specification); ♦ JPEG (see ISO 15444, required only if colour is supported); ♦ PDF/A (see ISO 19005).	Y
6.5.4	The ERMS scanning feature must be capable of saving images at different resolutions. *Ideally the scanning feature should provide a menu of options, programmable for the input of different types of document.*	Y
6.5.5	The ERMS scanning feature should be capable of saving images in colour or greyscale and at different resolutions.	Y

Ref	Requirement	Test
6.5.6	The ERMS scanning feature must be capable of handling standard paper sizes, including, but not limited to: ♦ A4; ♦ A3. *See ISO 216 for the definition of A4 and A3.*	Y
6.5.7	The ERMS scanning feature should have Optical Character Recognition (OCR) functionality. *OCR functionality produces text from a scanned image. Some kinds of OCR are sometimes referred to as Intelligent Character Recognitions, or ICR. For simplicity, MoReq2 refers to both as OCR.*	Y
6.5.8	Where the ERMS includes OCR functionality the ERMS should be capable of managing the scanned image and the text resulting from the OCR as a single record. *In other words, the OCR text should be regarded as metadata of the record rather than as a record in its own right.* *MoReq2 does not require that users be able to view the OCR text, as its purpose is to allow full text searching (see next requirement).*	Y
6.5.9	Where the ERMS includes OCR functionality it should support full text searching based on the text.	Y
6.5.10	The ERMS scanning feature should be capable of recognising and capturing individual documents in a bulk scanning process. *MoReq2 does not specify how this should be done. Common solutions rely on the recognition of patch codes, patch sheets, barcodes or blank sheet inserts.*	Y
6.5.11	The ERMS scanning feature must be capable of automatically sending scanned images to a queue after scanning. *For example, indexing, quality assurance.*	Y
6.5.12	The ERMS should include a facility for the inspection of the scanned images. *This includes the ability to accept or reject images; and, when they are rejected, to request a re-scan.* *The inspection may be carried out by a scanner operator, by a dedicated quality check user, or by other users who perform quality checking only as a part of their work*	Y

Ref	Requirement	Test
6.5.13	The ERMS scanning feature should allow an administrative role to set a threshold for image information content such that below the threshold an image is discarded as representing a blank page.	Y
6.5.14	The ERMS scanning feature should be able to store scanner set-up parameters (such as single/double sided, resolution, contrast, brightness) for different document types.	Y
6.5.15	The ERMS should allow users to annotate images. *This feature can be used to note exceptional scanning problems, or to make notes (much as handwritten annotations are sometimes used with paper documentation).*	Y
6.5.16	If the ERMS allows users to annotate images that are held as records, it must prevent the alteration and removal of these annotations. *This is required for records only; it is not required for other images. It is intended to prevent records from being modified (or from appearing to be modified) temporarily.*	Y
6.5.17	If the ERMS allows users to annotate images that are held as records, it must store with each annotation the identity of the user making the annotation and the time and date, in an unalterable way *This is required for records only; it is not required for other images. It is intended to ensure that any annotations are appropriate and traceable.*	Y
6.5.18	The ERMS scanning feature should log each scanning session with the following details: ♦ user login; ♦ workstation identifier; ♦ time and duration; ♦ session identifier; ♦ batch identifier(s); ♦ number of documents (if applicable); ♦ number of images scanned; ♦ number of images after removal of blank pages (if blank pages are removed automatically).	Y

Ref	Requirement	Test
6.5.19	The ERMS scanning feature should be able automatically to capture relevant metadata when scanning zoned forms. *A zoned form is one which includes areas defined in the scanning software as containing data to be scanned. The information outside the defined zones is not scanned, thus reducing image size and reducing storage and bandwidth requirements.*	Y
6.5.20	Where the ERMS scanning feature includes automatic capture of metadata it should be able to interpret this information for automated classification. *This feature is especially useful in casework environments, where paper records frequently bear case identifiers that contain sufficient information to classify the record – see section 10.5.*	Y
6.5.21	The ERMS should be capable of the bulk import of scanned images and their metadata. *See section 6.2 for further requirements regarding bulk import.*	Y
6.5.22	The ERMS should be able to display thumbnails of scanned images as an aid to navigation and searching.	Y
6.5.23	The ERMS must allow users to capture scanned images as records.	Y

7. REFERENCING

This chapter brings together requirements for referencing the entities (classes, files, sub-files, volumes and records) within a classification scheme, Section 7.1 lists requirements for Classification Codes and those for System Identifiers are listed in section 7.2.

All of the entities stored in the ERMS repositories (classes, files, sub-files, volumes, records etc.) need identifiers. These identifiers are needed to:

- Allow the software to process the entities;
- Allow users to retrieve, refer to, and use, the entities.

MoReq2 uses the following terminology to describe these identifiers:

- An identifier required for software usage is called “System Identifier”. This can be used by users as well as by software in some cases;
- A hierarchical identifier applied to entities in the classification scheme hierarchy and intended for users is called “Classification Code”;
- Other Identifiers are named as needed, for instance “Retention and Disposition Schedule Identifier”.

The difference between System Identifiers and Classification Codes is illustrated in the following three diagrams. These diagrams are also referred to later in the chapter.

Figure 7.1 shows a part of a fictitious, but realistic classification scheme. It shows some of the classes; each class has a class title (as required by 3.2.4).

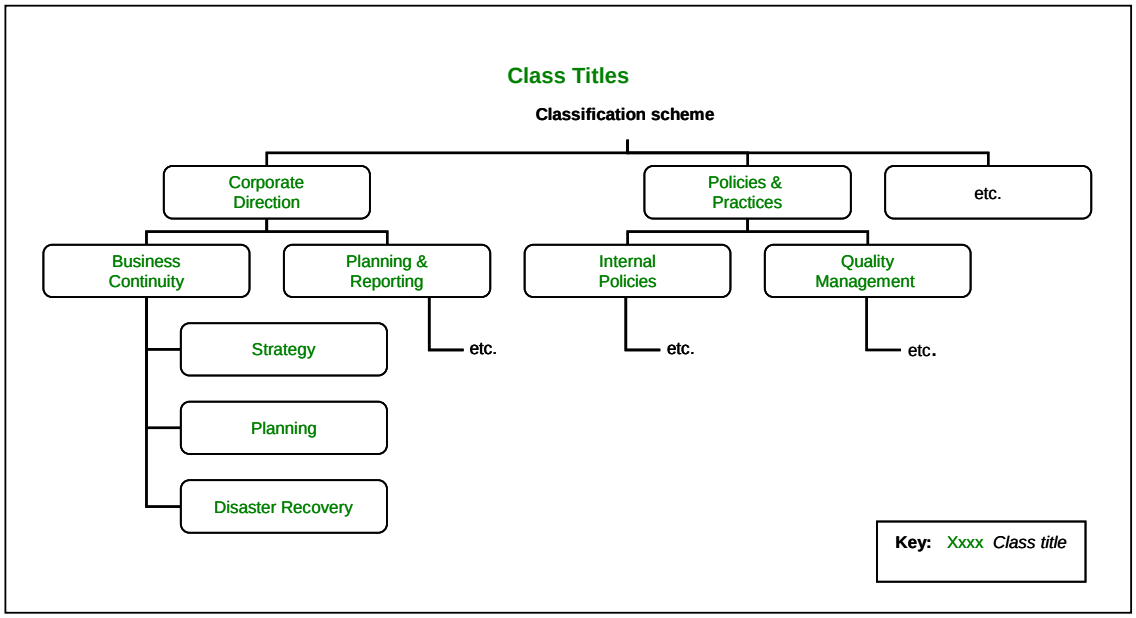

Figure 7.1

Each class is allocated a System Identifier, as shown in figure 7.2.

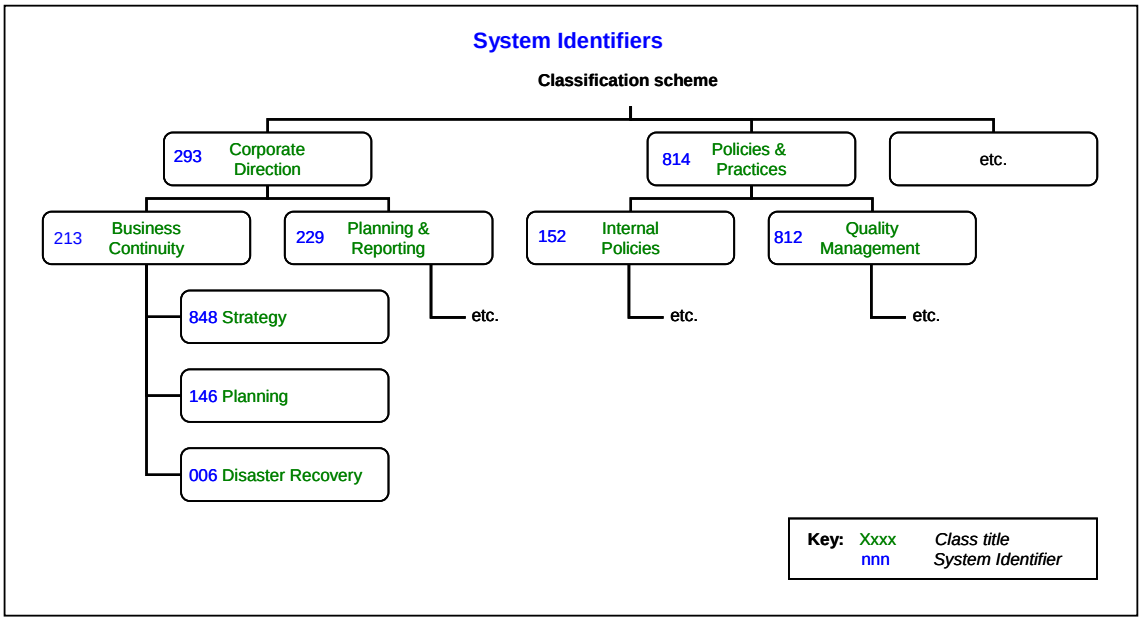

Figure 7.2

Note that the System Identifiers shown here are short and simple, purely for illustration. In reality they are likely to be longer and more complicated in structure. By way of illustration, an example of a System Identifier based on the "Globally Unique Identifier" algorithm is 0c7220e3-5646-44c4-82b0-67832c1efa1c.

Classes are also allocated a Classification Code. As specified in the requirements below, this can take several forms; one example is shown in figure 7.3.

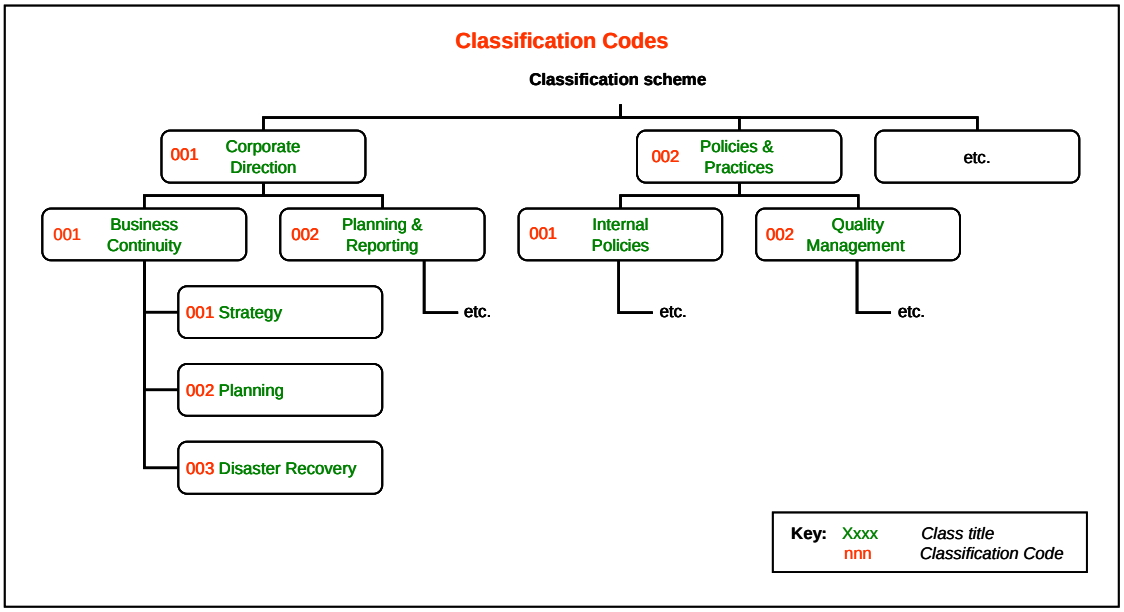

Figure 7.3

Here also, the Classification Codes are shown as relatively simple, for illustration.

Each class has a Classification Code that can be combined with the Classification Codes of its parent classes to make a "Fully-Qualified Classification Code". So, for example, the Fully-Qualified Classification Code of the class *Disaster Recovery* is 001-001-003. It is constructed as follows:

- start with the Classification Code of its highest parent in the hierarchy (001, being the Classification Code of the class *Corporate Direction*);
- add the Classification Code of its next parent down in the hierarchy (001, being the Classification Code of the class *Business Continuity*), making 001-001;
- repeat the previous step until the nearest parent class is reached (in this simple example, there are no repeats);
- add the Classification Code of the class (003), being the Classification Code of the class *Disaster Recovery*), producing the Fully-Qualified Classification Code 001-001-003.

Records and components are also allocated classification codes, to allow them to be referenced uniquely.

The expected usage determines the degree of uniqueness required. System Identifiers generally must be unique within one ERMS "instance" or "network node" at a minimum, and network-wide by preference. Fully-Qualified Classification Codes must be unique within a classification scheme, though because they are built up hierarchically, the individual Classification Codes may be unique only within one node (e.g. a class or sub-file) of the hierarchy.

Where uniqueness across a network is required, it is desirable that system identifiers should be based on an acknowledged standard that guarantees global uniqueness (that is, uniqueness across all systems at all times). This is also desirable for standalone, or non-networked, applications, so as to allow for possible future growth and for potential merger or acquisition activities. Several such standards have been proposed, none of which has a dominant position; MoReq2 therefore does not mandate the use of a specific standard for this purpose.

7.1 Classification Codes

Ref	*Requirement*	*Test*
7.1.1	Whenever a new occurrence of any of the following is created in or captured by the ERMS, the ERMS must associate with it a Classification Code: ♦ class; ♦ file; ♦ sub-file; ♦ volume; ♦ record; ♦ component.	Y
7.1.2	The ERMS must ensure that all Fully-Qualified Classification Codes are unique within a classification scheme hierarchy.	P

Ref	Requirement	Test
7.1.3	The ERMS must ensure that all Classification Codes and all Fully-Qualified Classification Codes retain the required degree of uniqueness regardless of any relocation (see requirement 3.4.1).	Y
7.1.4	The ERMS must be able to store Classification Codes as metadata elements of the entities to which they refer.	Y
7.1.5	The ERMS should allow the formats of Classification Codes and Fully-Qualified Classification Codes to be specified by an administrative role at configuration time. It should allow the following features of Classification Codes to be defined, for each level of the hierarchy: ♦ numeric, alphabetic or alphanumeric; ♦ presence or absence of leading zeroes; ♦ minimum length (in the case of leading zeroes); ♦ starting value; ♦ increment.	Y
7.1.6	Fully-Qualified Classification Codes must consist of a concatenation of Classification Codes separated by a separator character.	Y
7.1.7	The ERMS should allow the separator characters in Fully-Qualified Classification Codes to be selected from, at a minimum: ♦ " " (space); ♦ "-" (dash); ♦ "/" (forward slash); ♦ "." (dot). *For example, a Classification Code of 001-001-003 (as in the introduction above) could therefore be shown as any of the following, depending on the choices made for leading zeroes and separator at configuration time:* ♦ *1 001 003;* ♦ *001-001-003;* ♦ *1/1/3;* ♦ *001.001.003..*	Y

Ref	Requirement	Test

Remembering that requirement 3.2.7 allows for global prefixes and extensions these might also be shown as:

- *corporate/1/1/3;*
- *001.001.3.pt.*

7.1.8 The ERMS must allow an administrative role to specify, when a new class is created, whether its child entities will have Classification Codes generated automatically by the ERMS or provided by the user/an external application. The ERMS must either: P

- generate each Classification Code automatically and prevent users from inputting it manually, and from subsequently modifying it (for example, a sequential number, as in the example above);

or:

- allow an authorised user or an external application to provide the Classification Code, but subsequently prevent them from modifying it.

An example of the first option is if a new class titled "Incident Management" is added under the class "Business Continuity" in the example shown in figure 7.3; in this example, it would be allocated the Classification Code 004, as shown in figure 7.4.

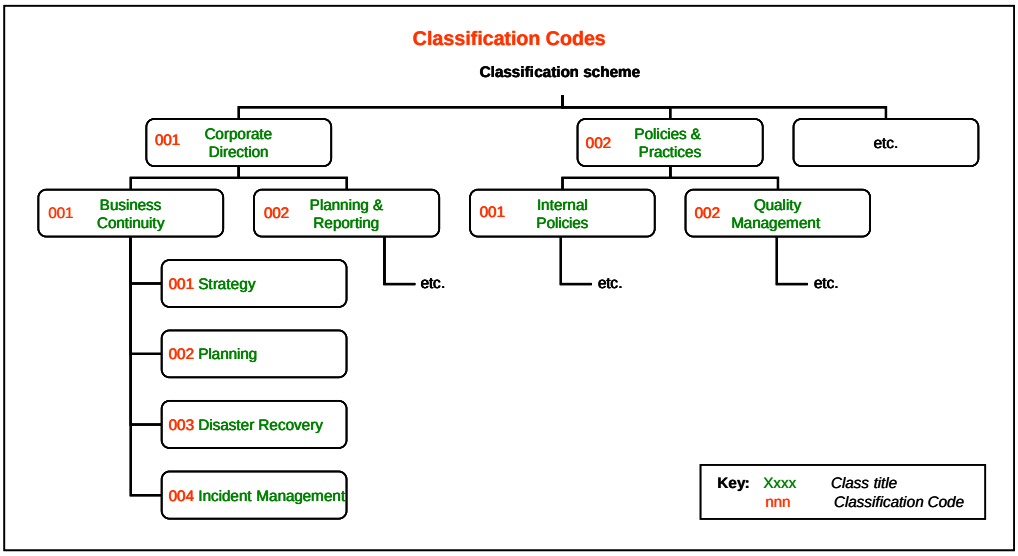

Figure 7.4

The second option is appropriate in case management settings.

Ref	Requirement	Test
7.1.9	When the ERMS generates a new Classification Code automatically (the first option in 7.1.8), it must generate the next sequential number taking into account: ♦ the most recently used Classification Code at that point in the classification scheme, or (for the first at that point) the starting value; ♦ the increment specified, see 7.1.5. *See figure 7.4 for an example.*	Y
7.1.10	When accepting a Classification Code from a user or from an external application, the ERMS must validate it for uniqueness within its parent.	Y

7.2 System Identifiers

Ref	Requirement	Test
7.2.1	Whenever a new occurrence of any of the following is created in the ERMS, the ERMS must associate with it a System Identifier: ♦ classification scheme; ♦ class; ♦ file; ♦ sub-file; ♦ volume; ♦ record; ♦ **redaction**; ♦ retention and disposition schedule; ♦ document.	Y
7.2.2	The ERMS must ensure that all System Identifiers are unique within a classification scheme hierarchy and within the ERMS instance. *Note that this requirement extends across geographical locations where a distributed classification scheme has been implemented and across classification schemes when more than one classification scheme has been implemented.*	N
7.2.3	The ERMS must be able to store System Identifiers as metadata elements of the entities to which they refer.	Y

Ref	Requirement	Test
7.2.4	The ERMS should allocate System Identifiers which are globally unique. *Globally unique means that the System Identifiers are allocated using an algorithm that guarantees no other System Identifier can have the same value, regardless of when it is produced or by which ERMS.* *This is desirable to allow for re-configurations, such as those caused by corporate re-organisations, acquisitions and mergers etc. If every entity is not allocated a globally unique System Identifier, the probability of difficulties during re-configurations is high.*	N
7.2.5	The ERMS should use the UUID algorithm (as specified in ISO/IEC 9834-8 and ITU-T Rec. X.667) to generate globally unique System Identifiers. *This algorithm, which in some implementations is commonly referred to as GUID (Globally Unique ID), can be used to guarantee uniqueness.* *Other approaches to the generation of unique identifiers may be used, including the Digital Object Identifier System (DOI®),the Uniform Resource Name (URN) scheme and the Archival Resource Key (ARK).*	P
7.2.6	The ERMS must not require users to enter or use System Identifiers for any ERMS function. *This requirement is included because globally unique identifiers tend to be long and not "user-friendly". However, it is acceptable for users to be allowed to use System Identifiers if they choose to.*	P

8. SEARCHING, RETRIEVAL AND PRESENTATION

This chapter lists requirements for searching and retrieval in section 8.1. Requirements associated with presentation are divided into three sections: section 8.2 lists requirements for display, section 8.3 deals with printing, and section 8.4 addresses the presentation of records which cannot be printed.

An integral feature of an ERMS is the ability for the user to retrieve files and records. This includes searching for them, whether or not precise details are known, and presenting them. Presentation is producing a representation on-screen ("displaying") or printing; it may also involve, as necessary, playing audio and/or video (see glossary).

Accessing files and records, and then viewing them, requires a flexible and broad range of searching, retrieval and presentation functions to meet the demands of different types of user. Although some advanced search features can be thought of as being beyond classical records management functions, the required functionality is described here on the grounds that an ERMS without good retrieval facilities is of limited value.

All of the features and functionality in this chapter must be subject to access controls as described elsewhere in this specification, including security controls. The ERMS must never present information to any user which that user is not entitled to receive. To avoid complexity, this is assumed and is not repeated in each detailed requirement.

8.1 Search and Retrieval

Searching is the process of identification of records or files through user-defined parameters for the purpose of locating, accessing and retrieving records, classes, files, sub-files, volumes and/or their metadata.

The ERMS search and navigation tools are used to locate metadata, classes, files, sub-files, volumes or records. These require a variety of searching techniques to support users ranging from (for example) the sophisticated "research" user to the "casual" and less "computer literate".

Ref	*Requirement*	*Test*
8.1.1	No ERMS search or retrieval function must ever reveal to a user any information (metadata or record content) where the access and security controls (sections 4.1 and 10.13 respectively) prevent access by that user.	P
8.1.2	The ERMS must allow users to search for and retrieve: ♦ records; ♦ every level of aggregation of records (class, file, sub-file, volume); and their associated metadata at any level of the classification scheme.	Y
8.1.3	The ERMS must allow users to specify any combination of metadata elements as search terms. *The search facility needs to be able to search on any of the metadata elements, for example, Title.*	Y

Ref	Requirement	Test
8.1.4	The ERMS must allow users to specify whether a search is to find records or a specified level of aggregation of records.	Y
8.1.5	The ERMS search function should appear to users to be the same for all searches specified in requirement 8.1.2. *In other words, users should see the same interface, features and options whether searching for classes, files, sub-files, volumes or records (though details of the presentation of results may vary according to what is being searched).*	Y
8.1.6	The ERMS must allow users to search for the text content of records. *This includes the text of records that are inherently textual in nature, such as e-mail messages, and (where the ERMS includes OCR functionality) records which have been converted to text by OCR (see requirement 6.5.7).*	Y
8.1.7	The ERMS must allow the use of searches to locate an aggregation for the purpose of declaration, as a part of the declaration process. *This is an ease of use requirement. It requires that search functionality be readily available to users who are in the process of capturing one or more records; in other words users must not be forced to quit a capture process to initiate a search.*	Y
8.1.8	The ERMS must allow users to use any combination of metadata elements and/or textual record content as search terms during a search operation. *For example a search could combine a named author together with a particular text string in the record.*	Y
8.1.9	The ERMS should provide a search function which operates in an integrated and consistent manner across both record content and metadata. *This means that the interface and its behaviour should be the same across these kinds of searches.*	Y
8.1.10	The ERMS must display the total number of items found as a result of the search, and display (or allow the user to request display of) the search results (the “hit list”).	Y
8.1.11	The ERMS should allow users to refine (i.e. narrow) a search without having to re-enter the search criteria. *A user should for example be able to start with the hit list from a search, and then perform a further search within that list.*	Y

Ref	Requirement	Test
8.1.12	The ERMS must allow administrative roles to configure and subsequently change the specification of default search metadata elements including: ♦ any element of record, volume, sub-file, file and class metadata, and ♦ optionally, text. *This refers to the default window that first appears when a search is initiated; it generally contains a set of fields for metadata elements that are commonly used in searches. This set comprises the default elements in the requirement.*	Y
8.1.13	The ERMS must provide a search function that allows the use of all Boolean operators namely: ♦ AND; ♦ OR; ♦ EXCLUSIVE OR; ♦ NOT; in any valid combination to combine an unlimited number of search terms.	P
8.1.14	The ERMS must allow users to search for objects by their keyword(s), where the objects have keywords.	Y
8.1.15	During any search involving keywords, the ERMS must allow users to select keywords from controlled vocabularies (or lists of permitted terms). *Noting requirement 8.1.7, this could be during a capture process, or during any other search.*	Y
8.1.16	The ERMS should incorporate the use of a thesaurus to enable users to search by concept.	Y
8.1.17	Where the ERMS incorporates the use of a thesaurus for concept searching, it should be compliant with at least one of the following standards: ♦ ISO 2788; ♦ ISO 5964. *This will allow retrieval of documents with a broader, narrower, or related term in their content or metadata. For example, a search for "ophthalmic services" might retrieve "health services", "eye test" or "ophthalmology".* *The former standard specifies a monolingual thesaurus and the latter a multilingual thesaurus. (See 3.2.13 and 3.2.14).*	Y

Ref	Requirement	Test
8.1.18	If a thesaurus compliant with ISO 2788 or ISO 5964 is integrated with the ERMS, the ERMS should allow a user who is searching using a keyword (or other metadata element related to the thesaurus) to use the full features of the thesaurus, such as broader, narrower and related terms and synonyms as an integrated part of the process. *In other words, if a user is searching for a file, the user may enter a term that is not in the scheme's controlled vocabulary, then use the thesaurus features to find the appropriate preferred keyword. An example is if "budgets" is a preferred keyword: in this case, a user might enter "estimates" and then be guided to its broader term "budgets"; or a user might enter "accounting records" and be presented with a list of narrower terms, one of which is "budgets".* *For ease of use, users must not have to leave the search interface to access the thesaurus to search for related search terms. Refer to the introduction to section 11.8 for a more detailed explanation of the phrase "as an integrated part of the process".*	Y
8.1.19	Where the ERMS incorporates the use of a thesaurus, the ERMS must allow an administrative role to maintain the thesaurus. *Maintenance is needed for the introduction of new terms and terms specific to the business.*	Y
8.1.20	The ERMS must restrict to authorised administrative roles the ability to change the keywords associated with a file. *This facility is intended for exceptional circumstances only, such as to correct clerical errors. Changing keywords inappropriately can seriously compromise the accessibility of records, even if logged in an audit trail, and so should be avoided.*	Y
8.1.21	The ERMS should provide for partial match and "wild card" searching that allows for forward, backward and embedded expansion, for both metadata values and content. *For example:* ♦ *the search term "proj*" might retrieve records containing "project" and "projection" and "PROJA";* ♦ *the search term "psycho*s" might retrieve records containing "psychosis", "psychotics" and "psychologists";* ♦ *the search term "*byte" might return "gigabyte" and "terabyte";* ♦ *the search term "organi?ation" might retrieve records containing "organisation" and "organization".*	Y

Ref	Requirement	Test
8.1.22	The ERMS should provide word proximity searching. *A proximity search finds terms separated by no more than a specified number of words, for example:* ♦ *"International" and "Organisation" separated by no more than one word.*	Y
8.1.23	The ERMS must allow users to limit the scope of any search to any aggregation specified by the user at the time of the search.	Y
8.1.24	The ERMS must be able to search for, and retrieve, a complete electronic file, sub-file or volume, and all its contents and contextual metadata, and display a list of all the, and only those, entries in the context of that aggregation in a single retrieval process. *This is needed when a user wishes to copy or print the entire contents of a file to take to a meeting, or to facilitate temporary working, or for any other reason.*	Y
8.1.25	The ERMS must behave in an identical manner when searching regardless of whether the objects being searched for are stored on-line, near-line or off-line, save that the mechanism and performance for presenting electronic objects may vary. *This requirement applies only when the ERMS uses near-line and/or off-line storage in addition to online storage.*	P
8.1.26	The ERMS should allow users to save and re-use search terms.	Y
8.1.27	The ERMS should allow users to make saved search terms available for use by other users.	Y
8.1.28	The ERMS should allow users to specify time intervals in search requests, e.g. calendar dates or number of days.	Y

Ref	Requirement	Test
8.1.29	The ERMS should allow the use of time intervals specified either as dates (e.g. 24 Dec 2008 – 5 Jan 2009) or in natural language, e.g. “last week”, “this month”, as search terms, allowing the use of at least the following words and/or their equivalents in other languages: ♦ last; ♦ this; ♦ next; ♦ week; ♦ month; ♦ quarter; ♦ year; ♦ names of days of the week; ♦ names of months.	Y
8.1.30	The ERMS should allow users or administrative roles to configure display of the search results, including: ♦ the order in which the search results are presented; ♦ the number of hits displayed on the screen per view from the search; ♦ the maximum number of hits for a search; ♦ which metadata elements are displayed in search result lists.	Y
8.1.31	The ERMS should provide implicit or explicit relevance ranking of the search results.	Y
8.1.32	When a hit list contains a redaction of an electronic record, or a record for which a redaction exists, (see section 9.3), the ERMS should relate the two, so that retrieval of one shows the existence of and allows retrieval of the other, subject to access controls, whilst retaining separate metadata for the two items.	Y
8.1.33	The ERMS should allow the configuration of a search engine other than the default search engine. *It may be desirable for an organisation to implement a search engine other than that which is supplied with the ERMS for system compatibility or other reasons.*	N

8.2 Presentation: Displaying Records

An ERMS may contain records with different formats. The user requires generic presentation facilities that will accommodate the display of a range of formats.

Ref	*Requirement*	*Test*
8.2.1	Whenever a user reaches a view that indicates the existence of a class, file, sub-file, volume or record, the ERMS must be able to present its contents and/or its metadata by a mouse click or keystroke. *This applies regardless of how the user reached the view – by navigating through the classification scheme, by searching, by following a link or any other way – and assumes that the user has appropriate permissions.* *For example:* ♦ *a user executes a search and obtains a hit list showing several records; for any record the ERMS must be able to present the content of any record in the hit list if the user enters a mouse click or keystroke, and must also be able to present the record's metadata similarly;* ♦ *a user navigates the classification scheme to a class that contains files; the ERMS must be able to present a list of all the files allocated to that class if the user enters a mouse click or keystroke, and must also be able to present the class's metadata similarly .* *If the ERMS is storing records in a proprietary application format, it may be acceptable for the presentation to be performed by an application outside the ERMS.*	Y
8.2.2	The ERMS should be able to present records that the search request has retrieved without loading a software application associated with the record. *This is typically provided by integrating in the ERMS a viewer software package. This is frequently desirable to increase speed of presentation.*	Y
8.2.3	The ERMS should be able to present all the types of electronic records specified by the organisation in a manner that preserves the information of the records (e.g. all the features of visual presentation and layout produced by the generating application package), and which presents all components of an electronic record together. *The organisation needs to specify the application packages and formats required, and in some cases acceptable levels of fidelity. In many cases (e.g. in typical office environments) the fidelity need not be specified in detail; however rigorous specification of fidelity may be needed for applications which rely on detailed interpretations, such as records including high-resolution X-ray images.*	N

8.3 Presentation: Printing

This section applies only to records and other information whose content can be printed in a way that is understandable. It does not apply to, for example, audio or video files.

The ERMS must provide printing facilities, to allow all users to obtain printed copies of printable records, their metadata, and of other administrative information.

In all requirements, "printing" is understood to include features normally associated with report production, such as multi-page reports, page numbering, dated headings, and the use of any configured printer. Sending screen image dumps to a printer is not normally considered sufficient for these requirements.

Ref	*Requirement*	*Test*
8.3.1	The ERMS must be able to print the content of records and specified elements of their metadata.	Y
8.3.2	The ERMS must allow the printing of all or specified metadata for any class, file, sub-file, volume or record.	Y
8.3.3	The ERMS must allow all records in a class, file, sub-file or volume to be printed in one operation.	Y
8.3.4	The ERMS must allow users to specify a subset of metadata elements (such as Title, Author, Creation date) and print out a summary list of these elements for selected aggregations of records.	Y
8.3.5	The ERMS should allow an administrative role to specify at configuration time that all printouts of records' contents have selected metadata elements appended to them (e.g. title, registration number, date, **security category**) by default. *This could be used, for example, to ensure that whenever a record is printed, its security category is printed at the same time, as a security measure.*	Y
8.3.6	The ERMS should allow users, at the time of printing, to amend the default metadata elements that are appended to printouts.	Y
8.3.7	ERMS must allow users to print hit lists (see section 8.1) resulting from a search.	Y
8.3.8	The ERMS must allow an administrative role to print all, or a selection of, administrative parameters. *For example a list of all users with a specific security category or all users in a particular user group.*	Y
8.3.9	The ERMS must allow an administrative role to print retention and disposition schedules.	Y
8.3.10	If a thesaurus is integrated (see 8.1.16) the ERMS should allow administrative roles to print the thesaurus.	Y

Ref	Requirement	Test
8.3.11	The ERMS must be able to print out a list of each controlled vocabulary (a list of all permitted terms). *It is acceptable to print the list from thesaurus management software where this is integrated with the ERMS.*	Y
8.3.12	The ERMS should be able to export a list of each controlled vocabulary (a list of all permitted terms).	Y
8.3.13	Where a controlled vocabulary of keywords takes the form of an ISO 2788-compliant or ISO 5964-compliant thesaurus, the ERMS should be able to print out the thesaurus entries, showing all terms and their relationships. *Printing of ISO standards-based thesauri should be compatible with the representational guidelines given in ISO 2788 and ISO 5964.* *It is acceptable to print this from separate thesaurus management software that is integrated with the ERMS.*	Y
8.3.14	The ERMS must allow authorised roles to print the classification scheme both as a complete scheme and as any class selected from the scheme.	Y
8.3.15	A user printing a classification scheme (as in 8.3.14) should be able to specify the content and format of the resulting printed output. *For example, the user should be able to specify the metadata elements to be printed, and preferably choose a list, or indented, or graphical representation.*	P
8.3.16	The ERMS must allow administrative roles be able to print a list (sometimes referred to as a repertory) of all files or of files classified against a specific class (and its child classes).	Y
8.3.17	A user printing a list of files (as in 8.3.16) should be able to specify the sequence, content and format of the list. *For example, the user should be able to sort in ascending or descending order, on title or code, and preferably on any attribute; and should be able to specify the metadata elements to be printed.*	Y
8.3.18	The ERMS must allow administrative roles to print all or part of audit trails (see 4.2.1).	Y
8.3.19	The ERMS must be able to print the formats specified by the organisation. Printing must: ♦ preserve the layout produced by the generating application package(s); ♦ include all printable components of the electronic record. *The organisation needs to specify the formats required.*	Y

8.4 Presentation: Other

This section applies only to records and other information whose content cannot be printed in a way that is understandable, such as audio or video files.

Ref	*Requirement*	*Text*
8.4.1	The ERMS must include features for presenting and outputting to appropriate media records which cannot be printed. *Examples include audio, video, and some web-sites. The organisation will need to specify the nature of these records.*	P

9. ADMINISTRATIVE FUNCTIONS

This chapter covers the maintenance and system support functionality required by an ERMS.

Requirements are listed in this chapter for:

- general administration (section 9.1);
- system reporting (section 9.2);
- changing, deletion and redaction of records (section 9.3).

Closely-related features are described in chapter 4, namely;

- access permissions in section 4.1;
- backup and restore in section 4.3.

These facilities allow administrative roles to manage change in the user population and parameters affecting the behaviour of the system. The ERMS needs to provide administrative roles with the ability to manage events such as maintaining the user base and, crucially, the permissions assigned to users, groups and roles. The system must also provide monitoring capability for system errors.

Some of these facilities may be provided by an associated EDMS, database management system, operating system, or by other applications.

9.1 General Administration

This section includes requirements for managing system parameters, system management and configuration, and user administration.

In large organisations, the functionality described in this section may be assigned to an operations function rather than to an application administrator. However, in small organisations, it may be assigned to an administrator.

Ref	*Requirement*	*Test*
9.1.1	The ERMS must allow administrative roles to retrieve, display and re-configure systems parameters and settings made at configuration time. *These settings include, for example, configuration of access rights or classification codes.*	Y
9.1.2	The ERMS must allow administrative roles to: ♦ allocate functions to users and roles; ♦ allocate one or more users to any role.	Y

Ref	Requirement	Test
9.1.3	The ERMS must monitor available storage space, and notify administrative roles when action is needed because available storage is below a level set at configuration time, or because of another error condition. *It is acceptable for administrative roles to be notified by means of separate system management software.*	P
9.1.4	Where the storage supports error rate reporting, the ERMS should monitor error rates occurring on storage media, and report to administrative roles any medium or device on which the error rate is exceeding a parameter set at configuration time or at a later date. *This applies particularly to optical media.* *It is acceptable for administrative roles to be notified by means of separate system management software.*	N
9.1.5	The ERMS should allow administrative roles easily to move users between user groups and roles. *In particular, it should be possible to move a user without having to delete the user from the ERMS and re-enter the user's details.*	Y

9.2 Reporting

Flexible reporting is an important feature in an ERMS. It is required so that administrative roles can manage the system; and so that management can monitor the ERMS to ensure that it is used appropriately.

An ERMS needs to be able to provide a number of management, statistical and ad hoc reports so that administrative roles can monitor system activity and status. This reporting is required across the entire system, including:

- the classification scheme;
- files and records;
- user activity;
- access and security permissions;
- disposition activity.

The ERMS must provide a number of standard reports capable of being configured by administrative roles and should be flexible to enable ad hoc reports to be produced on demand.

Ideally the ERMS will include a flexible report-writing sub-system. However, it is not appropriate to attempt to reproduce here the requirements for a comprehensive report writing sub-system, so this section gives outline requirements only. The amount and complexity of reporting will be determined by organisational features including the size, complexity and levels of change to the classification scheme, the amount and nature of the records, and the user base.

Ref	Requirement	Test
9.2.1	The ERMS must allow administrative roles to produce periodic reports (daily, weekly, monthly, quarterly) and to specify ad hoc reports.	Y
9.2.2	The ERMS must include features for printing reports, viewing them on-screen and storing them in electronic form. *As in section 8.3, "printing" is understood to include features normally associated with report production such as multi-page reports, page numbering, dated headings, configurable page headers and footers, and use of any configured printer). Sending screen image dumps to a printer is not normally considered sufficient for these requirements.*	Y
9.2.3	A user viewing an ERMS report should be able to capture it as a record. *This will be useful, for example, for storing securely reports that attest to the integrity of the records.*	Y
9.2.4	The ERMS should allow time periods covered by a report to be configured either as a date range (e.g. 24/12/2008 – 5/1/2008) or as a time interval specified in natural language (as in 8.1.29).	Y
9.2.5	The ERMS must include features for sorting and selecting the information included in reports. *For example, users should be able to specify which columns of a columnar report are used to sort the report contents.*	Y
9.2.6	The ERMS should include features for totalling and summarising report information.	Y
9.2.7	The ERMS should include features for graphical reporting. *For example, trend-reports showing changes in reported information over time, or histograms.*	Y
9.2.8	The ERMS must enable report requests to be saved for future re-use.	Y
9.2.9	The ERMS must enable reports to be exported for use in other applications. *For example, users may wish to work with the contents of a report using spreadsheet software. MoReq2 does not specify the format(s) to be used for such exports.*	Y

Ref	Requirement	Test
9.2.10	The ERMS must be able to provide reports on the total number and location of: ♦ files, sub-files and volumes; ♦ records, sorted by file format and version; ♦ files, sub-files and volumes, sorted by access control and security markings (where used); ♦ electronic files, sub-files and volumes, sorted by size; ♦ electronic files, sub-files and volumes, sorted by storage location; ♦ vital records.	P
9.2.11	The ERMS must be able to provide reports on: ♦ the rate of capture of records; ♦ the rate of retrieval of records; ♦ the rate of creation of new classes and files.	Y
9.2.12	If the document management option described in section 10.3 is present, the ERMS must be able to provide reports on ♦ the total number and location of documents; ♦ the rate of capture/creation of documents; ♦ the rate of retrieval of records.	Y
9.2.13	The ERMS should allow the reports described in 9.2.11 and 9.2.12 to be for any combination of: ♦ across the entire system or for specified classes; ♦ specified user groups or users; ♦ a specified range of dates.	Y
9.2.14	The ERMS should be able to provide reports on actions on files and records sorted by user, by workstation and (where technically appropriate) by network address.	P
9.2.15	The ERMS should allow the reports described in 9.2.11 to cover a specified time interval within several days. *For example, showing hourly figures, to allow peaks and troughs of activity to be monitored.*	Y

Ref	Requirement	Test
9.2.16	The ERMS must be able to produce a report listing files, sub-files and volumes, for all or part of the classification scheme, structured to reflect the classification scheme.	Y
9.2.17	The ERMS must be able to provide a report on the amount of system storage space currently in use and available.	Y
9.2.18	The ERMS must allow administrative roles to produce reports on the audit trail. These reports must include, at a minimum, reporting based on any selected: ♦ class; ♦ file; ♦ sub-file; ♦ volume; ♦ record; ♦ user; ♦ time period.	Y
9.2.19	The ERMS should allow administrative roles to enquire on and produce audit trail reports based on selected: ♦ security categories; ♦ user groups; ♦ other metadata.	Y
9.2.20	The ERMS must be able to report on the outcome of a disposition process listing the classes, files, sub-files, volumes and records successfully disposed of and any failures.	Y
9.2.21	The ERMS must be able to provide reports on the outcome of an export process listing the classes, files, sub-files, volumes and records successfully exported and any failures.	Y
9.2.22	The ERMS must be able to provide administrative roles with reports on disposition activity, including disposition actions that are overdue.	Y
9.2.23	The ERMS should allow administrative roles to restrict users' access to selected reports.	Y

Ref	Requirement	Test
9.2.24	The ERMS must be able to provide administrative roles with a report on attempted access control and other security policy violations. *This requirement only applies when the ERMS (and/or the operating system) is configured so as to allow an item's existence to be visible to a user even though the user is not allowed access to it. It is not relevant when the ERMS is configured to hide the existence of an item which cannot be accessed.*	Y
9.2.25	Administrative roles should be able to specify the frequency of retention and disposition schedule reporting, the information reported and highlighting exceptions such as disposition overdue.	Y
9.2.26	The ERMS should provide quantitative reports on the kinds of records to be reviewed within a specified period.	Y
9.2.27	The ERMS should support reporting and analysis tools for the management of retention and disposition schedules by an administrative role, including the ability to: ♦ list all retention and disposition schedules, sorted by reason or date; ♦ list all entities to which a specified retention and disposition schedule is assigned; ♦ list the retention and disposition schedule(s) applied to all entities in a class; ♦ identify, compare and review retention and disposition schedules (including their contents) across the classification scheme; ♦ identify formal contradictions in retention and disposition schedules across the classification scheme.	Y
9.2.28	The ERMS should be able to accumulate statistics of review decisions in a given period and provide tabular and graphical reports on the activity.	Y
9.2.29	The ERMS should be able to accumulate statistics of the imposition and lifting of disposal holds in a given period and provide tabular and graphical reports on the activity.	P
9.2.30	The ERMS must produce a report detailing any failure during a transfer, export, destruction or deletion. The report must identify any records, aggregations and associated metadata destined for transfer which have generated processing errors, and any entities which are not successfully transferred, exported, destroyed or deleted.	Y
9.2.31	The ERMS must produce a report detailing any failure during an importation. The report must identify any records, aggregations and associated metadata destined for import which have generated processing errors, and any entities which are not successfully imported.	Y

Ref	Requirement	Test
9.2.32	The ERMS should support the import process, by tracking and reporting on its progress and status, including the percentage completed and number of records imported.	Y
9.2.33	The ERMS should provide the ability to sort electronic files selected for transfer into ordered lists according to user-selected metadata elements.	Y
9.2.34	The ERMS should provide the ability to generate user-defined reports to describe electronic files and records that are being exported or transferred.	Y

9.3 Changing, Deleting and Redacting Records

A basic principle of recordkeeping is that records cannot normally be changed, and (except at the end of their life cycle in the ERMS) files, sub-files, volumes and records cannot normally be destroyed.

This section deals with the requirements for exceptional situations where the content of a declared record may need to be amended, or a record deleted and replaced.

In some situations, administrative roles may need to "delete" records to correct errors to meet legal requirements. An example may arise under data protection legislation, though other scenarios are possible.

The action of deletion may mean one of two things:

- destruction;
- retention, accompanied by a notation in the record's metadata that the record is considered removed from records management control.

In either case, deletion is to be exceptional, and so the ability to delete must be tightly controlled in order to protect the general integrity of the records. In particular, information about deletions must be stored in the audit trail.

If local legislation or regulation imposes different requirements, for example relating to the expunging of personal data (see ISO 12037), this should be addressed in a national chapter zero.

Administrative roles sometimes need to publish, or make available, records containing information which is still sensitive, without revealing the sensitive information. This can result from data protection rules, security considerations, commercial risk, etc. For this reason, administrative roles need to be able to mask the sensitive information, without affecting the underlying record.

The process is referred to here as redaction. When this process is carried out, the result is the original record (unchanged), and a copy of the record which has been masked in some way (the **redacted** copy, or redaction of the original record). The ERMS stores both the original record and the redaction.

In principle, redaction can apply to any kind of record – text, image, audio, video etc.

Note that deletion and change are also discussed in chapter 5.

Ref	Requirement	Test
9.3.1	The ERMS must allow a configuration option which prevents any record, once captured, from being deleted or moved by any administrative or user role; see also 9.3.3. *This requirement does not affect transfer or destruction of records in accordance with a retention and disposition schedule, as described in section 5.3. It is intended for environments in which the deletion of records (as described above) is either unnecessary or not permitted. The alternative to this option is specified in 9.3.2.*	Y
9.3.2	The ERMS must allow a configuration option, as an alternative to 9.3.1, that "deletion" of a record is implemented as destruction of that record, and that relocation of a record results in moving the record; see also 9.3.4. *This is not regarded as good practice in records management. It is included here only for situations in which it is considered unavoidable. In most situations, the option specified in 9.3.1 should be preferred. This option and the option in 9.3.1 are mutually exclusive.*	Y
9.3.3	If the option in 9.3.1 is selected, the ERMS must behave as follows: ♦ If an administrative role "deletes" a record (as in 9.3.5) the record's metadata must be marked accordingly, and the ERMS must hide the content and metadata of the record from all users save potentially for suitably-authorised administrative roles, as if it were deleted; and the ERMS must record this in the audit trail. ♦ If an administrative role "re-locates" a record (as in 3.4.1), the ERMS must behave exactly as for a deletion but with the addition that a copy (or a pointer, depending on the storage method used) must be inserted automatically at the new location. *This assumes that either no administrative roles would have such authorisation, or else a particularly small number.*	Y
9.3.4	If the option in 9.3.2 is selected, the ERMS must behave as follows: ♦ If an administrative role deletes a record (as in 9.3.5) the record must be deleted, along with its metadata except for metadata specified as part of its metadata stub (see 5.3.19); and the ERMS must log this in the audit trail. ♦ If an administrative role re-locates a record (as in 3.4.1), the ERMS must behave exactly as for a deletion but with the addition that the record (or a pointer to it, depending on the storage method used) must be inserted automatically at the new location.	Y

Ref	Requirement	Test
9.3.5	The ERMS must allow administrative roles to delete classes, files, sub-files, volumes and records outside the disposition process. *This is intended for use only in exceptional circumstances as described in this section. It must be read together with 9.3.1 and 9.3.2.*	Y
9.3.6	The ERMS must allow user roles to mark classes, files, sub-files, volumes and records as candidates for deletion. *The administrative role can then decide whether or not to carry out the deletion.*	Y
9.3.7	In the event of any deletion as defined above, the ERMS must: ♦ log the deletion in the audit trail; ♦ produce a report for administrative roles; ♦ delete the entire contents of a class, file, sub-file or volume when it is deleted; ♦ ensure that no documents are deleted if their deletion would result in a change to another record (for example if a document forms a part of two records, one of which is being deleted); ♦ highlight to administrative roles any links from another file, or record to a file, sub-file or volume which is about to be deleted, requesting confirmation before completing the deletion; ♦ maintain integrity of the metadata at all times. *In this context the phrase "maintain integrity of the metadata" means to ensure that no metadata in any entity (class, record etc.) refers to an entity that does not exist.*	Y
9.3.8	Administrative roles must be able to change any user-entered metadata element. *This functionality is intended to allow administrative roles to correct user errors such as data input errors, and to maintain user and group accesses. Good practice generally will require that users correct their errors whenever possible; this requirement does not prevent users from doing so.*	Y
9.3.9	Information about all changes to all metadata elements must be stored in the audit trail.	Y
9.3.10	The ERMS must allow administrative roles to create one or more redaction(s) of a record while retaining the original record. *It may be necessary, in some cases, to provide redactions for several parties in which different parts of the record have been redacted.*	Y

Ref	Requirement	Test
9.3.11	The ERMS must allow removal or hiding of sensitive information within a redaction for all record formats required by the organisation. *If the ERMS does not provide these facilities, it must allow for other software packages to integrate with it and do so. It is acceptable for the ERMS to render a record to a different file format to permit the redaction of a copy, provided that the rendition maintains sufficient fidelity.* *It is essential that, when this feature or any other redaction features are used, none of the removed or hidden information can ever be retrieved from the redaction, whether on screen, when printed, played back or in any other form of presentation. This is regardless of the use of any presentation features such as rotation, zooming or any other manipulation including opening the redaction in a different software package.*	P
9.3.12	When a redaction is created, the ERMS must store automatically its creation in the metadata of the redaction and the record, including date, time and creator.	Y
9.3.13	When a redaction is created, the ERMS must require the user creating it to enter a reason, and must store that reason in the metadata of the redaction and the record.	Y
9.3.14	Upon creation of a redaction the ERMS should automatically declare redactions as records, classifying them in the same aggregation as the original record and prompting the creator of the redaction for: ♦ a reason (see 9.3.13); ♦ security category (where applicable); ♦ optionally, an aggregation into which a copy of the redaction will be declared.	P
9.3.15	Upon creation of a redaction the ERMS should allow the copying of metadata elements to the redaction.	Y
9.3.16	Subject to access control rights the ERMS should enable amendment of selected metadata values, for example title.	Y
9.3.17	The ERMS should store a cross reference to a redaction in the same class, file, sub-file or volume as the original record, even if that class, file, sub-file or volume is closed. *This is in addition to the requirement to file a copy, in 9.3.14, to allow for cross referencing even in the same file, as the original and redaction may be separated by large numbers of records in the file.*	Y
9.3.18	When a record is retrieved the ERMS must show, or allow the user to see, the existence of all redactions made from that record and, subject to access and security controls, make them available for retrieval.	Y

Ref	Requirement	Test
9.3.19	When a redaction is retrieved the ERMS must show, or allow the user to see, the existence of the original record and, subject to access and security controls, make it available for retrieval.	Y
9.3.20	The ERMS must store in the audit trail any change made as a result of any requirement in this section.	P

10. OPTIONAL MODULES

This chapter contains requirements functionality closely allied to electronic records management. It covers requirements to support the management of physical (non-electronic) records, document management, workflow, electronic signatures and other functionality.

Each of the sections in this chapter corresponds to one optional module of the MoReq2 Testing Framework. These modules are optional in the sense that their requirements are not a mandatory part of the core functionality of a MoReq2 compliant ERMS.

The sections in this chapter list requirements for the following areas:

- management of physical records (section 10.1);
- disposition of physical records (section 10.2);
- document management and collaborative working (section 10.3);
- workflow (section 10.4);
- casework (section 10.5);
- integration with content systems (section 10.6);
- electronic signatures (section 10.7);
- encryption (section 10.8);
- digital rights management (section 10.9);
- distributed systems (section 10.10);
- offline and remote working (section 10.11);
- fax integration (section 10.12);
- security categories (section 10.13).

The requirements in this chapter are for optional functionality which may be integrated with an ERMS. They supplement the core requirements in the rest of MoReq2. These requirements are applicable only if the organisation needs to implement the optional functionality.

Conformity with the requirements in this chapter is not required for MoReq2 compliance. Therefore mandatory requirements in this chapter are mandatory only when the optional module in which they are located is included in a test.

In each case, requirements are presented at a high level. As they do not define the core functions of an ERMS, these requirements are not exhaustive but rather provide an indication of the appropriate activities.

10.1 Management of Physical (Non-electronic) Files and Records

In addition to the electronic records, an organisation's records repository may contain non-electronic records. These can include paper-based records and records on other analogue media, for example microfiche or audio tapes. They may also include digital records stored on portable media, such as CDs, DVDs and computer tapes.

The term physical records is used in MoReq2 to mean any record that is held in a medium outside the ERMS. This includes not only analogue media but also digital media holding records that are not individually controlled by the ERMS. For example:

- a CD-ROM containing 10,000 images which are not individually recognised by the ERMS as records is a physical record;

- a CD-ROM containing 10,000 images, and located in a drive or jukebox connected to the ERMS, and with each of the images recognised by the ERMS as a record is not a physical record – it is a removable medium on which electronic records are stored.

This specification does not address the business need to manage and maintain physical records. Such a need may or may not exist, according to the legislative and regulatory environment. Where it does exist, care needs to be taken to preserve the integrity and accessibility of electronic and physical records taken as a whole. These issues should be addressed by appropriate organisational policies.

The ERMS must be able to accommodate references to physical records as well as, and together with, electronic records; and to manage aggregations made up of both electronic and physical records. Classes, files, sub-files and volumes may all contain any combination of electronic records and physical records. This differs from the entity-relationship model in the previous version of MoReq.

Physical records can co-exist with electronic records in several scenarios. The scenarios include:

- A class, file, sub-file or volume contains only physical records. In this case, the entity represented in the ERMS represents a physical container for the records, such as a filing jacket;

- A class, file, sub-file or volume contains both electronic and physical records. The physical records are stored without a container relevant to records management – for example, an engineering drawing stored along with unrelated drawings in a cabinet.

The ERMS must provide features to allow physical containers (as in the first option) to be managed.

In order to manage physical records the ERMS must be able to capture and manage metadata about them. This metadata enables administrative and user roles to, subject to access controls, locate, track, retrieve, review and dispose of physical records, and to allocate access controls to them in the same way as to electronic records.

Similarly, the ERMS must be able to capture and manage metadata about physical containers.

Ref	*Requirement*	*Test*
10.1.1	The ERMS must allow an administrative role to identify classes, files, sub-files and volumes that exist as physical containers.	Y

Ref	Requirement	Test
10.1.2	The ERMS must allow administrative and user roles to enter and maintain metadata about classes, files, sub-files and volumes that exist as physical containers, as specified in the MoReq2 metadata model.	Y
10.1.3	The ERMS must allow user roles to enter and maintain information about physical records in classes, files, sub-files and volumes, following the same rules as when capturing electronic records.	Y
10.1.4	The ERMS must allow classes, files, sub-files and volumes to contain electronic records and physical records together, in any combination.	Y
10.1.5	The ERMS must allow physical records to be managed in the same way as the electronic records, including any inheritance of metadata.	P
10.1.6	When a user is browsing, retrieving, or otherwise working with a class, file, sub-file or volume, the ERMS should indicate the presence of any physical container or records in it with appropriate indicators. *A user needs to determine easily whether physical entities exist in order to ensure that all records are managed in the same manner. MoReq2 does not prescribe the nature of these indicators.*	Y
10.1.7	The ERMS must allow a different set of metadata elements to be configured by an administrative role for physical classes, files, sub-files, volumes and records than for the electronic equivalents. As an example, physical file metadata could include (but is not limited to) additional metadata for: ♦ information on its physical location; ♦ information regarding the format of the physical container or record.	Y
10.1.8	The ERMS must ensure that retrieval of any class, file, sub-file or volume simultaneously retrieves the metadata for both electronic and physical entities associated with it in a single operation.	Y
10.1.9	The ERMS should support tracking of physical containers and records by the provision of check-out and check-in to log their location, **custodian** and the date of check-in/check-out.	Y
10.1.10	The ERMS should allow the user checking out a physical aggregation or record to specify a date by which it is due to be returned.	Y
10.1.11	The ERMS should report to a specified user when the date due for return of a physical aggregation or record is approaching and when it is overdue.	Y
10.1.12	The ERMS should allow a suitably authorised user to change the date due for return of one or several physical aggregations or records, in a single operation.	Y

Ref	Requirement	Test
10.1.13	The ERMS must ensure that the metadata for physical aggregations and records is always subject to the same access controls as would be the case if they were purely electronic.	Y
10.1.14	The ERMS should provide a tracking function to allow users to log information about the location and movement of physical aggregations and records.	Y
10.1.15	The ERMS tracking function should allow for locations of physical entities to be selected from or validated against a list (such as a pull-down list). *Where the ERMS does not support a list of locations, non-validated free text is acceptable.*	Y
10.1.16	The ERMS tracking function must allow users to enter the checking out and checking in of physical entities. *In other words, the ERMS must provide facilities to log whether a physical entity is in its home location or has been checked out.*	Y
10.1.17	The ERMS tracking function must log information about the movements of a physical entity which includes: ♦ unique identifier; ♦ current location; ♦ an administrative role-defined number of previous locations (the number to be defined at configuration time); ♦ date moved from location; ♦ date received at location; ♦ user responsible for the move (where appropriate).	Y
10.1.18	The ERMS must allow a user role to see the current location of a checked-out physical entity, its custodian, and the date upon which the check out occurred, subject to access control rights.	Y
10.1.19	The ERMS must log all check in and check out activities and dates within the audit trail.	Y
10.1.20	The ERMS must be able to log in the audit trail all changes made to the metadata values of physical entities. *For example the location metadata element.*	Y

Ref	Requirement	Test
10.1.21	The ERMS should support the printing and recognition of bar codes for files, sub-files, volumes and records; or alternative tracking systems such as Radio Frequency Identification (RFID) technology. *This enables the ERMS to track the location and movements of physical records.*	Y
10.1.22	The ERMS should support the printing of labels for physical files, sub-files and volumes. *This enables a label to be produced containing essential metadata which can then be attached to the physical entity. This could include, but is not limited to, such metadata as:* ♦ *Title;* ♦ *Identifier – System;* ♦ *Classification Code;* ♦ *Date of Opening;* ♦ *Security Category (if used);* ♦ *Normal storage location.*	Y
10.1.23	The ERMS must behave in an identical manner when dealing with physical or electronic records in searches, save that: ♦ the content of physical records cannot be presented (instead, the ERMS displays its location metadata, see below); ♦ different metadata may be shown for physical and electronic records.	Y
10.1.24	The ERMS should be able to notify administrative roles of any events in the retention and disposition schedule relating to non-electronic records and aggregations scheduled since a restore was executed. *Section 4.3 Backup and Recovery sets out the requirements for restoring an ERMS. When the system is used for managing non-electronic records a disparity may arise following a restore whereby disposition actions have been carried out on the physical objects, but this is not shown in the ERMS. This requirement enables administrative roles to apply remedial action.*	Y

10.2 Disposition of Physical Records

Ref	*Requirement*	*Test*
10.2.1	When the retention period for a retention and disposition schedule ends, if that retention and disposition schedule applies to any physical entities the ERMS must notify an administrative role.	Y
10.2.2	The ERMS must alert an administrative role to the existence and location of any physical entity associated with any class, file, sub-file or volume that is to be transferred, exported or destroyed. *This may be either when the retention period for a retention and disposition schedule ends, or when a transfer or export is initiated.*	Y
10.2.3	Whenever any physical entities are exported or transferred, the ERMS must export or transfer the metadata for them in the same way as the metadata for the electronic entities.	Y
10.2.4	On transfer, export or destruction of physical entities the ERMS must require an administrative role to confirm the physical transfer, export or destruction before the transfer, export or destruction is completed. *This normally will require an administrative role to enter manually a confirmation that the physical records have been transferred or destroyed.*	Y

10.3 Document Management and Collaborative Working

Electronic Document Management Systems – EDMSs – are widely used in organisations to provide management and control over electronic documents. Many EDMS functions and facilities overlap with ERMS. EDMSs typically include indexing of documents, storage management, version control, close integration with desktop applications and retrieval tools to access the documents. Some ERMSs provide full EDMS capability, others only provide a subset. Conversely some EDMSs have incorporated core record management functions.

EDMSs often form part of a wider system implementation and contain collaborative working tools to enable a number of users to participate in document drafting.

Collaborative working is also an integral element of Content Management Systems. See section 10.6 for further requirements regarding these features.

By way of clarification, the following table shows typical differentiators between an EDMS and an ERMS.

An EDMS…	**An ERMS…**
♦ allows documents to be modified;	♦ prevents records from being modified;
♦ allows documents to exist in several versions;	♦ allows a single final version of a record to exist;
♦ may allow documents to be deleted by their owners;	♦ prevents records from being deleted except in certain strictly controlled circumstances;
♦ may include some retention controls;	♦ must include rigorous retention

An EDMS…	An ERMS…
	controls;
♦ may include a document storage structure, which may be under the control of users;	♦ must include a rigorous record arrangement structure (the classification scheme) which is maintained by an administrative role;
♦ is intended primarily to support day-to-day use of documents for ongoing business.	♦ may support day-to-day working, but is primarily intended to provide a secure repository for business records.

The rest of this section sets out key requirements to be considered in the provision of an integrated ERMS/EDMS solution. The requirements apply only where EDMS facilities are part of the ERMS. A central feature of these requirements is the concept that documents can be stored in (that is, classified to) the same classes and files as records, though this is optional. This allows draft documents to be filed in the same aggregations as the final versions, which will be records.

Note that the word 'document' is used here specifically to describe information or an object that has not been declared as a record in the ERMS.

Ref	*Requirement*	*Test*
10.3.1	The ERMS should be able to manage electronic documents and records in the context of the same classification scheme, using the same access control mechanisms. *The intention of this requirement is to allow users to store documents that are drafts in the aggregations that the eventual record will be classified to. This is optional.*	Y
10.3.2	Where the ERMS manages both documents and records within the same classification scheme it must clearly indicate which items are documents and which are records. *MoReq2 does not specify how this is achieved.*	Y
10.3.3	Where the ERMS manages both documents and records within the same classification scheme it must allow user roles to perform the following tasks for any specified class or file: ♦ declare all documents as records; ♦ delete all documents, leaving only the records; ♦ delete all documents that are older than a specified age.	Y

Ref	Requirement	Test
10.3.4	Where the ERMS manages both documents and records within the same classification scheme it must notify an administrative role if documents exist within a class or file being exported and provide options to: ♦ enable the documents to be deleted; ♦ declare them as records; ♦ export them with the records.	Y
10.3.5	Where an EDMS is part of an ERMS, or is tightly integrated with an ERMS, the EDMS must be able to pass automatically electronic documents arising in the course of business to the ERMS for automatic capture as records. *This is especially relevant to case working scenarios – see also section 10.5.*	P
10.3.6	The ERMS must allow users to: ♦ capture an electronic document and declare it as a record in one process; or ♦ capture an electronic document, store it, and complete the capture by declaring it as a record at a later time.	Y
10.3.7	The ERMS must be able to copy the contents of an electronic record, in order to create a new and separate electronic document without automatically creating a new record, while ensuring retention of the intact original record. *For example, a user may copy a record in order to send a copy to a recipient who is not a user of the ERMS. This copy may or may not be declared as a fresh record according to the context.*	Y
10.3.8	The ERMS must allow user roles to check out (see 10.3.11) any document to which they have appropriate access rights.	Y
10.3.9	The ERMS must allow user roles to check in any document that they have checked out, giving the user the option of checking it in as a new version or not (see 10.3.20).	Y
10.3.10	The ERMS should allow a user who checks in a document to enter, optionally, a textual explanation of the changes made while it was checked out.	Y

Ref	Requirement	Test
10.3.11	When a document is checked out by a user, the ERMS must prevent any other user from checking it out or changing it (subject to 10.3.13). *When a document is checked out, only the user who has checked it out can edit it.* *This applies to documents only. As a matter of definition, the ERMS must not allow any record to be checked out and amended.*	Y
10.3.12	When a document is checked out, if any other user attempts to check it out, the ERMS must prevent the user from checking it out a second time, must inform the user that it is checked out, and must either: ♦ show the identity of the user who performed the checkout; or ♦ conceal the identity of the user who performed the checkout; the option being specified at configuration time.	Y
10.3.13	The ERMS must allow an administrative role to cancel the check out of a document. *This is intended to allow for situations where the user who checked out the document is unable to check it back in. This situation can arise for several reasons, for example:* ♦ *the user checked it out to a PC that has failed or has been stolen;* ♦ *the checked out document has become corrupted;* ♦ *the user has forgotten to check it back in before starting a period of leave.*	Y
10.3.14	A user must not be able to check in a version of a document that has had its check out cancelled (as in 10.3.13) as the same document.	Y
10.3.15	If an attempt is made to close an aggregation within the ERMS that includes a checked-out document, it must report this as an exception to an administrative role.	Y
10.3.16	Users should be able to capture a document from within the EDMS.	Y
10.3.17	Users must be able to transfer smoothly to and from the ERMS to declare the document as a record from within the EDMS. *This requirement is especially important where the EDMS/ERMS is used in a general office environment.*	N

Ref	Requirement	Test
10.3.18	Where there are multiple versions of a document the ERMS must be able to capture the document as a record in all of the following ways, with one being selected as default at configuration time and the user being able to select one during capture: ♦ the most recent version; ♦ one version that is specified by the user; ♦ all versions stored, held as a single record; ♦ all versions stored, held as separate but linked records.	Y
10.3.19	The ERMS must maintain a version number for each document, and must make it clearly visible when the document is retrieved or searched for.	Y
10.3.20	The ERMS must automatically increment the document version number when a document is checked in as a new version.	Y
10.3.21	The ERMS should allow the version numbering scheme to be defined at configuration time, allowing at least the following options; ♦ simple sequential version numbering, that is numbers of the form 1, 2, 3; ♦ major and minor version numbering, that is that is numbers of the form x.y., where x is a major version and y a minor version, with the user deciding whether to increment the major or the minor version, and the minor version being reset automatically to 0 when the major version is incremented. *Other numbering schemes are acceptable.*	Y
10.3.22	The ERMS must allow document version storage to be configurable by an administrative role, at configuration time or later, at class and file level within the classification scheme, with at least the following default options for each class and file: ♦ all versions of all documents are stored in the class or file; ♦ only the most recent version (where an administrative role has the ability to specify major or minor versions) of each document is stored in the class or file; ♦ a number of versions of each document are stored in the class or file, the number being specified by an administrative role. *This is to enable version control to be used where a history of document development is required. In areas where this history is not required, the number of versions stored – and hence the storage required – can be reduced.*	Y

Ref	Requirement	Test
10.3.23	The ERMS should allow users who are storing a document to override the default value for the number of versions (as defined by 10.3.22) to be stored for that document. *For example, the time of creation and author of a document, also metadata identifiable from structured fields within documents if these exist, such as date and subject.*	Y
10.3.24	The ERMS must allow a user to enter metadata values for a record at the time of capture.	Y
10.3.25	The ERMS must ensure that any metadata that is captured is managed in accordance with the MoReq2 metadata model.	Y
10.3.26	The ERMS should allow an authorised user to map EDMS metadata elements to appropriate ERMS metadata fields.	N
10.3.27	Where there is any conflict in the metadata between the ERMS and the document-generating system, the ERMS must alert the user. *This can arise when the ERMS does not have control over the metadata elements in the document.*	Y
10.3.28	The ERMS should be capable of integration with new EDMS versions or systems as these are brought into use by the organisation. *MoReq2 does not specify how this is achieved. Organisations should consider specifying this capability in more detail.*	N
10.3.29	The ERMS must be capable of version control, that is, managing different versions of an electronic document as a single entity. *This supports the drafting process of a document and enables collaborative working*	Y
10.3.30	The ERMS should be able to restrict users to viewing: ♦ only the latest version of a document; ♦ selected versions of a document; ♦ all versions of a document; ♦ versions that have been captured or registered as records, the choice to be made at configuration or a later time by an administrative role.	Y

Ref	Requirement	Test
10.3.31	The ERMS should allow users to have a"personal" workspace for documents. *This can be used by users to store personal documents which are not expected to be captured as records, for example, early drafts which are not suitable for corporate access, or other documents. Use of this workspace should be optional (that is, it should be possible to configure the ERMS so that it is not available).*	Y
10.3.32	Where the ERMS includes personal workspace, an administrative role must be able to limit the size of this on a per user basis.	Y
10.3.33	Where the ERMS includes personal workspace, access of this must be restricted to the owner.	Y

10.4 Workflow

The Workflow Management Coalition (WfMC) – an international association for developing workflow standards – defines workflow as "the automation of a business process, in whole or part, during which documents, information or tasks are passed from one participant to another for action, according to a set of procedural rules." In this definition, a "participant" can be a user, a work group (for instance a team), or a software application.

The requirements in this section cover both basic routing functions, as described in 6.1.35, and more sophisticated workflow facilities including handling high volume transactions with exception cases, and reporting on system and individual performance. The latter may be provided by integrating a third party workflow product with the ERMS.

Workflow technologies transfer electronic objects between participants under the automated control of a program. In the context of an ERMS, workflow is used to move electronic files and/or documents and records between users, departments and application programs. It is commonly used for:

- managing critical processes such as registration and disposition procedures of files or records;
- checking and approval of records before registration;
- routing records or files in a controlled way from user to user for specific actions, for instance check document, approve new version;
- notifying users of the availability of records;
- distribution of records;
- managing records through case work processes.

Ref	Requirement	Test
10.4.1	The ERMS must allow workflows which consist of a number of procedural steps, each step being (for example) movement of a document, record or file from one participant to another for action or decision.	Y
10.4.2	The ERMS must recognise as “participants” both users and work groups.	Y
10.4.3	Where the participant is a work group, the ERMS workflow feature should include a facility to distribute incoming items to group members in rotation, or on a member’s completion of the current task, to balance team members’ workloads.	Y
10.4.4	The ERMS must allow pre-programmed workflows to be defined by administrative roles.	Y
10.4.5	The ERMS must allow administrative roles to save workflows for future use. *This implies that each saved workflow is assigned a unique identifier.*	Y
10.4.6	The ERMS should allow the administrative role storing the workflow to assign a unique textual title to it.	Y
10.4.7	The ERMS must restrict amendment of pre-programmed workflows to administrative roles, or authorised users.	Y
10.4.8	Whenever an administrative role changes and stores a workflow, the ERMS should store a copy of the workflow before the changes as a record, and should automatically assign a new version number to the changed workflow, with metadata specifying the date/time interval during which it was in effect.	Y
10.4.9	The ERMS must not limit the number of workflows which can be defined and stored.	P
10.4.10	The ERMS must log all creation of, and changes to, pre-programmed workflows in the audit trail.	Y
10.4.11	The ERMS should allow user roles to define, use and save immediately new, user-defined, workflows (sometimes called ad hoc workflows).	Y
10.4.12	The ERMS should include a graphical interface to enable administrative and user roles to define, maintain and edit workflows.	Y
10.4.13	The ERMS should support the disposition, review and export/transfer process, by tracking and reporting on: ♦ progress/status of the review, such as awaiting or in-progress, details of reviewer and date; ♦ records awaiting disposition as a result of a review decision; ♦ progress of the transfer process.	Y

Ref	Requirement	Test
10.4.14	The ERMS must notify an administrative role if a record or file within a workflow is scheduled for review or disposition.	Y
10.4.15	The ERMS must ensure that all records and files retain any links during a workflow process.	P
10.4.16	The ERMS should manage the files and records in queues which can be examined and controlled by administrative roles.	Y
10.4.17	The ERMS must allow user roles to initiate and use workflows defined by administrative roles.	Y
10.4.18	The ERMS must allow users to monitor the progress of workflows they initiate and in which they are participants.	Y
10.4.19	The ERMS should allow the automatic declaration of a document to be a step in a workflow.	Y
10.4.20	The ERMS should not limit the number of steps in each workflow.	P
10.4.21	The ERMS should be able to prioritise items in queues.	Y
10.4.22	The ERMS should include "**rendezvous**" processing. *This requires the workflow to be paused to await the arrival of a related electronic document or record. When the awaited item is received, the flow resumes automatically.*	Y
10.4.23	The ERMS must support the definition of distinct workflow roles to different users. *Examples of these roles include:* ♦ *a workflow administrative role (having permissions to reassign tasks or actions to another user or workgroup);* ♦ *a supervisor role (having permissions to designate a workflow for exception handling in a specific case);* ♦ *ordinary workflow users or workgroups.* *These workflow roles are distinct from the ERMS roles set out in section 13.4.*	Y
10.4.24	The ERMS should enable an administrative role to define the maximum number of steps in a workflow at configuration time.	Y
10.4.25	The ERMS should allow the administrative role defining a workflow to associate time limits with individual steps, and report items which are overdue according to these limits to a nominated user or administrative role.	Y

Ref	Requirement	Test
10.4.26	The ERMS should allow the administrative role defining a workflow to choose from a pre-defined list which actions shall be taken by the participants of the workflow.	Y
10.4.27	The ERMS should allow the administrative role defining a workflow to choose the participants: ♦ by name; ♦ by roles; ♦ by organisational units.	Y
10.4.28	Administrative roles should be able to allocate permissions to individual users so that they are able to reassign tasks/actions in a workflow to a different user or group. *A user may wish to send a file or record to another user because of the record content, because the assigned user is on leave, or for other reasons.*	Y
10.4.29	The ERMS should enable participants to view queues of work addressed to them and either should: ♦ allow the participants to select items for action; or ♦ present items for attention on a first-in-first-out basis; the option to be specified when the workflow is designed.	Y
10.4.30	The ERMS should provide conditional flows that depend on user input or system data to determine the direction of the flow. *In other words, flows which take the record or file to one of a number of participants depending on a condition decided by one of the participants. For example, a flow may take a record to either a credit control participant or an order consolidation section, depending on input from a sales supervisor; or the flow may depend on the value of an order, as computed by the system.*	Y
10.4.31	The ERMS should allow users to suspend a flow temporarily in order to be able to attend to other work, and to resume it later (including after logging off from the system).	Y
10.4.32	The ERMS must notify a user participant when a file or record(s) has been received in the user's electronic "in tray" for attention. *MoReq2 does not specify whether this in tray is the participant's e-mail account in tray, or separate from it.*	Y

Ref	Requirement	Test
10.4.33	The ERMS should support tracking of files and records by the provision of bring forward (also referred to as “tickler”) facilities which enable a user to request a reminder to access the file or record on a future date.	Y
10.4.34	The ERMS must provide a mechanism to allow users to notify other users of records requiring their attention. *This may use an existing e-mail system or a standalone or proprietary messaging system.*	Y
10.4.35	The ERMS should include the ability to trigger an instance of a specified workflow automatically when a record of a specified record type is received. *For example, a loan application workflow can be triggered automatically by the receipt of a record with record type “loan application form”.*	Y
10.4.36	The ERMS should allow the receipt, in specified folders, of electronic documents or records to trigger workflows automatically (the workflow being determined by the document type or other metadata value).	Y
10.4.37	The ERMS must provide comprehensive reporting facilities to allow authorised user and administrative roles to monitor quantities, performance and exceptions.	Y
10.4.38	The ERMS should support the capture of a workflow process as a record.	Y
10.4.39	When file(s) or record(s) have been processed using one or more workflows, the ERMS must allow users to determine the identifier(s) and the version(s) of the workflow(s) used.	Y
10.4.40	The ERMS must ensure that all access controls are maintained at all times. *In other words, it must not be possible to configure any workflow to grant any access to any user that the user would not otherwise have.*	P
10.4.41	The ERMS should be compatible with the Workflow Management Coalition (WfMC) Reference Model.	Y
10.4.42	The ERMS should support the export of a standard workflow process or any of its constituent parts according to any standard XML schema(s).	N
10.4.43	The workflow audit trail should be integrated with the ERMS audit trail.	Y
10.4.44	The workflow audit trail must be unalterable.	Y

10.5 Casework

This section specifies requirements for the handling of “case files” in a MoReq2-compliant ERMS. See the glossary for a definition and explanation of case files.

The term “case file” is defined in the MoReq2 glossary as a file relating to one or more transactions performed totally or partly in a structured way. In this context, “structured” means that the transactions follow rules that are (or that could be) documented, that they follow a consistent process (they do not allow for users to invent completely new parts of the process), and that they are repeated across many instances of similar transactions. The contents of the records in a case file may be structured (for instance completed online forms) or unstructured (for instance e-mail messages or scanned images of paper forms), in any combination; the key distinguishing characteristic of case files is that they result from processes which are structured, at least in part.

Typical characteristics of case files are that:

- they are numerous;
- they are structured or partly structured;
- they are used and managed within a known and predetermined process;
- they need to be retained for specified periods, as a result of legislation or regulation;
- they have similar content and/or structure;
- they have a known opening and closing date;
- they can be opened and closed by case-workers (practitioners, clerical staff or data processing systems) without the need for management approval.

Because case files are often structured, they generally contain several sub-files, usually configured by means of a template. They may also contain volumes. See section 3.3 for details of relevant functionality, all of which applies to case files as it does to other files.

Case management frequently involves another business application system (for example a licence application processing system or an enquiry tracking system). It also often depends on workflows (as described in section 10.4).

Ref	*Requirement*	*Test*
10.5.1	An administrative role must be capable of configuring the ERMS to allow at least one “case worker” role (see glossary), with the specific feature that case worker roles can have different access permissions for case work classes and non-case work classes. *In many cases case workers will be able to create, open and close case files as part of their day-to-day business, but they will not have permissions to create, open and close non-case files. In non-case files this level of authority may be granted only to administrative roles.*	Y
10.5.2	The ERMS should support an optional file titling mechanism, to be configured by an administrative role, which includes names (for instance persons’ names) and/or dates (e.g. dates of birth) or unique file identifiers as file names, derived and automatically validated from external lists.	Y

Ref	Requirement	Test
10.5.3	The metadata used for automatically constructing file titles (as in 10.5.2) must be mandatory metadata or suitable defaults should be provided when the titling mechanism is defined. If the underlying metadata values (for instance names, dates, etc.) which have been used to create the file title are modified, the ERMS should not automatically update the file's title.	Y
10.5.4	The rules for automatically constructing file titles (as in 10.5.2) should be configurable to be different for different classes. *The three requirements above can be appropriate for case files. Any list used for validation may be managed within the ERMS or may be external to it.* *Where a file title has been allocated automatically using a mechanism that incorporates metadata such as a person's name, dates of birth, etc. it is possible for this original metadata on which the title is based to be updated. For example, a person's name may change, a date of birth may have been entered incorrectly, etc. In these circumstances the file title based on the metadata should not be automatically modified to reflect the change as the file title may already have been used (e.g. in correspondence, registered on another system, etc.). Apart from the requirement that the file title is not automatically modified, MoReq2 does not mandate the possible outcomes.* *Several different outcomes are possible, including:* ♦ *the metadata change is ignored and the file title stays the same;* ♦ *an administrative role is alerted that the metadata has been changed, and the role is able to (optionally) update the file title;* ♦ *the user making the change is warned that the metadata has been used in the file title and asked to confirm the metadata change;* ♦ *the user making the change is prevented from updating the metadata and advised to forward the desired changes to an administrative role who is able to edit the metadata.*	Y
10.5.5	The ERMS must allow the creation of case files by any user authorised as a case worker.	Y
10.5.6	The ERMS must allow users to access and open a case file by entering its case-specific file identifier. *In most case files the file identifier, for instance title or reference number, will be provided by an external system. An interface should enable the user to validate a manually entered identifier against this. This is different from, and additional to, the system identifier and classification code.*	Y

Ref	Requirement	Test
10.5.7	The ERMS must provide an Application Programming Interface (or comparable capabilities) to enable integration with other business applications. This must include at least the following functionality: ♦ the other business application to create, open and close ERMS case files; ♦ the other business application to provide the ERMS case file title; ♦ the classification code of a newly-created case file to be passed to the other business application; ♦ the other business application to pass records to be declared into the ERMS case files; ♦ the other business application to apply a retention and disposition schedule to an existing closed file; ♦ error handling in case either system initiates an action which is considered invalid by the other system. *It is as if the business application should act as a normal user – the ERMS should not differentiate between the two.* *MoReq2 does not specify the nature of the error processing. However, specific outcomes are identified in the following two requirements.*	P
10.5.8	The ERMS must, upon receipt of an apparently invalid request from an external business application: ♦ not complete any invalid action; ♦ not result in a software failure in either the ERMS or the external application.	Y
10.5.9	The ERMS should, upon receipt of an apparently invalid request from an external business application alert an authorised user so that corrective action can be taken.	Y
10.5.10	Where the ERMS interfaces with another business application it must be possible for an administrative role to limit the other application's actions to one or more specified classes within the ERMS's classification scheme. *In other words, it must not be possible for the other application to take any actions that affect classes, files or records beyond the class(es) for the case files.*	Y

Ref	Requirement	Test
10.5.11	Where the ERMS interfaces with another business application it should be possible for a user to switch easily between the related files in both applications. *In other words, a user that has used the features of the other business application to locate or identify a case or case file (for example, using the application's postal address look-up features to identify a specific case) must be able to open that case file in the ERMS easily, that is without having to re-type the case file identifier. Likewise, a user who has opened a case file in the ERMS (by browsing the classification scheme, by searching or by any other means) must be able to switch to the corresponding case information in the other business application in the same way.*	N
10.5.12	Where the ERMS allows another business application to create new case files it must be able to receive relevant file metadata from the other application.	Y
10.5.13	The ERMS must allow case files to be configured with metadata elements that are specific to case files. *For example, a case file may need one or more metadata elements to indicate "status" or "progress".*	Y
10.5.14	The ERMS must allow users to retrieve, declare records into, and carry out all other valid actions on, case files by using a case file identifier instead of a classification code. *Most case files are identified by a unique case identifier such as an account number or a complaint number. Users must be able to work with these files simply by specifying this identifier, and without the need to use the ERMS classification code (though use of the code will remain possible).*	P
10.5.15	When the ERMS receives records with structured content from another business application, it should be able to extract metadata automatically from the records.	Y
10.5.16	When the ERMS receives records with structured content from another business application, it should be able to use the extracted metadata to declare the records into the appropriate file. *For example, if an ERMS receives electronic claim forms from a benefits claim processing application, it should be able to extract the claimant identifier and form type, then use these to classify the forms to the correct case file (using the claimant identifier) and sub-folder (using the form type).*	Y
10.5.17	The ERMS must ensure that all actions performed on any class, file, or record, whether by an authorised user or by another business application, are logged in the audit trail.	Y

Ref	Requirement	Test
10.5.18	The ERMS must be capable of producing reports on all actions performed on any specified file(s), whether by an authorised user or by another business system.	Y
10.5.19	The ERMS must be able to produce reports for administrative roles, showing at a minimum: ♦ the numbers of records declared into case files automatically from other business systems per time period; ♦ the numbers of records declared into case files manually per time period; ♦ the numbers of case files opened and closed automatically by other business systems per time period; ♦ the numbers of case files opened and closed manually per time period.	Y

10.6 Integration with Content Management Systems

This section addresses the requirements for the integration of "Content Management Systems" (**CMSs**) with ERMSs. Modern content management systems include most or all of electronic document management system (EDMS) functionality (see section 10.3); This section addresses only the CMS-specific functional requirements for an ERMS – it does not describe the functional requirements for CMSs or EDMSs, and does not include sufficient functionality to make the ERMS perform tasks normally associated with CMSs.

CMSs include and extend EDMS functionality across all forms of information (content), not just records. CMSs usually deal with different aspects of managing information than ERMSs. Common characteristics are:

- publishing information, often to websites or portals, and sometimes to several channels using different renditions;
- managing information that originates from several sources;
- reformatting information and/or migrating it to some different rendition(s);
- relating different versions, renditions and translations of documents to each other;
- managing components of documents.

At the time of writing, the most frequent use of the term CMS, and the most frequent need for integration with an ERMS, is likely to apply to web publishing. However, this section is intended to allow for both web publishing and other sorts of CMS.

Content management functionality may be provided by a CMS separate from the ERMS, or by an integrated package that provides both CRM and electronic records management functionality. For ease of explanation, this section describes MoReq2 requirements as if the CMS and ERMS are separate; this separation is not a requirement.

The relationship between an ERMS and a CMS is shown, in highly simplified form, in figure 10.1.

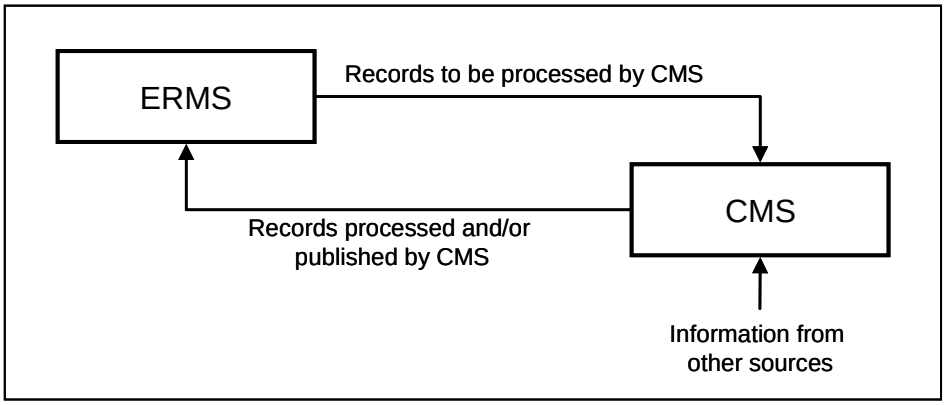

Figure 10.1

This figure shows that:

- Copies of records can be passed from the ERMS to the CMS for processing (the processing usually involves editing, migrating to different renditions, and publication).
- Records can be passed from the CMS to the ERMS for capture. This can happen while information is being processed by the CMS, or after it has been processed and published. The records may include (though are not limited to) web pages, web sites, and new renditions of existing records.
- The CMS can also receive information from other sources, so the records it passes back to the ERMS can consist of a combination of information that originated from the ERMS and information from elsewhere.

Note that the words “can be passed to…” cover several possibilities:

- copies of the records are transmitted between applications;
- the records are stored in a repository that is common to the CMS and the ERMS, and only messages identifying the documents or records are transmitted between the applications;
- the records are stored in a repository that is common to the CMS and the ERMS, and both act on them without the need to transmit any information;

In this section, the “passing of copies” may refer to any of these (or other such) scenarios.

CMS technology is evolving rapidly, so organisations that require CMS integration must specify their individual requirements; reliance on this section alone is not likely to suffice. This section should be viewed as a starting point, to prompt further analysis.

Ref	Requirement	Test
10.6.1	The ERMS must be able to receive as input from the CMS records, including specified metadata, and must either: ♦ automatically capture the records into the appropriate file(s) based on their metadata; or ♦ allow a user to specify the appropriate file(s).	Y
10.6.2	The ERMS must be able to capture as records CMS-specific components and file types, including: ♦ content management log files; ♦ style sheets.	Y
10.6.3	The ERMS must accommodate metadata required by the CMS in addition to the records management metadata specified by MoReq2. *For example, a CMS may use metadata elements to store information needed for content management, such as:* ♦ *IP address;* ♦ *status;* ♦ *language;* ♦ *publication date;* ♦ *effective date;* ♦ *reason for change.* *The ERMS must be able to store these elements, even though they are not required for records management. It is not necessary for the ERMS to be able to store all the metadata produced or used by the CMS; only the elements specified at configuration time need be stored. The elements to be stored need to be determined based on business need.* *Note that this is a highly general requirement. It allows for a wide variety of functions to be carried out by the CMS then stored as metadata stored in the ERMS.*	Y
10.6.4	When a record is being passed from the CMS to the ERMS for capture, if that record is related to an existing record stored in the ERMS (for example, it is a different rendition or a translation of the existing record), the ERMS must not delete or change the existing record, but must instead store the new record.	P

Ref	Requirement	Test
10.6.5	When a record related to an existing record (as in 10.6.4) is being passed from the CMS to the ERMS for capture, the ERMS must automatically link the existing and the new records (as in 3.4.23). *This will only be possible if the CMS passes, with the record, the identifier of the existing record, as a metadata value. If the CMS does not pass back this value, then the ERMS cannot fail a MoReq2 compliance test.*	Y
10.6.6	When a record related to an existing record (as in 10.6.4) is passed from the CMS to the ERMS and then captured as a record, the ERMS should ensure that the metadata of the new record is, as far as possible, identical to that of the original record by binding it to the same metadata, with only such relevant differences in the metadata as are required to log the changes and actions of the CMS.	N
10.6.7	When documents are being passed from the CMS to the ERMS in the form of web pages, the ERMS should be able to capture a web page, or a set of web pages, declaring them as a single record. *The ability to capture a set of pages as a single record may be useful in several circumstances, such as storing "snapshot" copies of a web site periodically.* *Capturing web pages is likely to require changes to the references (hyperlinks within the pages, hyperlinks to other web pages, and references to graphical or other components etc.) so as to allow the pages to appear correct and to retain as much as possible of their original functionality. This is unavoidable if web pages that include graphical elements, style sheets, hyperlinks etc. are to be stored in their original formats without losing all functionality and fidelity. The key aspect is that the information-providing content of the web page must not be modified. See requirements 6.1.5 and 6.1.6.*	Y
10.6.8	When records are being received by the ERMS from the CMS, this must be logged automatically in the ERMS audit trail and in the records' metadata.	Y
10.6.9	When a user is selecting records to copy from the ERMS to the CMS, the ERMS must allow the user to use any available CMS metadata values as a basis for selecting the records to be passed. *To continue the example in 10.6.3, a user may select records in a specified class with specified values of "effective date" and "status".*	Y
10.6.10	The ERMS must allow users to initiate the passing of copies of specified records, together with specified metadata, from the ERMS to the CMS. *The metadata to be passed can be specified at configuration time.*	Y
10.6.11	When records are being passed from the ERMS to the CMS, this must be logged automatically in the ERMS audit trail and in the records' metadata.	Y

10.7 Electronic Signatures

Electronic signatures (sometimes referred to as digital signatures) consist of information that is attached to, or is logically associated with, other information, such as an electronic record, and which serves as a method of authentication. The electronic signature typically takes the form of a sequence of characters. It is used with secure algorithms, procedures and "keys" (a long string of characters analogous to a password) to confirm the integrity of a record, and/or to authenticate the identity of the sender or the source of a record. Electronic signatures should not be confused with a bitmap, or scanned image, of a manual "pen and ink" signature on paper – this is not considered secure, and so is unlikely to add to the evidence about authenticity of a record.

An electronic signature, as the term is used in MoReq2, is a form of "advanced electronic signature" as defined in the European "Directive on a Community Framework for Electronic Signatures" 1999/93/EC. An advanced electronic signature is one that meets the definition in the Directive, namely that the signature is:

- uniquely linked to the signatory;
- capable of identifying the signatory;
- created using means that the signatory can maintain under his sole control;
- linked to the data (e.g. record) to which it relates in such a manner that any subsequent change of the data (e.g. record) is detectable.

Another, unrelated, standard for electronic signature frameworks is X.509 (see appendix 7).

Examples of widely-recognised electronic signature algorithms are the Digital Signature Algorithm (DSA) as defined in FIPS 186-2 (see appendix 7) and RSA/SHA-1.

E-mail has become the default means of communication for many organisations and this has resulted in the widespread movement of documents and records in relatively uncontrolled environments. The use of electronic signatures for authentication and integrity confirmation is therefore becoming widely adopted, especially where records of business transactions are involved.

Electronic signatures are also used to provide non-repudiation – repudiation refers to any act of disclaiming responsibility for a message. Non-repudiation provides proof of the integrity and origin of data which can be verified by any third party at any time. It prevents an individual or entity from denying having performed a particular action related to data such as approval, sending, receipt, knowledge (recognizing the content of a received message) or delivery (receipt and knowledge).

The requirements in this section apply only where there is a requirement to manage records bearing electronic signatures. At the time of writing, electronic signatures are still subject to change and uncertainty as new infrastructures and algorithms are tested and introduced. This state of affairs is likely to continue. Users of MoReq2 should therefore confirm requirements and implications for long-term storage with appropriate authorities.

There are no requirements in this section relating to individual countries' legislation on electronic signatures. By way of illustration, some laws require that a signature be retained complete to have value, while others require only the retention of metadata about a signature. Where these are relevant they may be dealt with in a country-specific chapter zero.

Ref	Requirement	Test
10.7.1	The ERMS must be able to capture, verify if required, and store, at the time of record capture, electronic signatures, associated electronic certificates and details of related certification service providers. *This is essential as it will not always be possible to recreate this information at later times.*	Y
10.7.2	The ERMS must enable administrative roles to configure the system, either at configuration time, or at a later date, to store verification metadata for electronically signed records, including public keys, with the record at time of capture in one of the following ways: ♦ the fact of successful verification; ♦ specified information regarding the verification process; ♦ all verification data. *This is essential as it will not always be possible to recreate this information at later times.*	Y
10.7.3	The ERMS should have a standards-based interface which permits the introduction of new electronic signature technologies as they are introduced. *An example of a suitable standard basis is the XML Key Management Spec (XKMS, see appendix 7).* *This is especially valuable given the changes occurring in this area.*	N
10.7.4	The ERMS should be capable of checking the validity of an electronic signature, including checking the certificate of a record at the time of capture against an electronic certificate revocation list and should store the result of the check in the record's metadata. It should report any invalid check result to a specified user or administrator role. *This is valuable as it may not always be possible to perform this check on the information at later times.*	Y

Ref	Requirement	Test
10.7.5	When capturing e-mail messages the ERMS must be able to capture automatically, and preserve as metadata, details about the process of verification for an electronic signature, including: ♦ the fact that the validity of the signature was checked; ♦ the identity of individual initiating the check (where relevant); ♦ the certificate issuer; ♦ the serial number of the electronic certificate, verifying the signature; ♦ the certification service provider with which the signature has been validated; ♦ the date and time that the checking occurred. *This is essential as it may not always be possible to recreate this information at later times. Because software changes, because certificates expire, and because external authorities can cease to exist, electronic signatures cannot be guaranteed to be verifiable over long periods; hence this requirement to log the fact that a signature was successfully verified.*	Y
10.7.6	The ERMS should include features which demonstrate that the integrity of records bearing electronic signatures has been maintained. *An example of this would be the verification of an electronic signature. This demonstration of integrity should apply even if an administrative role has made authorised changes to the metadata of the record.* *The way in which this might be achieved is not prescribed.*	N
10.7.7	The ERMS should be able to store with the electronic record: ♦ the electronic signature(s) associated with that record; ♦ the electronic certificate(s) verifying the signature.	Y
10.7.8	It should be possible for the ERMS administrator to define whether the ERMS will store the validation ticket returned by the system that checked electronic signature. *The validation ticket is sometimes referred to as a token.*	Y
10.7.9	The ERMS should enable an administrative role to apply an electronic signature to a file or record or transfer message during an export or transfer process so that the file's, record's or transfer message's integrity and origin can subsequently be verified. *A transfer message is a message sent between application systems as part of the protocol used to manage transfers between the systems.*	Y

Ref	Requirement	Test
10.7.10	An electronic signature applied during export or transfer (see 10.7.9) should be capable of external validation so that the file's, record's or transfer message's integrity and origin can subsequently be verified. *To do this the ERMS must be capable of exporting an electronic certificate with the organisation's public key, with the record.*	Y

10.8 Encryption

Encryption is the process of applying a complex transformation to an electronic object so that it cannot be presented by an application in a readable or understandable form unless the corresponding decryption transformation is applied. This can be used to secure electronic objects, by use of transformations which require the use of secure electronic key codes.

The requirements in this section apply only where there is a requirement to manage records which are encrypted.

Ref	Requirement	Test
10.8.1	Where an electronic record has been sent or received in encrypted form by a software application which interfaces with the ERMS, the ERMS must be capable of restricting access to that record to users listed as holding the relevant decryption key, in addition to any other access control allocated to that record.	Y
10.8.2	The ERMS must be able to capture and store, at the time of record capture, information relating to encryption and details of related verification agencies.	Y
10.8.3	Where an electronic record has been transmitted in encrypted form by a software application which interfaces with the ERMS, the ERMS should be able to keep as metadata with that record: ♦ the fact of encrypted transmission; ♦ the serial number of an electronic certificate (where appropriate); ♦ the type of algorithm; ♦ the level of encryption used; ♦ the date and time of the encryption and/or decryption process, where applicable.	Y
10.8.4	The ERMS should be able to ensure the capture of encrypted records from a software application which has an encrypting capability.	Y

Ref	Requirement	Test
10.8.5	The ERMS should allow encryption to be removed when a record is imported or captured. This feature should be configured by an administrative role at configuration time or later. *This feature may be desired in some large scale record archives which have a requirement for long-term access (because encryption etc. is likely to reduce the ability to read records in the long term). In this case, the organisation would rely on audit trail or similar information to prove that the encryption etc. had been present but has been removed. In other environments, this feature may be undesirable from a legal point of view. See sections 5.3 and 3.1 for more details on Transfer and Importing.*	Y
10.8.6	The ERMS should have a structure which permits new encryption technologies to be introduced.	N

10.9 Digital Rights Management

This optional module does not contain any requirements that are testable in their current form. As explained below, testing will be meaningful only when the requirements are adapted to specified technologies.

Digital Rights Management (DRM) and Enterprise Digital Rights Management (sometimes abbreviated to E-DRM) constitute a not yet standardised set of technologies used to protect intellectual property and/or to restrict the distribution of information. DRM is generally associated with the protection of intellectual property (especially in the music, electronic publishing and film industries), while E-DRM is generally associated with placing restrictions on the distribution of business information, for reasons of security or commercial sensitivity. However, the boundaries are not firm and either may be encountered in the context of an ERMS. Accordingly, in the remainder of this section these technologies are referred to as DRM/E-DRM.

Examples of DRM/E-DRM include:

- Electronic watermarking (also referred to as digital watermarking), which embeds visible information about intellectual rights ownership into electronic documents or records. The information is imposed in a complex manner that makes its removal difficult.
- Steganography, which similarly imposes information about intellectual rights, but in a way that is invisible or, in the case of an audio file, inaudible. Special software is required to read the intellectual rights information.
- Copy protection schemes, which use a variety of approaches to prevent copying.
- Features built into documents or records that allow them to be viewed on-screen but not to be printed.
- Expiry features built into documents or records that prevent them from being presented in any way after a specified date has passed.

DRM/E-DRM technologies are at a relatively early stage of development. They are likely to change significantly during the expected lifetime of MoReq2.

These, and similar technologies, can be applied to records in many formats, including digitised sounds and moving pictures.

These technologies provide a particular challenge in records management as they may make future presentation of records difficult or, in some cases impossible. For example:

- Some forms of watermark rely on the presence of "plug-in" software in the viewing application to be wholly effective. A record with such a watermark may be viewable without the plug-in, but it will not be possible to obtain all the watermark information if the plug-in is not available. As time passes the likelihood increases that the plug-in will not be available.

- An e-mail message contains an expiry feature, and so will no longer be readable after a specified date. This problem is particularly insidious as it may not be apparent at the time the record is captured.

At a minimum, user and administrative roles responsible for capturing and managing electronic records should be aware of any DRM/E-DRM features affecting records in the ERMS. Additionally, potential records management difficulties caused by these technologies can be minimised if the DRM/E-DRM features are removed from records at (or around) the time of their capture. However, both of these points are procedural issues, and therefore beyond the scope of MoReq2.

The applicable technologies vary widely, and their effect on records varies equally widely. For this reason, it is not feasible to formulate generic requirements that apply to all the technologies. Therefore this section specifies some high-level requirements that must be expanded by users of MoReq2 if they are to be used for specification and procurement. So, for example, if time-related expiry features are expected, the requirements must be adapted to give specific requirements for dealing with the expiry features.

Ref	*Requirement*	*Test*
10.9.1	The ERMS must be capable of capturing and storing records bearing DRM/E-DRM features.	N
10.9.2	The ERMS should be able to identify the presence of DRM/E-DRM features in a record at the time of capture. Where DRM/E-DRM features are identified, the ERMS should inform the user and provide the following options: ♦ keep the DRM/E-DRM features; ♦ remove DRM/E-DRM features if possible; ♦ stop the capture process.	N
10.9.3	The ERMS should be able to remove DRM/E-DRM features from records during capture. *This may be mandatory in some environments, but cannot be mandatory in the general case as it would require an arbitrary ability to circumvent security features. If DRM/E-DRM features are removed this should be logged in the audit trail.*	N

Ref	Requirement	Test
10.9.4	The ERMS should include the ability to control access to records based on intellectual property restrictions, and generate charging data for such accesses. *This brief statement encompasses a wide range of functionality which is beyond the scope of MoReq2. This requirement may be satisfied by providing the ability to link to a separate application.*	N
10.9.5	The ERMS must be capable of correctly presenting records with DRM/E-DRM features, to the extent that the DRM/E-DRM features permit.	N
10.9.6	The ERMS should be able to retrieve and store at the time of declaration, information stored in the DRM/E-DRM features, to the extent that the DRM/E-DRM features permit. *For example, the identities of the owners of the intellectual property, as encoded in a watermark; or an expiry date.* *This may be mandatory in some environments, but cannot be mandatory in the general case as it would require an arbitrary ability to circumvent security features.*	N
10.9.7	The ERMS should allow new DRM/E-DRM technologies to be introduced.	N
10.9.8	The ERMS should be able to apply DRM/E-DRM features to records during export. *This is especially desirable if a DRM/E-DRM feature has been removed.*	N

10.10 Distributed Systems

This section comprises requirements for organisations that require an ERMS to operate in multiple locations.

Many organisations operate from several sites. Where the sites are relatively close to each other geographically, or when the network connection between all the sites is good (with sufficient capacity), it may be that a single “instance” of an ERMS is most appropriate to cope with all sites. In these cases, all the sites operate as if they were co-located, and the requirements of this section need not apply. However, if the sites are widely separated, and/or if the connectivity between them is not good, then it may be necessary to implement a distributed ERMS; in that case the requirements in this section apply.

There are several different architectural approaches to distributed systems. These include one instance of an ERMS controlling multiple repositories; several instances of an ERMS, each with its repository(ies), communicating with each other; and other approaches. MoReq2 does not specify an architectural approach; it specifies only the key requirements for such distributed environments and uses the term “distributed ERMS” to refer to any such architecture.

Ref	Requirement	Test
10.10.1	The ERMS must be capable of being configured by an administrative role for use across multiple locations.	N
10.10.2	The ERMS should support a distributed classification scheme across a network of electronic record repositories.	Y
10.10.3	The ERMS must allow an administrative role to maintain classes, files, sub-files, volumes and records and their associated metadata and audit trails across the distributed ERMS such that maintenance operations can be carried out once to apply to the entire distributed ERMS. *Maintenance means performing transactions as specified in chapter 3, section 9.1 and elsewhere.*	P
10.10.4	Where the ERMS supports multiple repositories, it should allow an administrative role to specify which repository stores the 'master' copy of each class (and its child classes, records classified to it, etc.). *For example, an organisation may decide to implement one repository for each of its locations, with each location's records being stored in the location's repository (this assumes that the classification scheme design supports this configuration).*	Y
10.10.5	Where the ERMS supports multiple repositories, it should allow an administrative role to specify which repository(ies) automatically store a copy of each class (and its child classes, records classified to it, etc.). *For example, an organisation may decide that:* ♦ *all repositories have to be copied to the head office repository;* ♦ *in one territory, all repositories must be copied to each other.* *Note that this implicitly means that repositories need to be synchronised automatically. This includes the repositories':* ♦ *records and documents;* ♦ *metadata.*	Y

Ref	Requirement	Test
10.10.6	Where the ERMS supports multiple repositories, it should allow an administrative role to specify which repository(ies) the users at each location can access. *For example, an organisation may decide that:* ♦ *all users can access only the repository for their location;* ♦ *all users can access the repository for their location and the head office repository;* ♦ *all head office users can access any repository while all other users can access only the repository for their location;* ♦ *all users can access all repositories within their territory (i.e. within a specified set of repositories; this is not intended to imply that the ERMS has to recognise the concept of 'territory').*	Y
10.10.7	Where the ERMS supports multiple repositories, it should allow an administrative role to specify that all audit trails will be copied to one repository.	Y
10.10.8	The ERMS must prevent or resolve any conflicts caused by changes made in different locations. *For example, a potential conflict may arise if two administrative roles in different locations make a different change to the metadata of the same class which is stored in a third location.*	P
10.10.9	The ERMS must allow an administrative role to monitor both the entire distributed ERMS as a single entity and individual repositories, providing the same facilities as described in section 9.2.	Y
10.10.10	The ERMS should be able to produce reports (as specified in section 9.2) that cover multiple repositories.	Y
10.10.11	The ERMS should support caching of frequently and recently used files, sub-files, volumes and records accessed from locations using remote repositories. *The following two requirements relate to the performance of the distributed ERMS. They use the convention of expressing variable quantities in angle brackets (for example <xx minutes/hours>) as explained in the introduction to chapter 11.*	Y
10.10.12	Where the ERMS synchronises repositories, they must be synchronised within <xx minutes/hours> of any change (subject to availability of network connections).	N

Ref	Requirement	Test
10.10.13	The ERMS must be capable of propagating any administrative change across all repositories within <xx minutes/hours>. *Requirements 10.10.12 and 10.10.13 are example requirements. MoReq2 does not specify response times as these will be system dependant. See section 11.2 for a full description.* *It is critical that the system architecture allows acceptable response times across all locations. Users of MoReq2 should consider specifying response times for many of the requirements specified in section 11.2 separately for transactions involving information held in remote repositories.*	N
10.10.14	Where the ERMS is capable of creating workflows across distributed systems, it must be able to interchange data across these systems to control the workflow process.	Y
10.10.15	Where the ERMS supports multiple repositories, and where "master" copies are stored in specified repositories (see 10.10.4, it should allow an administrative role to change which repository stores the 'master' copy of each class (and its child classes, records classified to it, etc.); when such a change is made, the ERMS must move the contents from the old location to the new location. *This will be useful when creating or removing repositories, or when moving records to a different repository following geographical moves involving business functions*	Y
10.10.16	Where the ERMS supports multiple repositories, it must allow an administrative role to add a new repository.	Y
10.10.17	Where the ERMS supports multiple repositories, it must allow an administrative role to remove a repository.	Y

10.11 Offline and Remote Working

The requirements in this section cover all types of mobile and offline usage of the ERMS by users who are not permanently connected to the ERMS (or to the network hosting it).

There are several possible scenarios including:

- users who access the ERMS using portable computers (such as mobile, laptop, or notebook computers) or PCs that are connected to the ERMS intermittently;
- users who connect to the ERMS remotely through a dial up connection, or any other connection with low bandwidth connection (e.g. for telecommuting or in a temporary location);
- users who access the ERMS using other mobile devices such as PDAs or smartphones.

Portable computers can be used as normal workstations when connected to the ERMS. However users may need to be able to download and synchronise records and data so that they can work on them whilst offline.

To enable this functionality the ERMS needs to download not only records and aggregations but also their metadata. The ERMS will also need to synchronise all of the modified data when the user is next connected to the system.

In a similar way, portable computers can be connected intermittently to the ERMS, for example when they are used by telecommuters. When they are connected, the portable computer will need to synchronise with the ERMS. Once again there will be the need to download records etc, with the downloaded data being managed on the portable computer in between synchronisations.

PDAs, smartphones and other handheld devices can be used to view and access records, in many cases using a browser interface. Inherent limitations, such as a small screen and restricted performance, mean that in many cases such a device cannot offer the full functionality of a portable or fixed computer. However, such devices are often used for mobile e-mail, notes and calendar applications and there is therefore a necessity to synchronise these types of document with the central system.

MoReq2 does not specify requirements to allow mobile or offline users to maintain the classification scheme (for instance the creation of new classes) and files (for instance closing a file). It may be possible to develop systems that support such maintenance, and MoReq2 does not prevent this.

Ref	*Requirement*	*Test*
10.11.1	The ERMS should allow an administrative role to specify aggregations containing information that cannot be downloaded by any user. *This is a security provision to protect sensitive information from being downloaded and hence placed beyond the control of the ERMS.*	Y
10.11.2	The ERMS must enable a user to download any aggregation or record(s) with accompanying metadata for the user to work on whilst not attached to the network.	Y
10.11.3	The ERMS must log in its audit trail all activity on downloaded aggregations, records, and documents.	Y
10.11.4	The ERMS should note in the aggregation, record or document metadata that the entity has been downloaded for offline use.	P
10.11.5	The ERMS must enable the synchronisation of downloaded aggregations, records and documents upon connection to the system. *That is, it must update the metadata and provide for conflict handling by prompting the user if a conflict occurs.*	Y
10.11.6	The ERMS must update the audit trail with information on offline activity upon connection to the system.	Y

Ref	Requirement	Test
10.11.7	The ERMS must allow a user to capture documents created while offline then to capture them as records later when connected to the ERMS. *If the record has been created offline then the ERMS must either:* ♦ *when re-connected, prompt the user in the synchronisation dialogue to declare it within the appropriate class, file, sub-file or volume;* *or:* ♦ *when re-connected, declare it automatically, using the class, file, sub-file or volume specified by the user while disconnected (subject to validation).*	Y
10.11.8	The ERMS must apply all access and security controls to remotely connected devices. *The ERMS must not provide any opportunity for portable devices to breach the security rules of the ERMS. For example, a user must not be able to download any information which he could not access online. However, MoReq2 recognises that once information has been downloaded to a device the ERMS loses control of it, and that security breaches in this scenario cannot be prevented by the ERMS.* *The following four requirements apply only where the ERMS supports electronic document management, as defined in section 10.3. They use terminology defined in that section.*	P
10.11.9	The ERMS must allow a user to download documents with accompanying metadata for the user to work on whilst not attached to the network.	Y
10.11.10	The ERMS must allow users the option of checking documents out when they are downloaded.	Y
10.11.11	If a user checks out a document and works on it while not connected to the ERMS, the system must allow version numbering to be applied to the document.	Y
10.11.12	If a user checks out a document and changes its version number while not connected to the ERMS, when the user reconnects to the ERMS it must allow the user to upload the revised document, and must at that time automatically check it in and record the changes and the new version number.	Y

10.12 Fax Integration

While e-mail has taken over from facsimile as many organisations' preferred method of rapid communication, there are still some occasions and some locations for which fax is required.

This can be, for example, where the original document is not in electronic format and a copy needs to be sent to another organisation, or where a visible representation of, e.g., a signature is required.

Some fax servers integrate with e-mail systems so that both incoming and outgoing faxes are dealt with as e-mail attachments. In this case the requirements in section 6.3 apply.

Where an organisation's ERMS is integrated with a fax service the following requirements apply.

Ref	*Requirement*	*Test*
10.12.1	The ERMS should provide an application programming interface (API) to enable it to interface with a fax server.	N
10.12.2	The ERMS must be capable of storing faxes in standard formats, for example TIFF v6 image format with Group IV compression. *See ISO 12033 for implications of compression methods.*	Y
10.12.3	The ERMS must support the capture of faxes in an integrated way, so that the capture can be performed by a user from within the fax interface (if such an interface exists), without the user needing to switch to the ERMS.	Y
10.12.4	The ERMS must be tightly integrated with the fax interface to enable users to fax any electronic record that they are currently viewing or working with in the ERMS, from within the ERMS (so long as the record can be presented as a two-dimensional image).	Y
10.12.5	It must be possible for an administrative role to configure the ERMS so that it operates in one of the following ways when an ERMS user sends a fax: ♦ it automatically captures the fax as a record; ♦ it automatically prompts the user, giving the user an option to declare the fax as a record; ♦ it takes no action (and thus relies on the user to initiate declaration if appropriate). *Regardless of which way is chosen, it is acceptable for the ERMS to require the user to classify the record manually and enter metadata manually.*	Y
10.12.6	It must be possible for administrative role to configure the ERMS so that it operates in one of the following ways when an ERMS user receives a fax: ♦ it automatically prompts the user, giving the user an option to declare it; ♦ it takes no action (and thus relies on the user to initiate declaration if appropriate). *Regardless of which way is chosen, it is acceptable for the ERMS to require the user to classify the record manually and enter metadata manually.*	Y

Ref	Requirement	Test
10.12.7	The ERMS should be capable of automatically extracting fax metadata elements from incoming faxes, as specified in chapter 12, for example: ♦ title; ♦ sender; ♦ time and date; ♦ recipient. *This may be accomplished by means of a fax template, and is only relevant where faxes have a predictable internal structure.*	Y
10.12.8	The ERMS should be capable of automatically populating fax metadata elements for outgoing faxes, as specified in chapter 12, for example: ♦ title; ♦ sender; ♦ time and date; ♦ recipient. *This may be accomplished by means of a fax template, and is only relevant where faxes have a predictable internal structure.*	Y
10.12.9	The ERMS must allow a user who is capturing a fax to edit the title metadata element, in order to reflect the content of the fax.	Y
10.12.10	The ERMS should be capable of providing a fax record type for both inbound and outbound faxes to enable a user to enter metadata.	Y

10.13 Security Categories

Chapter 4 describes requirements for controlling access to aggregations and records by role and group. In some environments, such as those involving national security, healthcare, etc., there is a need to limit access further, using a scheme of security categories and security clearances.

These **clearances** take precedence over any access rights which might be granted using the features defined in chapter 4. The requirements in this section apply only in organisations which have this need.

This is achieved by allocating one or more “Security Categories” to classes, files, sub-files, volumes and/or records.

The term “Security Category” is used in this specification to mean “one or several terms associated with a record which defines rules governing access to it.” Note that this term is used expressly for this specification; it is not generally employed.

Users can be allocated a single security clearance which prevents access to all aggregations or records which have been allocated higher security categories.

Security categories can be made up of sub-categories. Some sub-categories are hierarchical in nature. Other sub-categories may be arranged differently, typically in a way which is unique to an organisation or sector.

MoReq2 describes in detail only the requirements for a hierarchical sub-category.

The examples given here are based on national security markings but the same principles apply to markings used in other sectors.

There can also be country-specific national security classification requirements. Where appropriate these can be addressed in chapter zero.

Ref	*Requirement*	*Test*
10.13.1	The ERMS must allow one of the following options to be selected at configuration time: ♦ security categories are assigned to classes, files, sub-files and/or volumes (and not to individual records); ♦ security categories are assigned to individual records (and not to classes, files, sub-files and/or volumes); ♦ security categories are assigned both to individual records and to classes, files, sub-files and/or volumes. *Some organisations will wish to control sensitive records individually, while others will wish to control them at the class, file etc. level.*	Y
10.13.2	The ERMS must allow an administrative role to specify, at configuration time, which roles can specify and change the security category of records and aggregations. *In some organisations, only the information owners will have this privilege. In others, different roles, such as security reviewers or line managers (if such roles exist) will have these privileges.*	Y
10.13.3	The ERMS must allow, but not necessarily require, security categories to be made up of one or more “sub-categories”. *For example, a security category may be made up of three sub-categories, as in the following fictitious example:* ♦ *Security Class;* ♦ *Caveat;* ♦ *Descriptor.*	Y

Ref	Requirement	Test

Each sub-category can be thought of as one dimension defining the security of information. So, in this example, any valid combination of the three sub-categories security class, caveat and descriptor can be applied to the record.

10.13.4 The ERMS must require controlled vocabularies to be defined and maintained by an administrative role, these vocabularies limiting the allowable values for each sub-category — Y

For example, the sub-categories might be as in the following fictitious example:

Sub-category	Allowable values
Class	*Top Secret* *Secret* *Confidential* *Restricted* *Unclassified*
Caveat	*NATO Eyes Only* *WEU Eyes Only*
Descriptor	*Commercial* *Personnel* *Management* *Audit and Accounts*

In this fictitious example, the sub-category "Security Class" is hierarchic (see 10.13.6) while the other sub-categories are not. Requirements for hierarchic sub-categories are common; these are specified below.

Requirements for non-hierarchic sub-categories can be complex and are specific to the sector in which they are employed; with the exception of requirements 10.13.5 and 10.13.7 they are not detailed here.

10.13.5 The ERMS should allow specific implementations of complex or unique security rules. — N

These may be provided by suitable application program interfaces. Examples of the need for these include a need to manage records using marking conventions not covered here such as IDO (International Defence Organisation) markings, or access restrictions for medical records.

10.13.6 For at least one sub-category, the ERMS must support a hierarchy of at least five levels, from unrestricted access at the highest level to highly restricted access at the lowest level. — Y

The sub-category "security class" in requirement 10.13.3 is an example of this.

Ref	Requirement	Test
10.13.7	Where a sub-category and the corresponding clearances are not hierarchical, the ERMS must allow one of the following options to be selected at configuration time: ♦ the ERMS must require a valid clearance to be entered for each new user; ♦ the ERMS must apply a default clearance for new users. An administrative role must be able to redefine the default clearance at configuration time or any other time. *In other words, the clearances must be mandatory for users.*	Y
10.13.8	Where the ERMS applies a default hierarchical clearance to new users (as in 10.13.7) it must apply a default clearance for new users that is the lowest level of clearance in the hierarchy (that is, the most restricted).	Y
10.13.9	The ERMS must restrict access to records (and classes, files, sub-files and volumes depending on the selection made for 10.13.1) to those users who have a security clearance equal to, or higher than, the security category. *Note that this clearance may not be sufficient to obtain access. Access to the electronic records may in addition be restricted to specified users, roles and/or groups, using features described in chapter 4.*	Y
10.13.10	Where a sub-category is hierarchical, the ERMS must use one of the following modes of operation to assign a sub-category to new classes, records etc., selectable by an administrative role at configuration time (or any later time): ♦ the ERMS must apply a default value that is selected by an administrative role; ♦ the ERMS must use the parent aggregation's value as a default; ♦ the ERMS must require an administrative role to enter a value.	Y
10.13.11	Where a sub-category is non-hierarchical, the ERMS must use one of the following modes of operation to assign a sub-category to new classes, records etc., selectable by an administrative role at configuration time (or any later time): ♦ the ERMS must apply a default value that is selected by an administrative role; ♦ the ERMS must use the parent aggregation's value as a default; ♦ the ERMS must allow but not require an administrative role to enter a value.	Y

Ref	Requirement	Test
10.13.12	When a new hierarchical security category or subcategory is defined, the ERMS must apply a default value for all existing classes, records etc. that is the lowest level in the hierarchy; in other words, the default must grant the smallest amount of access permitted by the hierarchy.	Y
10.13.13	The ERMS should allow security clearance to be allocated to a role and inherited by users. Where a security clearance is inherited from a role, the ERMS must allow a different security clearance to be applied at the individual user level.	Y
10.13.14	If the ERMS supports security categories for both records and classes etc. (see 10.13.1), it should be capable of preventing a class, file, sub-file or volume from having a lower security category than any record within it.	Y
10.13.15	If a user attempts to capture a record that has a higher security category than the aggregation into which it is being captured the ERMS must notify the user so that appropriate action can be taken; the ERMS must allow at least the following actions (subject to their being enabled at configuration time): ♦ the security category of the aggregation is raised to that of the record; ♦ the user is denied permission to capture the record into the aggregation; ♦ the record is automatically sent to a specified user for action; ♦ the user is invited to create a new aggregation for the record, with default values of metadata taken from the original aggregation; and then to capture the record into the new aggregation, as one integrated process.	Y
10.13.16	An administrative role must be able to determine the highest security category of any record in any class, file, sub-file or volume by means of one simple enquiry. *In some environments, this will be an important feature to aid manageability.*	Y
10.13.17	Subject to support for requirement 10.13.1, an administrative role must be able to change the security category of a class, file, sub-file, volume or record. *See also 10.13.27.*	Y

Ref	Requirement	Test
10.13.18	The ERMS should support routine, periodic, scheduled, review of security categories, where a review consists of: ♦ allowing a user (with appropriate clearance and permissions) to view specified records and their security categories; ♦ allowing the user to change the security categories. *MoReq2 does not prescribe how this is achieved.*	Y
10.13.19	The ERMS must automatically hold a history of security category values, in the metadata of the records, classes etc. to which they apply.	Y
10.13.20	When a user changes the value of a security category (either during a review as in 10.13.18 or otherwise), the ERMS must allow the user to enter a reason for the change, and must store the reason with the history (as in 10.13.19) as metadata. *See 10.13.2 for details of users allowed to change security categories.*	Y
10.13.21	The ERMS must allow users who have clearance and permissions that allow them to see a record to see the current value(s) of its security category(ies) and any history (as in 10.13.19).	Y
10.13.22	The ERMS should support the allocation of a security category to a class, file, sub-file or volume, which is valid for a defined period of time, and should automatically downgrade the marking to the lowest level security category at the end of that period.	Y
10.13.23	The ERMS should support the allocation of a security category to a class, file, sub-file or volume, which is valid for a defined period of time, and should automatically downgrade the marking to a lower, pre-selected, security category at the end of that period.	Y
10.13.24	The ERMS should support notification to an administrative role of the expiry of a selected time period for which a security category has been allocated to a class, file, sub-file or volume, and allow the security marking to be reassessed and amended. *For example the ERMS should send a notification at "Date of Birth + x years." This is for use in medical records or for other data protection purposes.*	Y
10.13.25	The ERMS must automatically log all changes to security category and sub-category values in the audit trail.	Y
10.13.26	The ERMS must not allow a user to apply a security category to a class, file, sub-file or volume that the user does not have access to.	Y

Ref	Requirement	Test
10.13.27	An administrative role must be able to change the security category of all records and child entities (subject to the option configured at 10.13.1) in a class, file, sub-file or volume in one operation. *This is routinely required to reduce the level of protection given to records as their sensitivity decreases over time.*	Y
10.13.28	The ERMS must provide a warning to an administrative role if any records are having their security category lowered, and await confirmation before completing the operation. *This is especially valuable if the security category of an aggregation is being lowered below the level of records that are stored in it.*	Y
10.13.29	The ERMS must automatically record the history, for example dates and details, of any changes to security category, in the metadata of the relevant class, file, sub-file, volume or record. *The history must include, for each change made, the date, user, values before and after the change, and reason.*	Y

11. NON-FUNCTIONAL REQUIREMENTS

Some of the attributes of a successful ERMS implementation cannot be defined in terms of functionality. In practice, non-functional requirements are important to success. This chapter brings these requirements together.

The sections in this chapter list requirements for the following areas:

- ease of use (section 11.1);
- performance and scalability (section 11.2);
- system availability (section 11.3);
- technical standards (section 11.4);
- legislative and regulatory requirements (section 11.5);
- outsourcing and third party management of data (section 11.6);
- preservation and technology obsolescence (section 11.7);
- business processes (section 11.8).

These non-functional requirements are often difficult to define, and, difficult to measure objectively. It is nevertheless valuable to identify them so that they can be considered, at least at a high level. Some are specific to EDRM, but several are generic to many kinds of IT system.

In addition to this chapter, users of this specification will need to consider organisational needs in relation to current technical and operational standards. They will also need to consider the ERMS supplier's support services including documentation, customisation, training and consultancy.

Organisations will need to add their own requirements in these areas, depending on their size and structure, physical characteristics and current technical operating environment. This section is intended as a checklist of aspects which users will need to consider. These specific requirements will need to be added to the generic requirements given in earlier sections.

Some of the Example Requirements in this chapter use angled brackets to indicate that a user of the specification needs to enter a quantified value or some other application-specific information. For example,

<xx minutes/hours>

means that a user of the specification should enter a length of time, probably measured in minutes or hours, to suit the specific requirement.

Similarly,

<4 seconds>

means that the specification user should specify a time interval; 4 seconds is here suggested as a starting point, for consideration.

In the same way, alternative phrases are also found in angled brackets. So for example the phrase

<every day/on all weekdays/xx days per year>

should be taken to mean “every day, or every weekday, or on a specified number of days per year or similar” as appropriate for the organisation.

In all cases, “xx” may mean any number, no matter how large or small.

Because the requirements are generic, and because different organisations will have widely differing requirements and priorities, the non-functional requirements in this chapter are not tested in the MoReq2 testing framework. The testable attributes that are given here are to be used as a guideline. Organisations and users of MoReq2 will need to analyse their requirements, set their priorities, and conduct their own tests in these areas.

11.1 Ease of Use

When considering non-functional requirements in developing an ERMS specification, these must include the degree of ease of use required, and how it is to be specified. This will depend on the kinds of user for whom the system is intended, and the amount of training that is to be undertaken. Examples of requirements for ease of use are listed below.

Ref	*Example Requirement*	*Test*
11.1.1	The ERMS must allow an administrative role to configure how much of the classification scheme each user role or group of users is able to access. *For example a user or group of users, e.g. caseworkers, may be limited to viewing a single class of the classification scheme or even specific files or sub-files.*	Y
11.1.2	The ERMS must provide online help throughout the entire system.	Y
11.1.3	The ERMS must present the classification scheme graphically in hierarchical form, and allow users to navigate it using the graphical representation.	Y
11.1.4	The online help in the ERMS should be context-sensitive.	Y
11.1.5	The ERMS should include help on use of the classification scheme, including, at a minimum, easy access to the description metadata for classes, files, sub-files and volumes.	P
11.1.6	The ERMS should include a thesaurus to assist users in selecting terms for keywords, descriptions etc. *See 11.4.1, 11.4.2 and 11.8.11*	Y
11.1.7	All error messages produced by the ERMS must be meaningful, so that users can decide how to correct the error or cancel the process. *Ideally, each error message will be accompanied by explanatory text and an indication of the action(s) which the user can take in response to the error.*	N

Ref	Example Requirement	Test
11.1.8	The ERMS user interface should be suitable for users with the widest range of needs and abilities; that is, designed according to suitable accessibility standards and guidelines, and compatible with common specialised accessibility software. *See appendix 7 for appropriate standards and guidelines.*	N
11.1.9	The ERMS documentation should be provided in a useful format such that users with widely differing needs and abilities are all able to use it. *See appendix 7 for appropriate standards and guidelines.*	N
11.1.10	The ERMS must be easy to use and intuitive throughout. *Ease of use may be assessed by a panel of typical users.*	N
11.1.11	The ERMS user interface rules and behaviour must be consistent across all aspects of the system including windows, menus and commands. These must also be consistent with the operating system environment in which the ERMS operates. *The rules should be consistent with other mainstream applications already installed.*	P
11.1.12	The ERMS must be able to display simultaneously multiple records and aggregations.	Y
11.1.13	The ERMS must support a graphical user interface.	Y
11.1.14	The ERMS must allow users to move, re-size and modify the appearance of the windows, and to save modifications into their user profile so that they take effect automatically each time the users log on to the ERMS.	Y
11.1.15	The ERMS must allow users to customise aspects of the graphical user interface. Customisation should include, but need not be limited to, the following changes: ♦ menu and toolbar contents; ♦ screen layout; ♦ use of function keys; ♦ on-screen colours, fonts and font sizes; ♦ audible alerts.	Y
11.1.16	The ERMS should allow users to select sound and volume of audio alerts, and to save modifications into their user profile.	Y

Ref	Example Requirement	Test
11.1.17	The ERMS must allow persistent defaults for data entry where desirable. These defaults should include: ♦ user-definable values; ♦ a fixed default value; ♦ values same as previous item; ♦ values derived from context, e.g. today's date, file reference, user identifier; ♦ as appropriate.	P
11.1.18	The ERMS must allow configurable drop down menus or "pick lists" of metadata element values for data entry. *The content of these lists should be configurable by an administrative role.*	Y
11.1.19	Frequently-executed ERMS transactions must be designed so that they can be completed with a small number of interactions (e.g. mouse clicks or keystrokes).	P
11.1.20	The ERMS should be tightly integrated with the organisation's e-mail system in order to allow users to send records and aggregations electronically without leaving the ERMS. *For example the user should be able to send from the ERMS mail client. The essence of this requirement is that the user must not have to switch to the e-mail application to send the record.*	N
11.1.21	Where requirement 11.1.20 is met, the ERMS should provide this by sending pointers or links to aggregations and records rather than copies, whenever an aggregation or record is sent to another user of the ERMS. *There may be exceptions to this, for example, a remote user who does not have consistent access to the central repository.*	N
11.1.22	The ERMS should indicate whether an e-mail message has an attachment. *For example, by means of an icon.*	Y
11.1.23	The ERMS should support user-programmable functions. *For example, user-definable macros.*	Y

Ref	Example Requirement	Test
11.1.24	Where users have to enter metadata from records which are images of printed documents (e.g. scanned images), the ERMS should provide features to allow the use of optical character recognition to capture metadata from the image (zoned optical character recognition). *For example, the user should be able to select a rectangle of the image that contains metadata such as a date or a title, then convert that image to a metadata value and insert it into the desired metadata element, all in one action.*	Y
11.1.25	The ERMS should allow users to define cross-references between related records, both within the same aggregation and in different aggregations, allowing easy navigation between the records.	Y
11.1.26	When viewing or working with a record or aggregation (class, file, sub-file or volume) of records, whether as the result of a search or not, a user should be able to use ERMS features to find information about the next-higher level of aggregation of records easily and without leaving or closing the record. *For example, when reading a record, the user should be able to find out what class, file, sub-file or volume it is in; if viewing file metadata, the user should be able to find out information about the class in which it is located.*	Y
11.1.27	The ERMS should allow a user who has access to a file or record to check whether another specified user, group or role has access to it. *This is to permit users to specify a user, group or role explicitly. Thus a user can enquire about the rights of another user, in the context of a record or file, without needing to know that user's group or role memberships.*	Y
11.1.28	The ERMS should allow a user to mitigate the risk arising from an error in filing a record by allowing users to place a temporary lock on a record or file with a single click. This temporary lock should bar access to that file or record to all users save for administrative roles; and the ERMS should automatically inform an administrative role that the temporary lock has been applied, allowing the administrative role (and nobody else) to remove the temporary lock. *This is to allow users to correct an error – such as accidentally placing a sensitive record into an unsecured file, perhaps as part of a "drag and drop" operation. Because users are not able to delete, remove or change records, this requires administrative action.* *In order to prevent misuse of this facility it is important that users are given guidelines into the use of temporary locking and that administrative roles check that these are not being abused.*	Y

Ref	Example Requirement	Test
11.1.29	Users should be able to copy records from the ERMS into other working environments, such as a “desktop” folder, using “drag and drop”, without this action resulting in any change to the record or its metadata. *When a copy of a record is dropped into any other environment, it will be acceptable for it to lose its metadata (on the basis that most other environments do not support the MoReq2 metadata model).*	P
11.1.30	The ERMS should provide help which provides visual guidance. *For example, including screen shots and/or animations showing users how to use system features.*	P
11.1.31	The ERMS should allow users to mark areas of the help system as “favourite” areas or similar, so that they can find them easily on later occasions.	Y
11.1.32	A user working with a file must be able to discover easily and quickly the keywords associated with that file. *It must be possible to discover the keywords without having to leave the file, in a way that allows work with the file to be continued without interruption.*	Y
11.1.33	The ERMS should allow users to define classes, files and records as “favourites”, so that they can find them easily on later occasions.	Y
11.1.34	The ERMS should allow users to send “favourites” to other users. *The favourites can be sent be e-mail or by another mechanism.*	Y

11.2 Performance and Scalability

Users of this specification should consider the extent to which the ERMS provides response times in line with user expectations, and whether it is capable of serving the size of user population for which it is intended. Some considerations and example requirements are given below.

The response times experienced by users will also depend on factors outside the ERMS, including:

- network bandwidth;
- network utilisation;
- network latency;
- configuration and utilisation of various server resources.

This specification cannot address such external factors, other than to point out that they must not be ignored. Usually, tests in the live environment are needed to obtain a reliable view of performance.

Accordingly, these requirements should be interpreted with a standardised understanding of "response time". This understanding will vary from environment to environment, depending on the status of the infrastructure.

For example, if the ERMS is being specified for an existing infrastructure, it may be appropriate to specify response time in terms of the time between receipt of a keystroke at the server, and the sending of the response; alternatively, if the ERMS is being specified for a new network it may be more appropriate to specify response time in terms of the time between keying a request at the workstation and receiving the response at the workstation.

Specific requirements for offline and remote working are covered in section 10.11 and these example requirements will need to be further modified in these environments.

The ERMS must be able to perform all functions and operate consistently to meet business and user needs as defined in the example requirements below.

Ref	*Example Requirement*	*Test*
11.2.1	The ERMS must provide adequate response times to meet business needs for commonly performed functions under standard conditions, for example: ♦ <100%> of the total anticipated user population logged on and active; ♦ <100%> of the anticipated total volume of documents managed by the system; ♦ users performing a typical mix of transaction types at various rates; ♦ with consistency of performance over at least ten transaction attempts.	N
11.2.2	The ERMS must be able to return the results of a simple search (the hit list) within <3 seconds> and of a complex search (combining four terms) within <10 seconds> regardless of the storage capacity or number of files and records on the system. *In this context, performing a search means returning a hit list (see 8.1.10). It does not include retrieving the records themselves.*	N
11.2.3	The ERMS must be able to retrieve and display within <4 seconds> the first page of a record which has been accessed within the previous <xx> months, regardless of storage capacity or number of files/records on the system. *This requirement, and that at 11.2.4, apply only to documents that can be presented in the form of pages. If the documents are unusually large, it may be necessary to extend the acceptable response time.* *The inclusion of "within the previous <xx> months" implies the use of a staged or "hierarchical" physical storage mechanism. See also the next requirement.*	N

Ref	Example Requirement	Test
	This requirement is intended to allow for rapid retrieval of frequently-used records, on the understanding that frequency of use is typically correlated with recent use. The timescale is to be inserted by the organisation, based on an evaluation of the time after which the heavy usage of records decreases.	
11.2.4	The ERMS must be able to retrieve and display within <20 seconds> the first page of a record which has not been accessed within the previous <xx> months, regardless of storage capacity or number of files/records on the system. *This requirement is intended to allow for cases where a form of hierarchical storage management is used, where records used infrequently are stored on slower media than more active records, or stored near-line. The timescale is to be inserted by the organisation, based on an evaluation of the time after which the heavy usage of records decreases.* *For both this and the preceding requirement, if all the electronic records are stored using a single physical mechanism (i.e. without staged or hierarchic storage) then the phrase "within the previous <xx> months" is irrelevant and should be deleted.*	N
11.2.5	The ERMS must allow a single implementation of the system to have an electronic record store of at least <xx gigabytes/terabytes/petabytes> or <xx thousand/million/billion> records, and to serve at least <xx hundred/thousand> users simultaneously with the performance levels specified in this section. *Estimates of storage requirements and record and user population to be inserted by the organisation. Note that in large organisations, large volumes of records may accumulate – in some cases this will extend into the billions of records.*	N
11.2.6	The ERMS must provide the performance levels specified in this section with volumes up to at least: ♦ <*xx*> classes; ♦ <*xx*> files per class; ♦ <*xx*> sub-files per file; ♦ <*xx*> volumes per sub-file; ♦ <*xx*> records per volume. *These are indicative metrics only. Organisations should consider whether other similar metrics apply to their circumstances.*	N

Ref	Example Requirement	Test
11.2.7	*It must be possible to expand the ERMS, in a controlled manner, to meet organisational growth up to at least <xx hundred/thousand> users while providing continuity of service.* *The intention of this requirement is that expansion should be possible with only "routine" upgrades that do not result in major interruptions in availability.*	N
11.2.8	The ERMS must support the above performance level, including routine maintenance of: ♦ roles, users and user groups; ♦ security categories; ♦ access profiles; ♦ classification schemes; ♦ databases; ♦ retention and disposition schedules; ♦ disposal holds; in the face of the anticipated levels of organisational change, without imposing undue systems down time or account administration overheads (see also chapter 9). *In cases where performance requirements are strict, it may be necessary to quantify the anticipated levels of organisational change.*	N
11.2.9	The ERMS must be scaleable and must be able to be used in small or large organisations, with varying numbers of differently-sized organisational units and across different geographical locations.	N

11.3 System Availability

In many organisations the introduction of an ERMS and EDMS together will increase users' dependence on the IT network to the extent that they will be unable to continue working if the ERMS and EDMS become unavailable.

Accordingly, users of this specification who are procuring a system should make every effort to identify user requirements for availability, and then to specify these for the procurement. Example requirements for availability are given below.

Ref	Example Requirement	Test
11.3.1	The ERMS must be available to users: from <xx:00> to <xx:00> <every day/on all weekdays/xx days per year>.	N

Ref	Example Requirement	Test
11.3.2	Planned downtime for the ERMS must not exceed <xx> hours per <rolling three month period>. *The definition of "downtime" may depend on the infrastructure and architecture. For example, in some environments, a failure caused by server hardware will be considered as a failure of the ERMS; in other environments such a breakdown will be considered as a different kind of failure, not attributable to the ERMS.* *A suitable definition needs to be agreed; as a starting point the following is proposed: "The ERMS is considered to be down if more than <xx%> of users are unable to perform any normal ERMS function and if this failure is attributed to any component of the ERMS other than the user's workstation."*	N
11.3.3	Unplanned downtime for the ERMS must not exceed <xx hours/minutes> per <rolling three month period>. *In a procurement it may be appropriate to request quantitative evidence about mean time to resolve problems support of this requirement.*	N
11.3.4	The number of incidents of unplanned downtime for the ERMS must not exceed <xx> per <rolling three month period>. *In procurement, it may be appropriate to request quantitative evidence about mean time between failures in support of this requirement.*	N
11.3.5	In the event of any software or hardware failure, it must be possible to restore the ERMS to a known state (no older than <the previous day's backup>) within no more than <xx> hours of working hardware being available.	N

11.4 Technical Standards

The ERMS should comply with relevant de facto and de jure standards. Where possible, it is desirable that the ERMS should make use of open rather than proprietary interfaces.

Users of this specification may need to specify requirements for standards covering:

- hardware environment (for server platforms and workstation environments);
- operating system environment (for server platforms and workstation environments);
- workstation (client) software architecture;
- user interface;
- relational database and interface;
- network protocol and network operating system;
- interchange standards;

- application program interface and developer kits.

When using this specification for procurement, it will be necessary to add further details of the technical environment, including all ERMS interfaces (e.g. legacy systems, office systems) and any plans for change.

Additionally, users of this specification will need to consider their individual requirements for standards:

See appendix 7 for a definitive list of the standards used in this specification.

Ref	*Example Requirement*	*Test*
11.4.1	If a monolingual thesaurus is implemented with the ERMS, it should comply with standard ISO 2788, Guidelines for the establishment and development of monolingual thesauri.	Y
11.4.2	If a multilingual thesaurus is implemented with the ERMS, it should comply with standard ISO 5964, Guidelines for the establishment and development of multilingual thesauri.	Y
11.4.3	The ERMS must support the storage of records using file formats and encoding which are either de jure standards or which are fully documented. *Users may wish to specify file format and encoding requirements for their organisation.*	P
11.4.4	The ERMS should store all dates in a format compliant with ISO 8601, Data elements and interchange formats – Information interchange – Representation of dates and times.	Y
11.4.5	The ERMS should store all language names in a format compliant with ISO 639, Codes for the representation of names of languages.	Y
11.4.6	If the ERMS is to manage records in multiple languages or using non-English characters, it should be capable of handling ISO 10646 encoding (Unicode).	Y

11.5 Legislative and Regulatory Requirements

The ERMS must conform to legislative and regulatory requirements, which typically vary from region to region and between industries.

MoReq2 does not address the need to maintain physical records. Such a need may or may not exist, according to the legislative and regulatory environment; where there is such a need, care needs to be taken to preserve integrity and usability of electronic and physical records taken as a whole. These issues should be addressed by appropriate organisational policies.

The following requirements will require localisation, in a “chapter zero”.

In addition, users of MoReq2 will need to consider requirements that are specific to their industry, market sector, etc.

Ref	Requirement	Test
11.5.1	The ERMS must conform to locally-applicable standards for legal admissibility and evidential weight of electronic records.	N
11.5.2	The ERMS must comply with locally-applicable records management legislation.	N
11.5.3	The ERMS must not include any features which are incompatible with locally-applicable data protection, freedom of information or other legislation.	N
11.5.4	The ERMS must comply with any locally-applicable European, national or local regulatory requirements, guidelines or codes of practice for the industry, business function or sector.	N

11.6 Outsourcing and Third Party Management of Data

Many organisations use external service providers to store and manage records. In some cases, these are records that are no longer active (or have low recall requirements) but which need to be retained for a legislative period demanded by legal/government stipulation, industry regulators or for long term preservation.

Other organisations use Application Service Providers (ASPs) to manage active records as well as those that have been archived. Organisations send their documents or records – invoices, customer correspondence, mortgage application documents etc. – to be indexed and stored by the ASP. The documents are then available for retrieval and presentation by the organisation's staff over the internet or through a wide area network.

The management of electronic records by a third party requires that the contract with the service provider has clearly defined procedures and controls in place in order to meet regulatory requirements, adhere to best practice for legal admissibility of electronic records, and meet the business demands of the client for access and availability.

The contract will need to include provisions that:

- the service provider's management must be to a standard at least as good as that of the client's management of its records internally;
- the client will be able to recover the records from the service provider in the future, and still be able to continue the management of the records to the organisation's standards and meet legal admissibility requirements.

This sub-section draws heavily on ISO 15801 (see appendix 7).

Ref	Requirement	Test
11.6.1	A contract or Service Level Agreement (SLA) must be agreed with the service provider detailing the services that are to be used. *An SLA is a formal negotiated agreement between the client and the service provider. It records the agreed position regarding services, priorities, responsibilities, etc.*	N

Ref	Requirement	Test
11.6.2	Details of the procedures for the transfer of records from the client to the service provider, and from the service provider to the client, must be documented. *This may use communication links between the sites to transfer files and records automatically on a daily or regular basis. The client must be satisfied that the link between the two sites is secure and the protocols are in place to check all records are received, and reports produced listing any discrepancy.*	N
11.6.3	The service provider must be able to provide the client with copies of the audit trail of the processes for logging and storing of the records/files.	N
11.6.4	The service provider must demonstrate that the files/records and metadata stored can be easily transferred back to the client's ERMS without any loss of structure, metadata or content of the records.	N
11.6.5	The service provider must have procedures in place to allow the client to transfer individual files and records.	N
11.6.6	The service provider must be able to provide ready access to the managed records by the client. The service provider must either deliver a presentation of the record, or the original record to the client to a contracted agreed time and price.	N
11.6.7	The service provider should be able to provide the client with the ability to request, view and print records and or files from the client's office. *This can be achieved, for example, by a network connection.*	N
11.6.8	The service provider should be able to provide the client with the ability to request on-line the downloading or transmitting of records and or files between the client's ERMS and service provider's storage facility.	N
11.6.9	The client should be able to request reports on the records held by the service provider and details of retention and disposition schedules etc. This facility should be provided on-line from the client's offices.	N
11.6.10	Services specified in requirements 11.6.7, 11.6.8 and 11.6.9 should: ♦ have contracted response and/or turnaround times; ♦ operate in a secure environment.	N
11.6.11	The client should check that the proposed location of the work is acceptable and that the location meets security criteria appropriate to the client's needs.	N

Ref	Requirement	Test
11.6.12	The client should check that the proposed procedures and storage management processes involve no greater risk to the records than the client's own procedures. *The service provider will need to demonstrate that all the client's records are backed up and in the event of system failure they can be recovered to a contracted timescale.*	N
11.6.13	The client should check that the service provider will provide suitable operational staff where the security of the records is important. *It is an advantage if all employees of the service provider sign a confidentiality agreement as part of their conditions of employment.*	N
11.6.14	Each shipment of records to/from the client and the service provider should be accompanied by a control document stating the identity and number of records and files.	N
11.6.15	Third parties providing transportation services should be organisations that meet the quality and reliability criteria of the client.	N

11.7 Long Term Preservation and Technology Obsolescence

Background

Electronic records held over a long term face technological risks from three directions:

- media degradation;
- hardware obsolescence;
- format obsolescence.

These are discussed briefly below. More detailed consideration is found in ISO 18492, and in a large number of guidance publications produced by cultural memory institutions and others.

Media Degradation

The risk from media degradation arises because all digital storage media have a limited lifetime. The lifetime varies between media, and also according to environmental conditions.

The following precautions can be taken to avoid loss of information due to media degradation:

- ensure all media is stored, used and handled in suitable environmental conditions;
- routinely replace media (by copying information from them to fresh media) before the expected end of life;
- keep several copies of each record, and systematically compare the copies periodically. This approach is typically used in specialist long term data archives; it requires automated systems, further description of which is beyond the scope of this specification.

Hardware Obsolescence

Storage peripherals – tape drives, disc drives – have a limited life expectancy. As they near or exceed this life expectancy, they typically require more maintenance, while at the same time becoming expensive to maintain and repair; eventually they become unrepairable for practical purposes. Information stored on obsolete devices will be lost permanently when the device fails unless it has been copied onto other media.

Format Obsolescence

Format obsolescence presents the most difficult problem for any period longer than a few years.

The problem arises because the many protocols and software components involved in the processing "chain" between media and presented information are constantly evolving. They include encoding standards, file formats, and software. Their evolution is rapid, and often does not retain compatibility – this is especially true over periods longer than a few years. Currently, the following techniques are recognised:

- migration (converting information to new formats which can be accessed by current hardware and software);
- emulation (moving the information to new hardware but with a additional software component which emulates the old hardware, thus allowing execution of the old application software);
- technology preservation (continual maintenance of the original hardware; not practical in the long term);
- encapsulation of data and software (a theoretical approach which is involves packaging together records, metadata, ERMS and other software in a standard software "wrapper").

There is at the time of writing no simple, generic method which will guarantee long term access to electronic records. The consensus is that:

- the most appropriate strategy is to hold information only in widely-accepted, stable, open formats (i.e. formats which are comprehensively documented in publicly-available specifications) which have a long expected life, such as XML and PDF/A;
- migration and/or emulation are likely to be the safest options; in practice, both will require attention to preservation metadata – see below.

The requirements in this section support these approaches. Further sources of information are given in appendix 7.

Specific Requirements

Ref	Requirement	Test
11.7.1	The ERMS storage media must be used and stored in environments which are compatible with the desired/expected lifespan, and which are within the tolerance of the media manufacturer's specification.	N

Ref	Requirement	Test
11.7.2	The ERMS must support the monitoring and replacement of storage media to guard against media degradation. *This requires the ERMS, or the storage sub-system it uses, to report on media error rates and to permit the replacement of media that is faulty or that is nearing the end of its life, without compromising the records.*	Y
11.7.3	The ERMS should include features for the automated periodic comparison of copies of information, and the replacement of any copy found to be faulty, to guard against media degradation.	P
11.7.4	The ERMS must allow the bulk migration (rendition) of records (together with their metadata and audit trail information) to new media and/or systems in line with the standards relevant for their format(s).	Y
11.7.5	The ERMS supplier must have a system upgrade programme in place to ensure that the existing information can continue to be accessed without changes to the content.	N
11.7.6	Any system modifications that have been made to the ERMS for organisational requirements must remain in place following a system upgrade.	N
11.7.7	The ERMS should be able to report on the file formats and versions of components. *For example, the ERMS should be able to produce lists of components in specified file formats. This facility would be used in conjunction with a software intelligence, or preservation monitoring, function that aims to identify file formats that are at risk of obsolescence.*	Y
11.7.8	The ERMS should be able to render (see glossary) records from their original format(s) to any specified long term preservation file format(s) at the time of capture, at any subsequent time, or on export. *It is acceptable for the rendering process to be undertaken by a program external to the ERMS so long as the context and links are maintained at all times.*	P
11.7.9	Wherever possible without compromising the integrity of the records, the ERMS should be able to render components from their original format to any specified long term preservation file format(s) at the time of capture, on a subsequent occasion, or on export. *It is acceptable for the rendering process to be performed by a program external to the ERMS so long as the context and links are maintained at all times.*	P

Ref	Requirement	Test
	Where components are rendered, it is essential that the integrity of the records that they form is maintained. The feasibility of this approach generally will depend on the capabilities of both the rendition process and of the software application or viewer used to present the records. For example, if the records are web pages that include (say) GIF image files, it would be acceptable to render the GIF images alone only if the following are all true: ♦ *the GIF components are rendered to a file format that can be presented by the application used to access the web pages; in this example, it is likely that JPEG would be suitable;* ♦ *the references to the GIF images in the web pages are amended as part of the migration process so that they refer instead to the new JPEG images;* ♦ *the original components (the unamended web pages and unrendered GIF components are retained alongside the new components.* *The ERMS must at least support all these actions, and should at best perform them automatically.* *This example is chosen solely for illustration; it does not indicate that there is any reason to migrate GIF images at the time of writing.*	
11.7.10	Whenever records or components are rendered, the ERMS must allow the administrator performing the rendition to enter a reason.	Y
11.7.11	When a record has been rendered into a preservation file format, the ERMS must provide suitable facilities to retrieve the original format and/or renditions, as appropriate. *See also 5.2.3.*	P
11.7.12	The ERMS should be able to export records and their metadata in the form of a Dissemination Information Package as defined in Appendix 7 of the OAIS standard, ISO 14721.	Y
11.7.13	The ERMS should hold at a minimum the following metadata items for a rendered component: ♦ the original file format and version; ♦ date of rendition.	Y

Ref	Requirement	Test
11.7.14	The ERMS should be able to extract from a component, and then store as metadata, technical metadata stored in components. *This metadata would be in addition to the metadata specified in the MoReq2 metadata model. For example, it might include technical; details of an image, such as the TIFF v6 format's metadata or the byte order (little endian or big endian), image length, and image width.*	P
11.7.15	If the ERMS uses any proprietary encoding or storage or database structures, these must be fully documented, with the documentation being available to administrative roles. *This implies it may not be sufficient for the supplier to retain a copy of the documentation; in the timescale being considered, the stability of the supplier is not assured. It may therefore be desirable for a copy of this documentation to be lodged with the user organisation or with a neutral party.*	Y
11.7.16	The ERMS should be able to manage a range of preservation metadata elements for the records and their component parts. *See appendix 9.*	P
11.7.17	The source code of the ERMS should either be open, or a copy of the source code should be lodged in escrow with a neutral party.	N

11.8 Business Processes

Experience has shown that the success of ERMS installations depends, among other factors, on whether it is compatible with the way people work in real life situations. Even if an ERMS contains all the features needed for records management, document management etc., an implementation will only succeed if users find it easy to use. If users find it difficult to use, it will be rejected despite its capabilities.

In recognition of this finding, this section describes requirements intended to promote flexibility and ease of use. Accordingly, most of the requirements are desirable rather than mandatory. The requirements may be met by workflow software that is integrated with the ERMS.

Some of the requirements below call for the ability to perform a specified function "…as an integrated part of a process". In all cases, this means that a user who is performing a process should:

- have the option of performing the process, or of not performing it;
- be able to initiate the function easily, preferably with a single click, and without needing to re-enter information that has already been entered;
- be able to choose, at the end of the function, either to cancel the original process or to return to it at the same point and with the same status as before the function was initiated (without needing to re-enter information that has already been entered).

This is illustrated in figure 11.1.

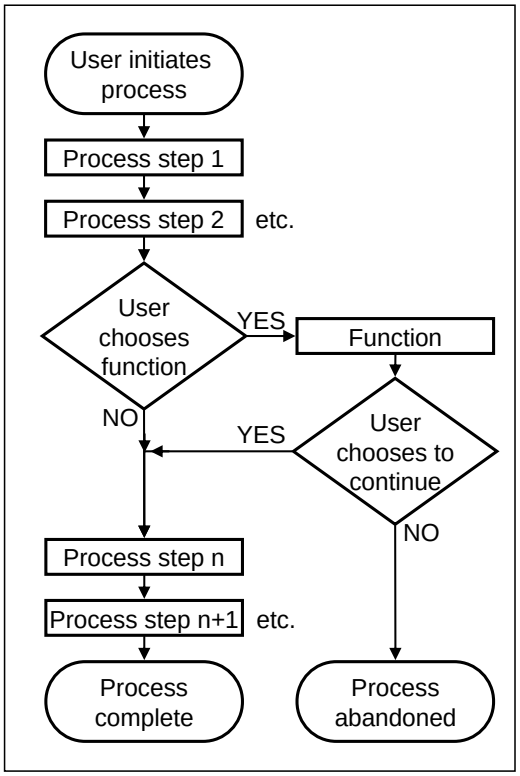

Figure 11.1

All of the following requirements are to be interpreted as being dependent on user access rights.

Ref	Requirement	Test
11.8.1	The ERMS should allow a user who is allowed to change the security category of any record, file or class to check its existing category and permissions as an integrated part of the process of changing it.	Y
11.8.2	When an administrative role user is warned about the lowering of a security category of a record (see 10.13.28) the administrative role should be able to examine the record and/or its metadata as an integrated part of the process.	Y

Ref	Requirement	Test
11.8.3	Whenever a new file or sub-file or volume is created, and where a physical container exists for it, the ERMS should allow the user to print an appropriate label for the physical container, as an integrated part of the process. *This enables a label to be produced containing essential metadata which can then be attached to the physical entity. This could include, but is not limited to, such metadata as:* ♦ *Title;* ♦ *System Identifier;* ♦ *Classification Code;* ♦ *Date of Opening;* ♦ *Security Category (if used);* ♦ *Normal storage location.*	Y
11.8.4	Whenever a user deleting any information receives a warning about existing links (see section 9.3) the user should be able to examine the links and the linked information and/or its metadata as an integral part of the process.	Y
11.8.5	The ERMS should allow a user who is redacting a record, to achieve the following in a single integrated process: ♦ create a redaction; ♦ decide where in the classification scheme the redaction should be filed, and declare it as a record; ♦ link the redaction to the original record; ♦ link the original record to the redaction.	Y
11.8.6	When a user is declaring a record, the ERMS should allow the user to check whether a document has already been declared as a record, as an integrated part of the process. *This should apply to any kind of document.*	Y
11.8.7	The ERMS should warn a user who is capturing a document as a record if that document has already been captured, informing the user of where it is allocated (class, file etc.) and giving the user the option to continue with or abandon the capture.	Y

Ref	Requirement	Test
11.8.8	When a user is capturing a record, the ERMS should allow the user to: ♦ browse the classification scheme (to find the desired class, file etc); ♦ look at the metadata (permissions, keywords, descriptions etc) of any classes and files; ♦ before the capture is completed, as an integrated part of the process.	Y
11.8.9	Whenever a user sees any class, file, record etc. on screen, as the result of a search, while browsing the classification scheme or in any other context, the user should be able to perform any valid action on it directly, without needing to navigate to another part of the ERMS, including at least: ♦ opening it; ♦ determining its parents in the classification scheme; ♦ viewing its metadata or audit trail; ♦ viewing and following its links; ♦ sending it by e-mail; ♦ changing its security category; ♦ viewing users and roles allowed access to it; ♦ printing (or presenting) it; ♦ redacting it; ♦ relocating or deleting it.	Y
11.8.10	The ERMS should allow an authorised user to change the security category of any record, file or class, including the updating of all affected metadata element values, in a single process.	Y
11.8.11	If a thesaurus compliant with ISO 2788 or ISO 5964 is integrated with the ERMS, the ERMS should allow a user who is entering or updating a keyword value (or other metadata element value related to the thesaurus) to use the full features of the thesaurus, such as broader, narrower and related terms and synonyms as an integrated part of the process. *Note that 8.1.18 contains a related requirement for searching.*	Y

12. METADATA REQUIREMENTS

This chapter presents functional requirements for managing metadata. The MoReq2 metadata "model" is presented in appendix 9. Section 12.1 covers the principles of metadata and section 12.2 lists the general metadata requirements.

Metadata includes, in the context of this specification, indexing information and other data needed for effective records management, such as access restriction information. A formal definition is given in the glossary. A more detailed explanation of the role of metadata in records management is found in ISO 23081 (see appendix 7).

12.1 Principles

Scope

It is not possible to define here all the metadata requirements for all possible kinds of ERMS implementation. Different kinds of organisations and applications have particular needs and traditions which vary enormously. For example, some organisations will need indexing focused on account names and transaction dates, while others will need strict hierarchical numbering; some will need volumes, which relate to financial years, while others will not; some will need access controls for security reasons, others for intellectual property reasons, and so on.

This chapter of MoReq2 therefore suggests minimum requirements which are intended as the starting point for customisation and expansion. These minimum requirements are closely related to lists of specific metadata "elements" which the ERMS must be able to capture and process. These elements make up the MoReq2 metadata model in appendix 9.

12.2 General Metadata Requirements

Ref	*Requirement*	*Test*
12.2.1	The ERMS must not present any practical limitation on the number of metadata elements allowed for each entity (e.g. class, file, sub-file, volume, record). *The definition of "practical limitation" will vary according to the application. For example, some organisations with a simple classification scheme may not need as many metadata elements as other organisations with a complex classification scheme.*	P
12.2.2	Where the contents of a metadata element can be related to the functional behaviour of the ERMS, then the ERMS must use the contents of that element to determine the functionality. *For example, where the ERMS stores file opening date metadata, it must populate that metadata automatically whenever a file is opened rather than requiring a user to populate it. Note that this is a general requirement which stretches across many metadata elements. MoReq2 does not attempt to identify all cases in which this is relevant.*	P

Ref	Requirement	Test
12.2.3	The ERMS must allow different sets of metadata elements to be defined for different record types at configuration time. *For example:* ♦ *invoices may need account number metadata;* ♦ *correspondence needs multi-value recipient metadata elements;* ♦ *records which are scanned images will need metadata relating the scanning and indexing processes.*	Y
12.2.4	The ERMS must allow an administrative role to define at configuration time whether each metadata element is mandatory or optional.	Y
12.2.5	The ERMS must support at least the following metadata element formats: ♦ alphabetic; ♦ alphanumeric; ♦ numeric; ♦ date; ♦ logical (i.e. YES/NO, TRUE/FALSE).	Y
12.2.6	The ERMS should support metadata element formats, definable by an administrative role, which consist of combinations of the formats in 12.2.5. *For example, a case might have a reference number in the format nnnnn/aa-n.*	Y
12.2.7	The ERMS must support date formats defined in ISO 8601 for all dates.	Y
12.2.8	At time of configuration, the ERMS should allow definition of the source of data for each metadata element. *Possible sources are described in requirements 12.2.9, 12.2.10, 12.2.11 and 12.2.13.*	Y
12.2.9	The ERMS must allow an administrative role to specify which metadata element values are to be entered and maintained by manual entry or from selection from a controlled vocabulary.	Y

Ref	Requirement	Test
12.2.10	The ERMS should allow for the values of metadata elements to be inherited automatically by default from the next higher level in the classification scheme hierarchy. *For example, for a volume, the value of some of the metadata elements must be inherited from its parent sub-file; and for a record, the value of some metadata may be inherited from the volume into which it is stored.*	Y
12.2.11	The ERMS should allow values of metadata to be obtained from lookup tables or from calls to other software applications. *For example, the ERMS might provide name and post code to an addressing application which then returns a street name to be used as metadata.*	Y
12.2.12	Where the metadata element is populated by lookup tables, if the selection of a value excludes other values in subsequent lookup tables, this should be reflected in the values shown to users in those subsequent tables.	Y
12.2.13	The ERMS should be able to acquire metadata values from: ♦ a document-creating software application (see 6.1.12); ♦ operating system; ♦ network software; ♦ the user at the time of capture or declaration; ♦ rules defined at configuration time for generation of metadata by the ERMS at the time of declaration.	Y
12.2.14	The ERMS must be able to validate metadata when it is entered by users, and when it is imported. The validation must use at least the following mechanisms: ♦ format of the element contents; ♦ range of values; ♦ validation against a list of values maintained by an administrative role. *An example of format validation is that the contents are all numeric, or are in a date format (consistent with 12.2.5). An example of range format validation is that the contents fall in the range between 1 January 1999 and 31 December 2001. An example of validation against a list of values is verifying that an export destination is present on a list.*	Y

Ref	Requirement	Test
12.2.15	The ERMS must be capable of validating metadata using calls to another application (for instance to a personnel system to check whether a personnel number has been assigned, or to a post code database system) or using an internal look-up table.	Y
12.2.16	The ERMS must allow an administrator role to configure the validation (as specified in 12.2.14 and 12.2.15) to be applied to each metadata element. *Different metadata elements will require different validation. So, for example, dates will call for format and range validation while descriptions will not need any validation.*	Y
12.2.17	For metadata element values that are entered manually, the ERMS should allow an administrator role to configure the element so that it supports one of the following data entry modes: ♦ persistent user-definable default values; ♦ a fixed default value; ♦ today's date (for date elements only); ♦ blank element. Additional modes for data entry, not specified above, may also be supported. *A persistent default appears as the default in the data entry field for each item in succession until it is changed by a user. Once changed, the new value remains, i.e. becomes persistent. It should persist at least until the end of a session and ideally between sessions. This applies to all entities for which users may enter metadata values.*	Y
12.2.18	The ERMS should allow configuration such that any metadata element value can be used as a search field in a free text search.	Y
12.2.19	Where a metadata element value is stored in date format, the ERMS should allow searches which recognise the value of the date. *For example, the ERMS should support searches in a date range. It is not sufficient for the date to be stored as a text field.*	Y
12.2.20	Where a metadata element value is stored in numeric format, the ERMS should allow searches which recognise the value of the number.	Y
12.2.21	The ERMS must allow administrative roles to restrict the ability to make changes to metadata values as defined in the access control model (section 13.4).	Y

Ref	Requirement	Test
12.2.22	The ERMS must allow reconfiguration of the ERMS metadata model by an administrative role, and must log such in the audit trail. *For example, it may be necessary to add a new data element such as "Department Identifier" to some document types following an organisational change.*	P
12.2.23	The ERMS must allow metadata elements to be configured at configuration time such that values generated from other application packages, the operating system or the ERMS (for example, e-mail transmission data) cannot be modified by users once they have been captured.	Y
12.2.24	The ERMS must allow metadata elements to be configured at configuration time such that their values cannot be modified by users once they have been captured.	Y

13. REFERENCE MODEL

This chapter provides the reference model for the requirements listed elsewhere in MoReq2.

The sections in this chapter are:

- Glossary (section 13.1);
- Entity Relationship Model (section 13.2);
- Entity Relationship Narrative (section 13.3);
- Access Control Model (section 13.4)

13.1 Glossary

This glossary defines key terms used in MoReq2.

Some significant definitions are taken from, or closely adapted from, glossaries presented in the reference publications listed in appendix 1; these sources are acknowledged below each definition.

Terms defined within this glossary are shown in *italics*.

administrative role

A set of functional permissions allocated to *users* allowed to perform administrative actions.

Note: in MoReq2 this term is used also to specify the people with these permissions.

administrator

A role responsible for the day to day operation of the corporate records management policy within the organisation.

Note: this represents a simplification. Especially in large organisations, the tasks attributed in this specification to Administrators may be divided between several roles, with titles such as Records Manager, Records Officer, Archivist etc.

aggregation

(in the context of MoReq2 only) A class, file, sub-file or volume.

audit trail

Information about transactions or other activities which have affected or changed entities (e.g. *metadata* elements), held in sufficient detail to allow the reconstruction of a previous activity.

Note: an audit trail generally consists of one or more lists or a database which can be viewed in that form. The lists can be generated by a computer system (for computer system transactions) or manually (usually for manual activities); but the former are the focus of this specification.

authenticity

(in the context of records management only) The quality of being genuine.

Source: Adapted and abbreviated from the definition of "record authenticity" in the UBC-MAS Glossary (appendix 1).

Note: an authentic record is one that can be proven

"a) to be what it purports to be,

b) to have been created or sent by the person purported to have created or sent it, and

c) to have been created or sent at the time purported."

Source: ISO 15489.

Note: in the context of a *record*, this quality implies that a record is what it purports to be; it does not address the trustworthiness of the record's content as a statement of fact.

authorised user

A *user* who has permission to carry out the action be described.

Note: the details depend on the context. Different users will have different permissions. MoReq2 does not assume anything about which users or which roles have which permissions. The permissions that authorise a user to carry out an action are granted by the organisation, according to its policies and business requirements.

bulk importing

The process of capturing a set of electronic records, usually from another application and usually with some or all of their metadata.

capture (verb)

(1) The act of recording or saving a particular instantiation of a digital object (source: InterPARES 2 Project Terminology Database).

(2) Saving information in a computer system.

Note: in the context of MoReq2, capturing *records* is used to mean all of the processes involved in getting a record into an ERMS, namely registration, classification, addition of metadata, and freezing the contents of the source document. The term is used more generally to mean inputting to the ERMS and storing other information such as metadata values.

case file

A file relating to one or more transactions performed totally or partly in a structured or partly-structured way, as a result of a concrete process or activity.

Note: there is no universally-accepted definition of these terms, nor of the distinction between case files and the other kinds of files often managed by an ERMS. This definition is therefore developed for, and intended to facilitate the understanding of, MoReq2; its applicability in other situations is not guaranteed.

Note: the records in a case file may be structured or unstructured. The key distinguishing characteristic of case files is that they result from processes which are at least partly structured and repeatable. Examples include files about:

- applications for permits;
- enquiries about a routine service;
- investigation of an incident;
- regulatory monitoring.

Note: typically, other characteristics of case files are that they often:

- feature a predictable structure for their content;
- are numerous;
- are structured or partly structured;
- are used and managed within a known and predetermined process;
- need to be retained for specific periods, as a result of legislation or regulation;
- can be opened and closed by practitioners, end-users or data processing systems without the need for management approval.

case worker

A *user* who works with *case files*.

class (noun)

(in MoReq2 only) The portion of a hierarchy represented by a line running from any point in the *classification scheme* hierarchy to all the files below it.

Note: this can correspond, in classical terminology, to a “primary class”, “group” or “series” (or sub-class, sub-group, sub-series etc.) at any level in the classification scheme.

Note: in MoReq2 class is also used to mean all the records allocated to a class.

classification

In records management, the systematic identification and arrangement of business activities and/or *records* into categories according to logically structured conventions, methods, and procedural rules represented in a classification system.

Source: ISO 15489 (see appendix 7).

classification code

An identifier given to each *class* in a *classification scheme*. Within each class, the classification codes of its child classes are unique.

classification scheme

(In MoReq2) A hierarchic arrangement of classes, files, sub-files, volumes and records.

clearance

See security clearance.

close (verb)

The process of changing the attributes of a *file, sub-file or volume* so that it is no longer able to accept the addition of *records*.

closed

Describes a *file, sub-file or volume* which is no longer open and so cannot accept the addition of *records*.

CMS

Content Management System.

component

A distinct bit stream that, alone or with other bit streams, makes up a *record* or *document*.

Note: this term is not in general use.

Note: the phrase “distinct bit stream” is used to describe what is usually called a “file” in information technology; the word “file” is avoided here to prevent confusion with the records management meaning of “file”. The key concept is that a “component” is an integral part of the content of a record, despite the fact that it can be handled and managed separately.

Note: examples of components include:

- An HTML document and JPEG images that make up a web page;
- A word processing document and a spreadsheet, where the record consists of the word processing document that contains an embedded link (a hyperlink) to the spreadsheet.

Note: components have to be distinct, i.e. separate from each other. If a word processed document contains an embedded spreadsheet (as opposed to an embedded link to a spreadsheet) then the spreadsheet is not considered to be a component; in this case, the word processed document complete with its embedded spreadsheet is a record made up of one component.

Note: an e-mail message with attachments may be one component, as several components, or as several records, depending on the format in which it is stored.

- If the message is stored in a format that includes the body and all its attachments, then there is only one component.
- If the attachments are stored separately from, and linked internally to, the body of the e-mail message, then each attachment and the body of the message is a component.

- If the attachments are stored separately from the body of the e-mail message but they are not linked internally, then each attachment and the body of the message is a separate record; good practice suggest that these records should be linked to each other manually.

configuration time

The point in the lifecycle of the ERMS at which it is installed and its parameters are established.

custodian

(of a record or aggregation) the person or organisational unit having possession of the record(s).

destruction

Process of eliminating […] records, beyond any possible reconstruction.

Source: ISO 15489 (see appendix 7).

Note: Depending on system configuration, this may be the same as deletion, or different from deletion.

Note: This is not intended to imply overwriting of destroyed data or other security measures. Such additional security measures can be implemented but are not required by MoReq2.

digital

Describes information made of distinct digits or numerical values rather than continuously variable values.

Note: this term is not used in MoReq2 to describe records. Although "digital record" is more accurate than "electronic record", the former is rarely used in practice. See *electronic.*

disposal hold

A rule that prevents the *destruction* or *transfer* of *records*.

disposition

Range of processes associated with implementing *records* retention, *destruction* or *transfer* decisions which are documented in retention and disposition schedules or other instruments.

Source: ISO 15489 (see appendix 7).

document (noun)

Recorded information or object which can be treated as a unit.

Source: ISO 15489 (see appendix 7).

Note: a document may be on paper, microform, magnetic or any other electronic medium. It may include any combination of text, data, graphics, sound, moving pictures or any other forms of information. A single document may consist of one or several *components*.

Note: documents differ from *records* in several important respects. MoReq2 uses the term document to mean information that has not been captured as a record, i.e. classified, registered

and locked against change. The word "recorded" in the definition does not imply the characteristics of a *record.* However, note that some documents become *records*.

document type

Describes *documents* that share common characteristics.

Note: for example, documents with common layout, content, retention and disposition requirements, and/or *metadata*. Document types could include, for example:

- application form;
- correspondence (includes letters and faxes and memoranda);
- curriculum vitae;
- e-mail message;
- invoice;
- medical report;
- web page.

Note: in this example, e-mail messages are treated differently than other correspondence, as they may have different metadata requirements; this will not be the case in every organisation.

Note: each organisation needs to define its document types, according to its business needs; the above are purely illustrative.

EDMS

Electronic Document Management System.

Computer-based application dealing with the management of documents throughout the document life cycle.

Source: IEC 82045-1 Document Management.

Note: the functionality required for EDMSs is not included in this specification. However, an EDMS is often used in tight integration with an *ERMS*. See section 10.3 for more details.

electronic

For the purposes of this specification, the word "electronic" is used to mean the same as "digital".

Note: analogue recordings, though they may be regarded as electronic, are not considered as "electronic" for the purposes of this specification as they cannot be stored within a computer system unless they are converted to digital form. It follows that, in the terminology of this specification, analogue records can only be stored as *physical records.*

electronic document

A *document* which is in electronic form.

Note: use of the term *electronic document* is not limited to the text-based documents typically generated by word processors. It also includes e-mail messages, spreadsheets, graphics and images, HTML/XML documents, multimedia and compound documents, and other types of office document.

electronic record

A *record* which is in *electronic* form.

Note: it can be in electronic form as a result of having been created by application software or as a result of digitisation, e.g. by scanning.

ERMS

Electronic Records Management System.

Note: ERMSs differ from *EDMSs* in several important respects. See section 10.3 for more details.

export (verb)

The process of producing a copy of electronic *records*, along with their *metadata*, for another system.

Note: the records remain in the ERMS after export, unlike *transfer*.

file (noun)

An organised unit of *records* grouped together because they relate to the same subject, activity or transaction.

Source: shortened and adapted from ISAD(G) (see appendix 7).

Note: this is the Records Management usage of the term *file*. It differs from the IT usage, for which MoReq2 uses the term *component.*

file format

The internal structure and/or encoding of a *record* or *component* which allows it to be presented into human-accessible form.

Note: examples include:

- HTML v3.2 (a file format for web pages);
- PDF/A v1 (an archival file format for portable documents);
- TXT (ASCII plain text file format);
- XML v1.0 (a file format for extensible markup language which itself relies on ASCII plain text).
- Many proprietary file formats produced by desktop applications such as office suites.

format (noun)

See file format.

group (noun)

A set of *users*.

Note: a group may include users with the same, or different, *roles*. A group is sometimes used to define users' affiliation to an organisational unit such as a department (in which case it typically will include several roles); it is sometimes used to define membership of a virtual team that crosses organisational boundaries, such as all Procurement Officers (in which case it may consist of only users with a specified role); or it may be used in other ways.

import

See *bulk importing.*

keyword

Optional metadata used to describe classes, files, sub-files, and records but not volumes.

Note: it is good practice for keywords to be picked from or validated against a controlled vocabulary, or to be extracted automatically by the ERMS, but this is not mandatory.

metadata

(in the context of records management) Data describing context, content and structure of records and their management through time.

Source: ISO 15489 (see appendix 7).

Note: some models are based on a different conceptual view of metadata. For example, they may treat audit trail information as being entirely metadata. These alternative views are valid and valuable in their contexts, but are not helpful in specifying the functionality of systems, and so are not considered here.

metadata stub

The subset of the metadata for an item that is retained after the item has been disposed of, to act as evidence that the item used to be held and has been properly disposed of.

non-case file

Any file that is not a *case file*.

open

(verb) The process of creating a new *file, sub-file or volume* such that it can accept the addition of records.

(adjective) Describes a *file, sub-file or volume* which has not yet been *closed*, and so is able to accept the addition of *records*.

owner

The person or role responsible for a record or aggregation.

Note: this is the usage in MoReq2; the legal owner of a *record* is the organisation that holds the record.

Note: see also *custodian*.

paper file

A kind of physical file.

Note: examples of paper files include, among others, envelopes, box files and ring binders.

PDF

Portable Document Format, a *file format* primarily for the representation of two-dimensional information.

Note: At the time of writing, this widely used file format is proprietary to Adobe Inc., but a recent version of the format (v1.7) is under consideration as an International Standard (ISO/DIS 32000). Inclusion of the term PDF in this glossary does not represent any form of endorsement. Extensions for the representation of three-dimensional information are under development.

PDF/A

A subset of *PDF* designed for archival use, as defined in the ISO 19005 series of standards.

physical file

A device for holding physical *documents* and *physical records*.

Source: Adapted from PRO Functional Specification (see appendix 1).

physical record

A *record* that is held in a medium outside the *ERMS*, such that the record itself is not individually under the management of the ERMS.

Note: examples include paper records, microform records, and electronic records held on removable media so long as the records are not individually managed by the ERMS.

presentation

The manifestation of an *electronic record* presented by the ERMS to which a *user* can refer.

Note: this may include on-screen display, printed and audio and multimedia presentations.

Note: the exact nature of the presentation can be affected by the software and hardware environment. Typically different presentations of the same *record* can vary in details of font metrics, line endings and pagination, resolution, bit depth, colour space etc. In most cases these differences are acceptable. However, in some cases their potential effects have to be considered separately; these considerations are beyond the scope of this specification.

Note: in the previous version of MoReq the term *rendition* was used with this meaning.

profile

The set of permissions allocated to a *user* or *group* or *role*.

record (noun)

Information created, received, and maintained as evidence and information by an organisation or person, in pursuance of legal obligations or in the transaction of business.

Source: ISO 15489 (see appendix 7).

Note: local national definitions may also apply.

Note: a record may incorporate one or several *documents* (for instance when one document has attachments), and may be on any medium in any format. As a consequence, it may be made up of one or more *components.* In addition to the content of the document(s), a record should include contextual information and, if applicable, structural information (for instance information which describes the components of the record). A key feature of a record is that it cannot be changed.

Note: both *electronic records* and *physical records* can be managed by an *ERMS*.

record type

Describes a *record* made from a *document* with the corresponding *document type*.

redact

The process of hiding sensitive information in a *record*.

Note: this can include applying opaque rectangles to obscure names etc. (the electronic equivalent of censoring paper documents with ink), more secure methods of obscuring information, or removing pages from the copy of a record.

Note: in all cases the totality of the original *electronic record* is not affected. Redaction is carried out on a copy of the electronic record; this copy is called a *redaction*.

redaction (noun)

(of a *record*) A copy of a *record* to which some changes have been applied to remove or mask but not to add to or meaningfully amend existing content.

Source: definition of “instance” in PRO Functional Specification (see appendix 1).

Note: the changes usually result from restrictions on disclosure of information. For example, a *record* may be made available only after individuals’ names are masked or removed from it; in this case, a redaction of the record is created in which the names have been made illegible. The process of masking is sometimes referred to as *redacting*.

Note: In the previous version of MoReq the term “extract” was used with this meaning.

registration

The act of giving a *record* a unique identifier on its entry into a system.

Source: ISO 15489 (see appendix 7).

Note: in the context of MoReq2, registration is part of the process of *capture*.

render

The process of producing a *rendition*.

rendezvous

A point in the workflow where two or more parallel executing activities converge into a single common thread of control.

Source: Workflow Management Coalition Terminology & Glossary, issue 3.0.

rendition

A manifestation of a *record* or *component* in or using one or more *file format(s)* different from the record's native *file format(s)*.

Note: renditions are usually produced to preserve electronic records, that is to minimise the risk of loss of access to their content over time. For example, records produced in a proprietary file format may be stored as renditions in a standard format such as PDF/A or XML.

Rendering a record means rendering some or all of its *components*. After the rendition, the record may have the same number of components as before or it may have a different number of components. For example, a record consisting of 30 components including 10 GIF image objects could be rendered in several ways, including:

- Rendition of the record into PDF/A file format: in this case, the initial record has 30 components and its rendition has one;
- Rendering the GIF components into JPEG file format only: in this case both the record and its rendition have 30 components, and in addition some of the objects in the rendition have to be changed to refer correctly to the newly rendered JPEG images instead of the GIF images.

Note: rendition was used with a different meaning in the original version of MoReq.

repertory

A list of existing *file* titles within each of the lowest levels of the classification scheme.

retention and disposition schedule

A formal instrument that defines the retention periods and consequent disposition actions authorised for *records* described in the schedule.

Source: adapted from National Archives of Australia recordkeeping glossary.

Note: in the previous version of MoReq this was referred to as a retention schedule.

role

The aggregation of functional permissions granted to a predefined subset of users.

Source: PRO Functional Specification (see appendix 1).

security category

One or several terms associated with a *record* or *aggregation* which define rules governing access to it.

Note: security categories are usually assigned at an organisational or national level. Examples of security categories used in government organisations throughout most of Europe are: “Top Secret”, “Secret”, “Confidential”, “Restricted”, “Unclassified”. These are sometimes supplemented by other terms such as “WEU Eyes Only” or “Personnel”.

Note: this term is not in general use. It has been adopted in MoReq2 instead of the term “classification” that is often used by the security community to avoid confusion with the records management meaning of *classification*.

security clearance

One or several terms associated with a *user* which define the *security categories* to which the *user* is granted access.

stub

See metadata stub.

sub-file

Intellectual subdivision of a file.

Note: sub-files are often used in case file management environments. Typically, each sub-file is named, and each sub-file is used to store a specified kind or kinds of records for one instance of a case, such as “invoices”, “assessments” or “correspondence”. They can, however, also be used, in a similar fashion, in non-case file environments.

transfer (verb)

The process of moving complete *electronic files*, along with their *metadata*, to another system.

Source: adapted from PRO Functional Specification (see appendix 1).

Note: the files are often transferred together with all other files in a *class* of the *classification scheme* when the purpose of transfer is to move the files to an archive for permanent preservation.

Note: see also *export*.

user

Any person utilising the *ERMS*.

Note: this may include (among others) administrators, office staff, members of the general public and external personnel such as auditors.

Note: a user may both have *roles* and be a member of *groups*.

user group

See *group*.

user profile

The *profile* of a *user*.

user role

A set of functional permissions allocated to *users* allowed to perform actions that manage records.

A *user* may have several *user roles* but has only one user profile.

Note: in MoReq2 this term is used also to specify the people with these permissions.

version

(of a *document*) The state of a document at some point during its development.

Source: PRO Functional Specification (see appendix 1).

Note: a version is usually one of the drafts of a *document*, or the final document. In some cases, however, finished documents exist in several versions, e.g. technical manuals. In other cases, the versions are translations. By contrast *records* cannot exist in more than one version; see also *redaction*.

vital record

A *record* that is essential for the functioning and/or survival of an organization during and/or after an emergency.

volume

A subdivision of a *sub-file*.

Note: the subdivisions are created to improve manageability of the sub-file contents by creating units which are not too large to manage successfully. The subdivisions are mechanical (for instance, based on number of records or ranges of numbers or time spans) rather than intellectual.

13.2 Entity-Relationship Model

This section repeats part of section 2.3, for ease of reference.

It contains an entity-relationship model which can be used as an aid to understanding the specification. Section 13.3 contains a narrative explanation that describes and explains the model.

The entity-relationship model is shown as figure 13.3. An important aspect of this model is that it need not represent actual structures stored in the ERMS. It represents a theoretical view of the entities associated with records. An ERMS uses these relationships to produce behaviour equivalent to the structures in the diagram. See section 2.2 for further explanation of this point.

The relationships between files, volumes, records and other important entities are depicted in the following entity-relationship model. This is a formal representation of selected structures which can be used to describe the behaviour of an ERMS.

In the diagram, entities – files, records and so on – are represented by rectangles. The lines connecting them represent the relationships between the entities. Each relationship is described by text in the middle of the line; and this should be read in the direction of the arrow. Each end of the relationship has a number which represents the number of occurrences (strictly, the cardinality); the numbers are explained in the key. So, for example, figure 13.1 means “one record is made up of one or more components” (note the direction of the relationship arrow).

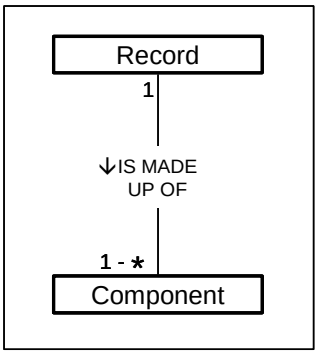

Figure 13.1

A curved line crossing two or more relationships indicates that the relationships are mutually exclusive, for any given instance. So, for example, the curved line in figure 13.2 means “each record is stored in either a volume or in a sub-file but not in both”.

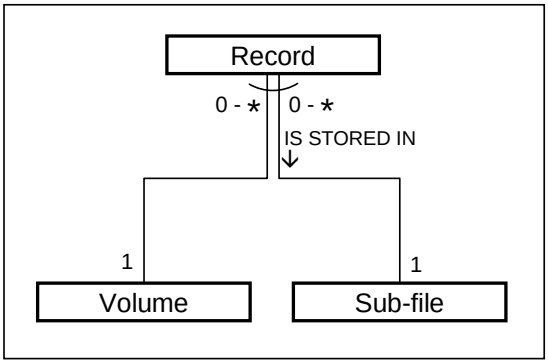

Figure 13.2

Note that the entity class is related to itself by the relationship “is made up of”. This relationship describes, in formal terms, the relationship between classes in a hierarchical classification scheme, where a class may be made up of one or more other classes. If this relationship (sometimes called a recursive relationship) is removed, the model applies equally to non-hierarchical relationships.

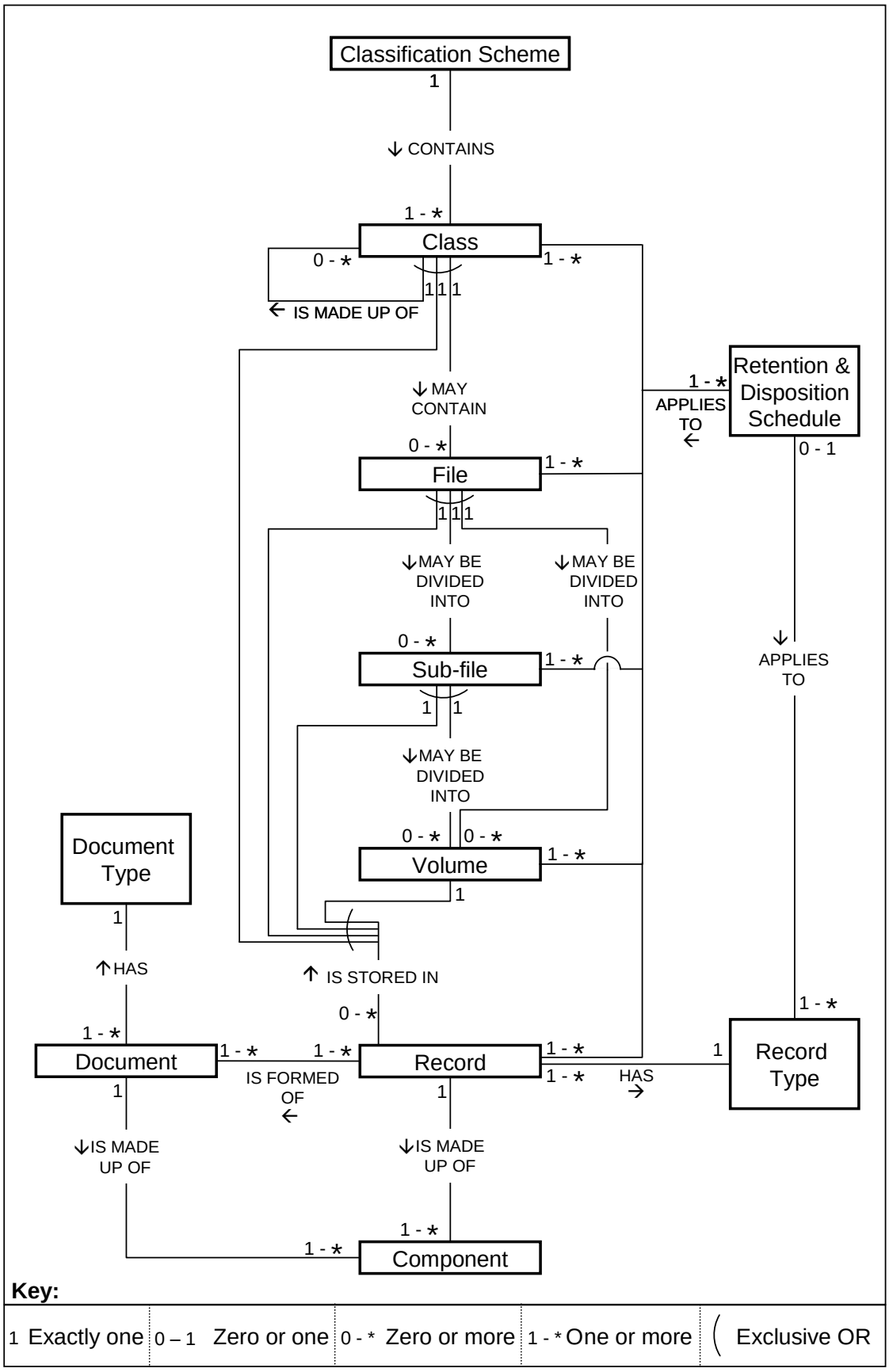
Classification Scheme
1
CONTAINS
1 - *
Class
0 - *
1 - *
1 1 1
IS MADE UP OF
MAY CONTAIN
0 - *
File
1 - *
1 1 1
MAY BE DIVIDED INTO
MAY BE DIVIDED INTO
0 - *
Sub-file
1 - *
1 1
MAY BE DIVIDED INTO
0 - *
0 - *
Volume
1 - *
1
Retention & Disposition Schedule
1 - *
APPLIES TO
0 - 1
APPLIES TO
Document Type
1
HAS
1 - *
IS STORED IN
0 - *
Document
1 - *
1 - *
IS FORMED OF
Record
1 - *
1 - *
HAS
1
Record Type
1 - *
1
1
IS MADE UP OF
IS MADE UP OF
1 - *
1 - *
Component
Key:
1 Exactly one
0 – 1 Zero or one
0 - * Zero or more
1 - * One or more
Exclusive OR

Figure 13.3

13.3 Entity Relationship Narrative

Figure 13.3 shows a simplified model; it does not attempt to represent all possible entities or relationships. Rather, it shows only the most significant ones for this application. For example, it does not show users, roles etc.

The remainder of this narrative describes the entities in the diagram, and their inter-relationships.

Classification Scheme

In order to practice records management principles, an organisation must have at least one classification scheme. This sets out the filing structure (typically consisting of a hierarchy) for a defined part of the organisation. A classification scheme contains several classes.

Class

Hierarchical classification schemes can be viewed as a hierarchy made up of a number of classes, much as a tree is made up of branches. Each class is connected to the hierarchy at one level; can extend over several levels; and can contain smaller classes. Several classes can start at any single level; but each class starts at one level only. As indicated by the “exclusive or” relationship, each class can:

- be made up of classes; or
- can contain files; or
- can store records;

but combinations of these are not allowed.

File

Files occur within classes, at any level in the hierarchy. Files can occur only in classes that do not contain other classes. As indicated by the “exclusive or” relationship, each file can:

- be divided into sub-files; or
- can be divided into volumes; or
- can store records;

but combinations of these are not allowed.

Sub-file

Each file can be divided into sub-files (a configuration option determines whether sub-files can or cannot exist). In practice, some files are not divided into sub-files. Where there is only one sub-file, the concept of sub-file is transparent to users, for all practical purposes. Sub-files are often used in case management applications. As indicated by the “exclusive or” relationship, each sub-file can:

- can be divided into volumes; or
- can store records;

but combinations of these are not allowed.

Volume

Each sub-file can be divided into volumes (a configuration option determines whether volumes can or cannot exist), according to specific rules. In practice, most sub-files are not divided into volumes. Where there is only one volume, the concept of volume is transparent to users, for all practical purposes. The rules may depend on size or number of records, or may depend on transactions or time periods. This practice originated with physical files, in order to restrict them to a manageable size and weight. The practice is, where appropriate, continued with electronic files, to limit them to a manageable length for review, transfer, etc.

Where a file consists of only one sub-file, then its volumes may seem to users to be volumes of the file rather than of its sub-file.

The terms file, sub-file and volume are, in practice, sometimes used loosely or interchangeably – because of the above requirement for transparency. For example, a user will typically ask for "a file" rather than (more accurately) asking for "a volume." This is especially apparent in the case of a physical file that consists only of a single one-volume sub-file. In this case, although the file analytically consists of one sub-file which is made up of one volume, the sub-file and the volume are not always labelled as such (often, the label is only applied when the second sub-file or volume is opened).

Retention and Disposition Schedule

A retention and disposition schedule specifies the rules for keeping and disposing of records. The ERMS can contain several retention and disposition schedules, one or more of which are applied to each class, file, sub-file and volume; they can also be applied to records, and one retention and disposition schedule may be applied to each record type.

Record

At the heart of the system lies the most important entity, the records. These are the reason for the entire records management infrastructure, as they form the account of the organisation's activities.

Records are made from documents. Each record can comprise one or several documents; and each document can appear in several records.

Records are normally stored in volumes. However, records can also be stored in classes (this is an exception described elsewhere). MoReq2 allows for a configuration option that prevents volumes and/or sub-files from being used, in which case records would be stored either in sub-files or in files. Each record can only be stored in one of a volume, sub-file, file or class.

Record Type

Records are assigned a record type. This is used to indicate, and to allow the ERMS to manage, the records in certain ways. Examples of record type might include "invoice" and "web page".

Component

Each record and document is made up of at least one component; some are made up of more than one. For example, a simple web page may consist of only one component – an HTML "file" in IT terms – while a more complex web page may consist of dozens – an HTML "file", GIF "files", JPEG "files" and so on.

13.4 Access Control Model

This section contains a simple model of example roles within an ERMS.

The matrix recognises two main roles, which are themselves divided into roles. The main roles are user roles and administrative roles. These are defined in terms of access to ERMS functionality.

The number of roles shown in this model is only illustrative. It is not meant to indicate that any organisation should implement these roles, not that any organisation should implement this number of roles. Each organisation should define the roles it needs; and these needs will tend to vary over time.

The roles below illustrate an example of access control rights for specific aspects of system functionality according to organisational responsibilities.

There are four example roles defined in the example matrix:

- Central Administrator – this role has control over the configuration of the entire ERMS and the management of the aggregations and records themselves.
- Local Administrator – this is a role with administrative rights over a sub-set of the ERMS or its classification scheme. These roles usually are useful in geographically dispersed organisations.
- Reviewer – this is a specialist role which is primarily concerned with the application of disposition actions defined by Retention and Disposition Schedules.
- End User – the end user role is the standard level of access to the ERMS and comprises those who need to save records into, and access records from, the ERMS for their routine work.

Administrative roles are here divided into two roles only as an example; responsibilities can be divided in other ways. For some small organisations this division might be needlessly complicated, as only one person, with a single role, can manage all the administration. For large organisations it might be an over-simplification because more than two roles are needed (such as Records Manager, Records Officer, Archivist and Data Manager or IT Manager). MoReq2 does not attempt to specify how many administrative roles would be needed in any real organisation.

The role of Local Administrator is given here as an example of one of these. This role can also have several titles in different organisations. In some cases this may be a Local Records Officer, or a Super-user etc.

In any event, administrative roles are only implementing, from a system perspective, decisions taken by more senior management. Such decisions are typically based on the organisation's business requirements and records policy. The decisions also are informed by laws and regulations, such as information laws, data security laws, archival laws and industry regulations; these are addressed in section 11.5.

This matrix is not intended to imply that administrative roles must take management decisions, though in some environments that may be the case.

Administrative roles take actions related to the management of records themselves; their interest is in managing records as entities rather than their content or business context. They may also manage the ERMS hardware, software and storage, ensure backups are taken and manage the performance of the ERMS.

Many organisations also need to integrate the management of business processes with the management of records. In this case there is scope to allocate a particular set of administrative permissions to individual business managers. This could include the ability to monitor and manage a specific group of users or area of the classification scheme.

Although MoReq2 refers to a user role there will be, in the majority of organisations, a number of different user roles and the ERMS should not limit the number of roles that can be configured.

One example of this could be that of a case worker (see section 10.5 Casework). Such a role would have specific permissions within a particular branch of the classification scheme.

Unlike administrative roles, user roles have access to facilities which an office worker or researcher needs when using records. This includes adding documents, searching for and retrieving records. Their interest is primarily in the contents, properties or business context of records rather than their management – in other words, they are interested in the business processes evidenced by the records.

In the matrix, the role of end user shows the access rights that typically are appropriate for the majority of users in an organisation to carry out their business functions.

A further example of a user role is given: reviewer. This shows a level of access control that may be allocated to a sub-set of users for the purposes of reviewing records.

This matrix is best viewed as a starting point, and as the formal basis for assigning rights. Users of this specification will need to consider additional requirements which are specific to their environment.

The formal requirements dealing with this table are in section 4.1; they confirm that **the requirement is not for an ERMS to incorporate the sample access matrix shown here, but to be capable of being configured to the level of detail of an access matrix defined by the user organisation** which may contain an unrestricted number and type of roles and functions**.** It must be possible to configure each cell of the matrix as “yes” or “no”, but with the table having as many columns as the organisation needs.

Other possible roles that might be implemented by organisations include but are not limited to:

- assistant;
- auditor;
- freedom of information manager;
- manager;
- records creator;
- records manager;
- supervisor.

This matrix is divided into sections. These sections group, for convenience, the functions normally associated with classes and files, records, records management and administration.

Function	Roles			
	User Roles		Administrative Roles	
	End User	Reviewer	Local Administrator	Central Administrator
Add new classes	No	No	Yes	Yes
Create new files	Yes	No	Yes	Yes
Change file metadata	No	Yes	Yes	Yes
Maintain classification scheme and files	No	No	Yes	Yes
Delete files	No	No	Yes	Yes
Capture records	Yes	No	Yes	Yes
Relocate a record to a different file	Yes	No	Yes	Yes
Search for and read records	Yes	Yes	Yes	Yes
Change content of records	No	No	No	No
Change record metadata	No	Yes	Yes	Yes
Delete records	No	No	Yes	Yes
Place and remove disposal holds	No	Yes	Yes	Yes
Retention and disposition schedule and disposition transactions	No	Yes	Yes	Yes
Export and import files and records	No	Yes	Yes	Yes
View audit trails	No	Yes	Yes	Yes
Configure and manage audit trail	No	No	No	Yes
Change audit trail data	No	No	No	No
Move audit trail data to off-line storage media	No	No	Yes	Yes
Perform all transactions related to users and their access privileges	No	No	Yes	Yes
Allocate access permissions to local administrators	No	No	No	Yes
Allocate own access permissions also to other users	Yes	Yes	Yes	Yes
Set up and manage case management roles	No	No	No	Yes
Maintain database and storage	No	No	Yes	Yes
Maintain other system parameters	No	No	No	Yes
Define and view other system reports	No	Yes	Yes	Yes

APPENDIX 1 – REFERENCE PUBLICATIONS

This specification was prepared with reference to the following existing specifications and publications:

Ref	Name and Ownership or Source	URL or Publication Details
[1]	Dublin Core Metadata Element Set, Version 1.1: Reference Description	http://dublincore.org/documents/dces/
[2]	Functional Requirements for Electronic Records Management Systems (The National Archives of the UK)	http://www.nationalarchives.gov.uk/electronicrecords/reqs2002/default.htm
[3]	Code of Practice for legal admissibility and evidential weight of information stored electronically (British Standards Institution)	Published by British Standards Institution (www.bsi-global.com) as BSI BIP 0008
[4]	The Preservation of the Integrity of Electronic Records (UBC-MAS Project)(University of British Columbia)	http://www.interpares.org
[5]	Standard 5015.2 “Design Criteria Standard For Electronic Records Management Software Applications” (US Department of Defense)	http://jitc.fhu.disa.mil/recmgt/
[6]	National Archives of Australia – Functional Specifications for Electronic Records Management Systems Software- Exposure Draft	The exposure draft is no longer available. A similar document is available from http://www.naa.gov.au/Images/ERMSspecifications_tcm2-1007.pdf
[7]	Riksarkivet – The National Archives of Norway – NOARK-4 Norwegian recordkeeping system Version 4 – Part 1 Functional description and specification of requirements	http://www.arkivverket.no/arkivverket/lover/elarkiv/noark-4/english.html
[8]	Functional Requirements for the Sustainability of Electronic Records	http://www.nationalarchives.gov.uk/documents/functional_requirements.pdf
[9]	InterPARES 2 Project Terminology Database	http://www.interpares.org/ip2/ip2_terminology_db.cfm
[10]	DLM Forum Guidelines	http://dlmforum.typepad.com/gdlines.pdf

APPENDIX 2 – DEVELOPMENT OF THIS SPECIFICATION

Overview

The MoReq2 specification has been developed for the European Commission by a team from Serco Consultancy (formerly Cornwell Management Consultants plc), based in the United Kingdom. MoReq2 is based on a detailed scoping report that was produced by the DLM Forum.

The Serco project team included specialist consultants, who authored the specification, a small project management and administration team and an Editorial Board, comprising records management experts from throughout Europe and North America (see appendix 4). A late draft of the entire document was reviewed by a semi-independent reviewer.

The test framework documentation was produced by a team from imbus AG.

The requirements were subject to several levels of review.

First, the Authoring Team members conducted peer reviews of each other's work. Then the draft requirements were submitted to a peer review process by panellists representing a broad spectrum of interested parties across the records management community. For ease of reference these were broken down into:

- Archives Panel;
- Specialists Panel;
- Users Panel;
- Vendors Panel.

At selected points, drafts were also reviewed by the MoReq2 Editorial Board The Board met with the Authoring Team on two occasions, providing invaluable direction and guidance; and its members later conducted a third review by e-mail.

An interim and a further draft of MoReq2 were submitted to the European Commission for approval. The drafts were reviewed on behalf of the European Commission by a DLM Forum review group consisting of leading experts from a representative portion of the EU member states.

The structure of the project team is outlined in figure A2.1; see appendix 4 for details of its members.

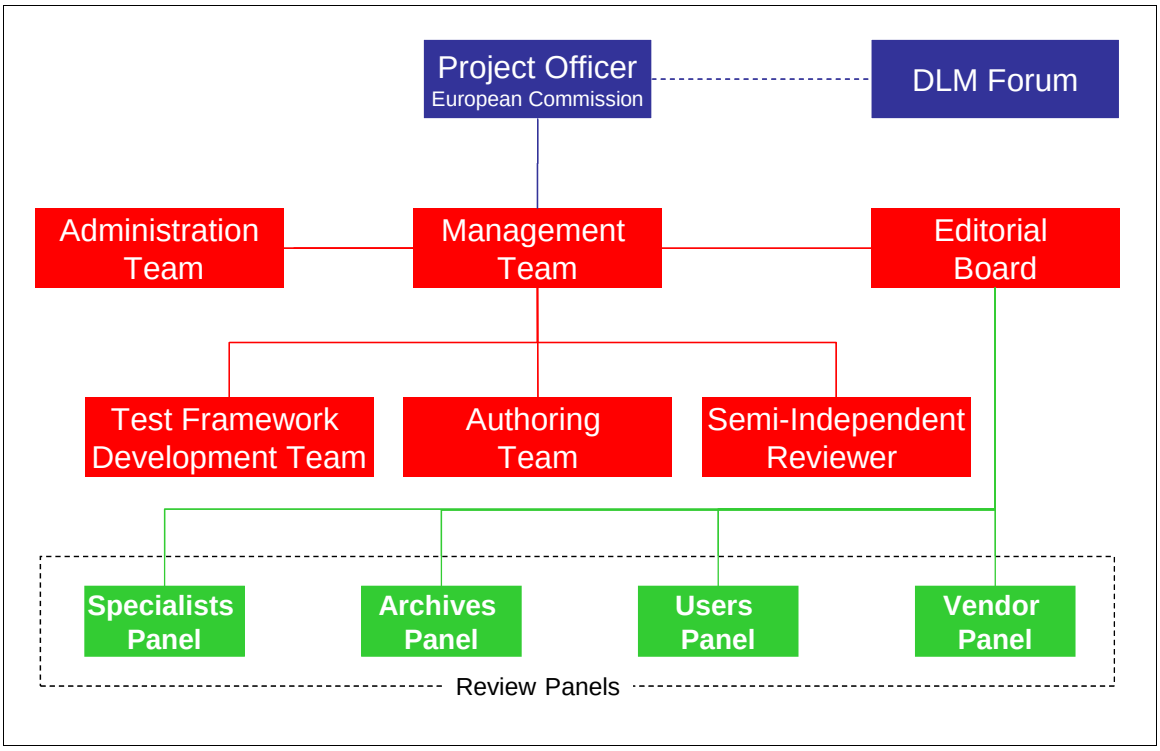

Figure A2.1

A project initiation meeting was held in London, involving the authoring team and the Editorial Board. At this meeting, working protocols and other principles were agreed, and some key references were identified. This was followed by desk research and the identification of relevant reference works which are listed in appendix 1.

A close examination of the reference works was undertaken to ensure that the revised specification includes all relevant requirements.

The original MoReq was imported into a software tool (Telelogic DOORS), a specialised requirements and change management package which was used throughout the authoring process to manage the drafting between the members of the team and to track and incorporate the comments on the drafts from the reviewers. The document was restructured to reflect the MoReq2 scoping document so that the relationship to the original MoReq could be maintained.

As the draft of each chapter was completed it was published onto the MoReq2 web site and all panellists were notified. They were asked to provide their contributions in a specially designed comment form which allowed their contributions to be incorporated in the DOORS software for further processing by the authoring team.

When the majority of chapters had been issued in this way a semi-complete draft of the entire document was compiled and this was distributed to the Editorial Board for them to identify major concerns in advance of a second meeting of the Board.

At this meeting, which was also held in London, consensus was reached between the members on the majority of the issues that had been identified. Following this, the document was redrafted in the light of the agreed way forward.

The redraft of MoReq2 was issued to the EC Project Officer and members of the DLM Forum for review. The official assessment of this interim draft was submitted to Serco and discussed at a project progress meeting in Brussels

The authoring team studied all the comments received, both from the official review and from all the other panellists individually, these were then implemented or rejected as appropriate. This process was intense and iterative as many of the comments were mutually incompatible or not suitable for inclusion in MoReq2. However the overall quality of the comments was extremely high and this led to a refining of the previous draft.

This led to the publication of Draft 2, an essentially complete document, for all panellists to comment on. Upon receipt, these comments were reviewed and implemented or rejected as before.

A complete was issued to the EC Project Officer and members of the DLM Forum for review in October 2007. Upon receipt of the EC review comments a final draft of MoReq2 was prepared which was subject to review by a semi-independent reviewer before publication in January 2008.

APPENDIX 3 – USE OF THIS SPECIFICATION IN ELECTRONIC FORM

This specification has been prepared so that it can be used in electronic form. It has been prepared using Microsoft® Word 2003.

The main advantage of using the specification in electronic form is that it can easily be customised.

The requirements (chapter 3 to 11) are presented in the form of tables, with one requirement per table row. This is illustrated in figure A3.1.

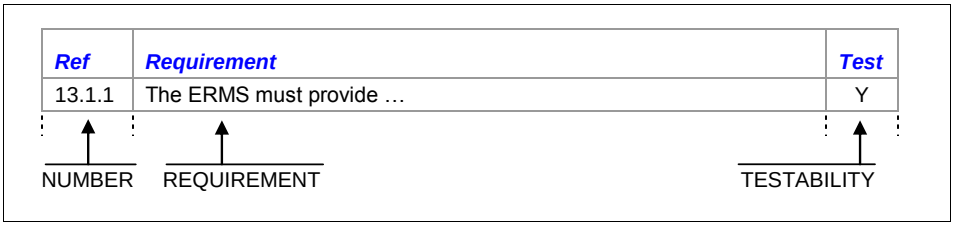

Figure A3.1

The tables consist of three columns:

- **Ref:** a requirement reference number. This is generated automatically by Microsoft Word, as the reference numbers use a "heading" style. The result is that if chapters, sections or requirements are added or subtracted, the numbering changes automatically;
- **Requirement:**
  - the requirement text. This always uses one of the verbs "must" (to indicate a mandatory requirement) or "should" (to indicate a desirable requirement);
  - the rationale text. This is always *in italic text* and provides examples or a further description of the requirement;
- **Test:** each requirement is followed by an attribute called "testable", abbreviated to "test." This indicates whether it will be possible to test compliance with the requirement. Possible values of the "testable" attribute are described below, with examples:
  - Y – The requirement can be tested formally. An example is "The ERMS must allow at least three hierarchical levels in the classification scheme". This can be tested by attempting to set up a hierarchy with three levels.
  - N – The requirement cannot be tested formally. An example is "The ERMS must support the organisation's business classification scheme". There is no way to test this in the general case.
  - P – The requirement can be tested but the coverage of the test is partial, or it is not formally testable but it is possible that lack of compliance can be discovered. An example is "the ERMS should not limit the number of levels in the hierarchy." There is no way, formally, to test for the absence of a limit. However, the requirement is considered testable with partial coverage, for example by testing for a large number of levels; and during the testing it is possible that a limitation on the number of levels might be noticed, indicating that the ERMS does not comply with the requirement.

If chapters, sections or requirements are deleted, Microsoft Word will replace any cross-references to them (if there are any) with an error message. These can be located by searching for the text "**error!**"

By default, the table borders are not visible. They can be seen by use of the "Show Gridlines" command.

APPENDIX 4 – ACKNOWLEDGEMENTS

Project Team

Mr Frank Brady	European Commission (client)
Mr Tim Burrows	Serco Consulting
Mr Peter Campbell-Burns	Serco Consulting
Mr Keith Cornwell	Serco Consulting
Ms Alayne Crozier	Serco Consulting
Mr Steve Davies	Serco Consulting
Mr John Deverill	Serco Consulting
Mr Marc Fresko	Serco Consulting
Mr Michael Haimerl	Imbus AG (testing partner)
Mr Tilo Linz	Imbus AG (testing partner)
Mr Wasif Mehdi	Serco Consulting
Mr Thomas Rumi	Imbus AG (testing partner)
Mr Josephus Schram	European Commission (Project Officer)
Mr John Seeley	Serco Consulting
Ms Caroline Senior	Serco Consulting
Mr Michael Sill	Imbus AG (testing partner)
Ms Natasha Smith	Serco Consulting

The project team wishes to express its thanks to Mr John Worsfold of the Royal National Institute of Blind People (UK) for assisting with the requirements concerning accessibility.

Editorial Board

The Project Team was advised and guided by the following Editorial Board, constituted of international experts.

Mr Miguel Camacho	Sadiel S.A, Spain
Ms Marie-Anne Chabin	Archive 17, France
Ms Anne Mette Dørum	National Archives of Norway, Records Management Department, Norway
Professor Luciana Duranti	School of Library, Archival and Information Studies, University of British Columbia, Canada
Professor Mariella Guercio	University of Urbino, Italy
Mr Peter Horsman	Archiefschool (Netherlands Institute for Archival Education and Research), The Netherlands
Dr Ulrich Kampffmeyer	PROJECT CONSULT Unternehmensberatung GmbH, Germany
Mr Paul Murphy	Ministry of Finance, Ireland

DLM Forum

The DLM Forum provided a Review Group to review drafts on behalf of the European Commission.

Mr Richard Blake	UK
Ms Dolores Carnicer Arribas	Spain
Mr Olivier de Solan	France
Dr Andrea Hänger	Germany
Ms Paivi Happonen	Finland
Mr Toivo Jullinen	Estonia
Mr Göran Kristiansson	Sweden
Mr Ian MacFarlane	UK
Mr Atle Skjekkeland	Secretariat
Mr Jože Škofljanec	Slovenia
Mr Malcolm Todd (chair)	UK
Mr Martin Waldron	UK

Review panellists

The project team is extremely grateful to the following individuals and companies who kindly volunteered and devoted considerable amounts of time to participate in the review and validation exercise. Their valuable input into MoReq2 has ensured that the specification can address the needs of the widest community of users.

Mr Francisco Barbedo	Archives Panel	Instituto dos Arquivos Nacionais/Torre do Tombo, Portugal
Mr. Jan Dalsten Sørensen	Archives Panel	Danish National Archives
Ms Inta Feldmane	Archives Panel	National Archives of Malta
Mr Håkan Lövblad	Archives Panel	Riksarkivet/National Archives, Sweden
Mr. Michal Wanner	Archives Panel	Department of the Archives Administration and Records Management, Czech Republic
Mr Sergey Afanasiev	Specialists Panel	Records Managers Guild, Russia
Ms Phédra Clouner	Specialists Panel	document@work, Belgium
Mr Michiel Gen	Specialists Panel	ARMA International, Belgium
Ms Kimberley Barata	Users Panel	Parliamentary Archives, UK
Ms Kathy Bashaar	Users Panel	PNC Bank, USA
Mr Daniel J Beard	Users Panel	Xerox Corp, USA
Ms Alissa Burger	Users Panel	Rail Corp, Information and Records Management, Australia
Mr Barry Cahill	Users Panel	Nova Scotia Archives & Records Management Department of Tourism Culture & Heritage,Canada
Mr Lluís-Esteve Casellas Serra	Users Panel	Ajuntament de Girona. SGDAP, Spain
Mr Alejandro Delgado Gómez	Users Panel	Servicio de Archivo y Bibliotecas del Ayuntamiento de Cartagena Archivo Municipal parque de Artilleria, Spain
Mr Paul Dodgson	Users Panel	Leicestershire County Council, UK
Ms Susan Em	Users Panel	Royal Pharmaceutical Society of GB
Ms Trish Fallen	Users Panel	Information Management Practitioner, Australia

Ms Lucia Filimon Stefan	Users Panel	Joint Research Centre of the European Commission, Italy
Ms Fiorella Foscarini	Users Panel	European Central Bank, Germany
Ms Alison Gibney	Users Panel	Cimtech, UK
Mr Stefan Gradmann	Users Panel	Universität Hamburg, Germany
Ms Frances Grey	Users Panel	Parliamentary Archives, UK
Mr Harold C Heard Jr.	Users Panel	Citigroup, USA
Ms Sarah Higgins	Users Panel	Digital Curation Centre, University of Edinburgh, UK
Mrs Caroline Ives	Users Panel	Salford City Council, UK
Mr Philip Jones	Users Panel	Staffordshire County Council, UK
Mr Ben Kettell	Users Panel	Cactus Tecnologia, Spain
Ms Natasha Khramtsovsky	Users Panel	Electronic Office Systems, Russia
Mr Stewart Kirkup	Users Panel	Derbyshire County Council, UK
Mr Päivi Laakso	Users Panel	National Agency for Medicines, Finland
Ms Jessica Lila	Users Panel	USA
Mr Stephen Macintosh	Users Panel	Federal Court of Australia
Ms Sònia Oliveras Artau	Users Panel	Ajuntament de Girona. SGDAP, Spain
Mr Matt O'Mara	Users Panel	
Mr Adam Pope	Users Panel	Information Handy Man, UK
Ms Barbara Reed	Users Panel	Recordkeeping Innovation, Australia
Dr David Reeve	Users Panel	Dorset County Council, UK
Ms Maria Reixach Urcola	Users Panel	Ajuntament de Girona. SGDAP, Spain
Mr Jordi Serra Serra	Users Panel	Generalitat de Catalunya, Spain
Ms Deirdre Sharp	Users Panel	Norfolk County Council, UK
Mr Alan Shipman	Users Panel	Group 5 Training, UK
Ms Marija Šimunović	Users Panel	Supreme Court of the Republic of Croatia
Mr Sundeep Vaid	Users Panel	International Fund for Agricultural Development, Italy
Mr Peter Van Garderen	Users Panel	Artefactual Systems, Canada
Mr Willem Vannester	Users Panel	Stadsarchief Antwerpen, Belgium
Mr Gérard Weisz	Users Panel	Sirius Systems, France
Mr Martin Bartonitz	Vendors Panel	SAPERION, Germany
Mr Solomon Barron	Vendors Panel	IBM, UK
Mr Martin Bould	Vendors Panel	ErgoGroup AS, Norway
Mr Reynolds Cahoon	Vendors Panel	Lockheed Martin, USA
Mr Ian Capon	Vendors Panel	Open Text Corporation, UK
Mr Simon Cole	Vendors Panel	Meridio, UK
Mr Jon Garde	Vendors Panel	Objective Corporation, UK
Mr Graham Hadingham	Vendors Panel	FileNet, UK

Mr Joachim Haessler	Vendors Panel	Haessler Information, Germany
Ms Tamara Hoagland	Vendors Panel	EDRM Solutions, USA
Mr Mike Huberty	Vendors Panel	Lockheed Martin, UK
Mr Chris Hughes	Vendors Panel	Tower Software, UK
Dr Gregor Joeris	Vendors Panel	SER Solutions Deutschland, Germany
Mr Volker John	Vendors Panel	SAPERION, Germany
Dr Annegret Kampe	Vendors Panel	Docuware, Germany
Ms Mary Kelly	Vendors Panel	IBM, UK
Mr Andy King	Vendors Panel	Getronics, UK
Ms Karen McKenzie	Vendors Panel	UK Software
Mr Chris Palmer	Vendors Panel	CA, UK
Ms Shaheen Ramdiane	Vendors Panel	Open Text Corporation
Mr Miroslav Širl	Vendors Panel	ICZ, Czech Republic
Mr Andrew Snowden	Vendors Panel	Fujitsu, UK
Mr Dan Taillefer	Vendors Panel	EMC, Canada
Mr Nigel Wood	Vendors Panel	Fabasoft, UK

Trademarks

All trademarks appearing in this specification are acknowledged. Proprietary products are mentioned for illustrative purposes only; their inclusion does not represent any form of endorsement. Similarly, exclusion of other products implies no criticism of these products.

APPENDIX 5 – CORRESPONDENCE TO OTHER MODELS

This appendix summarises how the metadata model specified in appendix 9 can be related to:

- ISO 23081 – Metadata for records;
- ISO 15836 – The Dublin Core metadata element set.

ISO 23081– Metadata for Records

The entities considered in MoReq2 can be mapped approximately to their equivalents in ISO 23081 as follows:

MoReq2 entity	ISO 23081 entity sub-class
Component	-
Record	Item
	Transaction sequence
Volume	File/folder
Sub-file	
File	
Class	Series
Classification scheme	Archive
-	Archives

These mappings are necessarily approximate.

The metadata elements in the MoReq2 model each have a name consisting of two or three parts (as described in appendix 9.6). Wherever possible the second part of the name is taken from ISO 23081-2, but several have been developed for MoReq2, as shown in the following table:

ISO 23081 Metadata Group	2nd part of MoReq2 Element Name	Source of name
Identity	system_identifier	MoReq2
	system_identifier_rendition	MoReq2

ISO 23081 Metadata Group	2nd part of MoReq2 Element Name	Source of name
Description	abstract	ISO 23081
	author	MoReq2
	classification	ISO 23081
	copy_recipient	MoReq2
	counter_signature	MoReq2
	date	MoReq2
	external_identifier	MoReq2
	place	ISO 23081
	recipient	MoReq2
	sender	MoReq2
	title	ISO 23081
Event plan	abstract	MoReq2
	agent	ISO 23081
	date	ISO 23081
	event_description	ISO 23081
	event_trigger	ISO 23081
	period	MoReq2
	reminder	MoReq2
	status	MoReq2
	volume	MoReq2
Event history	abstract	MoReq2
	date	ISO 23081
	disposal_hold	MoReq2
	transfer_or_destroy	ISO 23081
	transferred_to	MoReq2
Use	administrator	MoReq2
	inactive	MoReq2
	language	ISO 23081
	status	MoReq2
	technical_environment	ISO 23081

ISO 23081 Metadata Group	2nd part of MoReq2 Element Name	Source of name
Relation	agent	MoReq2
	applies_to_agent	MoReq2
	applies_to_class	MoReq2
	cross_referenced_to	MoReq2
	disposal_hold	MoReq2
	entity_agent	MoReq2
	has_redaction	MoReq2
	has_role	MoReq2
	has_user	MoReq2
	is_child_of	MoReq2
	is_member_of	MoReq2
	is_redaction_of	MoReq2
	is_parent_of	MoReq2
	previous_fully_qualified_classification_code	MoReq2
	r&d_schedule	MoReq2
	record_type	MoReq2

Further aspects of the correspondence between MoReq2 and ISO 23081 are in appendix 9.

ISO 15836 – The Dublin Core metadata element set

The elements defined in the Dublin Core can be mapped to the elements in the MoReq2 model as follows. Where only a partial MoReq2 element name is shown, it indicates all elements that start with this partial name. So for example “Description.abstract” indicates all of the following:

- Description.abstract;
- Description.abstract.keywords;
- Description.abstract.reason_for_rendition.

Dublin Core Element	MoReq2 Element
contributor	Description.sender
coverage	-
creator	Description.author
date	Description.date
description	Description.abstract.description Description.external_identifier.internal_reference

Dublin Core Element	MoReq2 Element
format	Use.technical_environment.format Use.technical_environment.file_format
identifier	Identity
language	Use.language
publisher	-
relation	Relation
rights	-
source	-
subject	Description.abstract.keyword
title	Description.title
type	Description.record_type
-	Description.abstract.mandate Description.abstract.reason_for_rendition Description.copy_recipient Description.place.current_location Description.place.home_location Description.recipient Event_history Event_plan Use.status Use.technical_environment (save as above)

However, these mappings are necessarily approximate.

Further aspects of the correspondence between MoReq2 and ISO 23081 are in appendix 9.

APPENDIX 6 – DATE PROCESSING

The ERMS is required to process all dates correctly, regardless of millennium, century or other date representation issues. This appendix presents a statement of the requirement for year 2000 processing which could be adapted, if necessary, to deal with other dates. This will be especially relevant for Electronic Records Management Systems which may include metadata dates for previous or future centuries.

The following is reproduced verbatim, with permission, from BSI DISC PD2000-1:1998 A Definition of Year 2000 Conformity Requirements.

Year 2000 conformity shall mean that neither performance nor functionality is affected by dates prior to, during and after the year 2000.

In particular:

Rule 1 No value for current date will cause any interruption in operation.

Rule 2 Date-based functionality must behave consistently for dates prior to, during and after year 2000.

Rule 3 In all interfaces and data storage, the century in any date must be specified either explicitly or by unambiguous algorithms or inferencing rules.

Rule 4 Year 2000 must be recognized as a leap year.

APPENDIX 7 – STANDARDS AND OTHER GUIDELINES

7.1 Standards

This appendix lists standards and other resources referenced in the specification or applicable to electronic records management.

The standards include those that are particularly relevant to ERMSs; they omit generic standards such as those dealing with storage hardware and database languages.

The standards include international standards, both de jure and de facto. National standards are omitted from this list. They may be added to a chapter zero by the authority for a member state. Only standards that have a direct bearing on systems design are included; standards that address organisation and ongoing management are not included. In most cases, the short name of the standard (not the fully qualified name) is shown, for ease of understanding.

FIPS 186-2	NIST Digital Signature Standard (http://csrc.nist.gov/publications/PubsFIPS.html)
ISAAR(CPF)	International Standard Archival Authority Record for Corporate Bodies, Persons, and Families (International Council on Archives) (http://www.ica.org/en/node/30230)
ISAD(G)	International Standard for Archival Description (General). (http://www.icacds.org.uk/icacds.htm)
IETF RFC 2821	Simple Mail Transfer Protocol. http://www.ietf.org/rfc/rfc2821.txt)
IETF RFC 2822	Internet Message Format. (http://www.ietf.org/rfc/rfc2822.txt)
ISO 216	Writing paper and certain classes of printed matter – Trimmed sizes – A and B series
ISO 639	Codes for the representation of names of languages.
ISO 2788	Guidelines for the establishment and development of monolingual thesauri.
ISO 5964	Guidelines for the establishment and development of multilingual thesauri.
ISO 8601	Representation of dates and times.
ISO 9834-8	Procedures for the operation of OSI Registration Authorities: Generation and registration of Universally Unique Identifiers (UUIDs) and their use as ASN.1 Object Identifier components (see also ITU X.667).
ISO/TS 12033	Guidance for selection of document image compression methods.
ISO/TR 12037	Recommendations for the expungement of information recorded on write-once optical media.
ISO 12142	Media error monitoring and reporting techniques for verification of stored data on optical digital data disks.
ISO/TR 12654	Recommendations for the management of electronic recording systems for the recording of documents that may be required as evidence, on WORM optical disk.
ISO 14721	Open archival information system – Reference model (OAIS).
ISO/IEC 15444	JPEG 2000 image coding system: Core coding system.
ISO 15489	Records Management.

ISO/TR 15801	Information stored electronically – Recommendations for trustworthiness and reliability.
ISO 15836	The Dublin Core metadata element set.
ISO 18492/TR	Long-term preservation of electronic document-based information.
ISO 19005-1	Electronic document file format for long-term preservation – Part 1: Use of PDF 1.4 (PDF/A-1).
ISO 23081	Metadata for records.
ITU X.667	Generation and registration of Universally Unique Identifiers (UUIDs) and their use as ASN.1 object identifier components. (http://www.itu.int/ITU-T/studygroups/com17/oid/X.667-E.pdf).
TIFF	Tagged Image File Format. (http://partners.adobe.com/public/developer/tiff/index.html)
X.509	ITU-T Recommendation X.509: Open systems interconnection – The Directory: Public-key and attribute certificate frameworks. (http://www.itu.int/rec/T-REC-X.509-200003-I/en).
XKMS	XML Key Management Spec. (http://www.w3.org/TR/xkms/).
XML	W3C Extensible Markup Language (XML) (http://www.w3.org/TR/REC-xml/)

7.2 Other Guidance

ISO/DIS 9241-171	Ergonomics of human-system interaction – Part 171: Guidance on software accessibility
ISO/TS 16071	Guidance on accessibility for human-computer interfaces (due to be superseded by ISO 9241-171).
WfMC	Workflow Management Coalition Terminology & Glossary. (http://www.wfmc.org/standards/referencemodel.htm)
1999/93/EC	Directive on a Community Framework for Electronic Signatures. (http://europa.eu/scadplus/leg/en/lvb/124118.htm)
DLM Forum Guidelines	Guidelines on best practices for using electronic information. INSAR (European Archives News) Supplement III (1997). ISBN: 92-828-2285-0. (http://dlmforum.typepad.com/gdlines.pdf)

7.3 Accessibility Guidelines and Resources

This section lists guidelines and resources for developers and purchasers of ERMSs. Whereas other sections of this appendix contain only open and international publications, this section includes information that is originally national and that originates from the supplier community. This is because no internationally-accepted documentation has been identified; this may be added in any later editions of MoReq.

For developers
W3C Web Content Accessibility Guidelines (for websites and web applications) (http://www.w3.org/WAI/)
RNIB Web Access Centre (http://www.rnib.org.uk/webaccesscentre)

RNIB Software Access Centre (http://www.rnib.org.uk/softwareaccesscentre)
IBM Human Ability and Accessibility Centre (http://www-03.ibm.com/able/guidelines/)
ISO/IEC 18019 Guidelines for the design and preparation of user documentation for application software (see especially clause 4.2.6). (Due to be replaced by ISO/IEC 26514.)
ISO/IEC 26514 User documentation requirements for documentation designers and developers. (Under development).
For procurement
ACCENT – Accessibility in ICT Procurement: EU project (http://www.verva.se/english/international-network/the-accent-project/)
PAS 78:2006 A guide to good practice in commissioning accessible websites (http://www.equalityhumanrights.com/en/publicationsandresources/Disability/Pages/Websiteaccessibilityguidance.aspx)
RNIB Software Access Centre (http://www.rnib.org.uk/softwareaccesscentre)

7.4 Digital Preservation Guidelines

InterPARES project (http://www.interpares.org) Preserving Access to Digital Information (PADI) project National Library of Australia (http://www.nla.gov.au/padi/)
The National Archives Functional Requirements for the Sustainability of Electronic Records (http://www.nationalarchives.gov.uk/documents/functional_requirements.pdf)

7.5 Graphical Model of Relationship of MoReq2 with Other Guidance

This section contains a graphical model that shows how key standards are related to electronic records management. It uses a new model of electronic records management at figure A7.1, prepared only for this purpose.

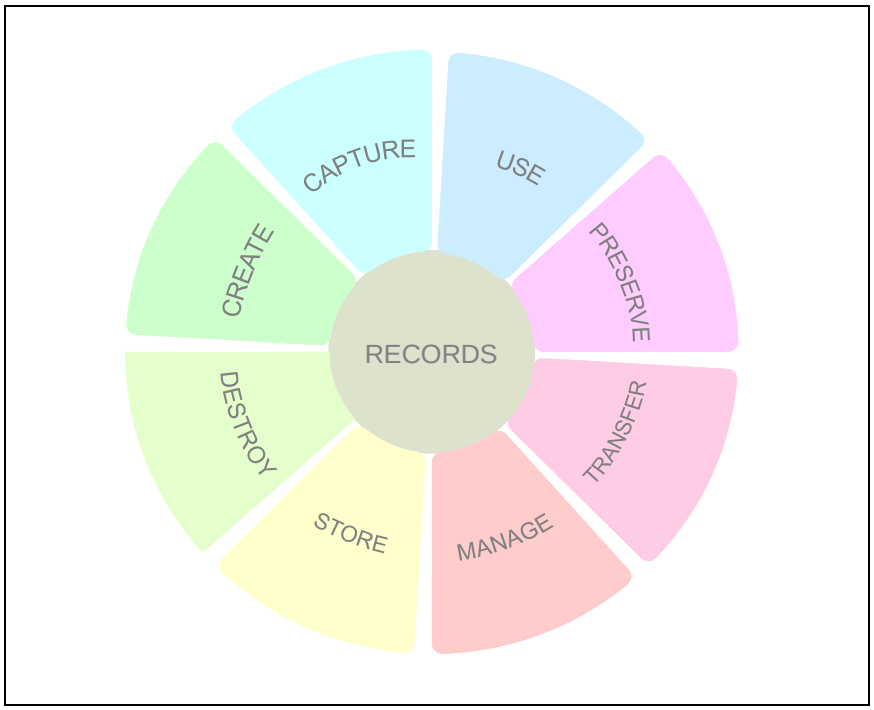

Figure A7.1

The model shows key processes that affect electronic records. The records are represented by the central, grey, circle. The processes ("create", "capture" etc.) are represented by the coloured shapes surrounding the records.

The number of processes shown (the granularity, or level of detail of the model) is somewhat arbitrary. Several other representations are possible, and would be more appropriate for different purposes; the above has been chosen specifically to relate to standards. To interpret these processes:

- **Create** includes not only the creation of records within an organisation, but also receipt of records from outside the organisation.
- **Capture** includes the registration, classification, and the entry of records management metadata.
- **Use** includes search, retrieve, browse, render, maintain, review etc.
- **Preserve** is the processes required to maintain accessibility over time.
- **Manage** includes maintaining access controls and disposition authorities.

The order of the processes shown is not significant, because they can occur in different sequences in different settings.

Simplifying greatly, the key standards applying to electronic records management can be related to these processes as shown in figure A7.2.

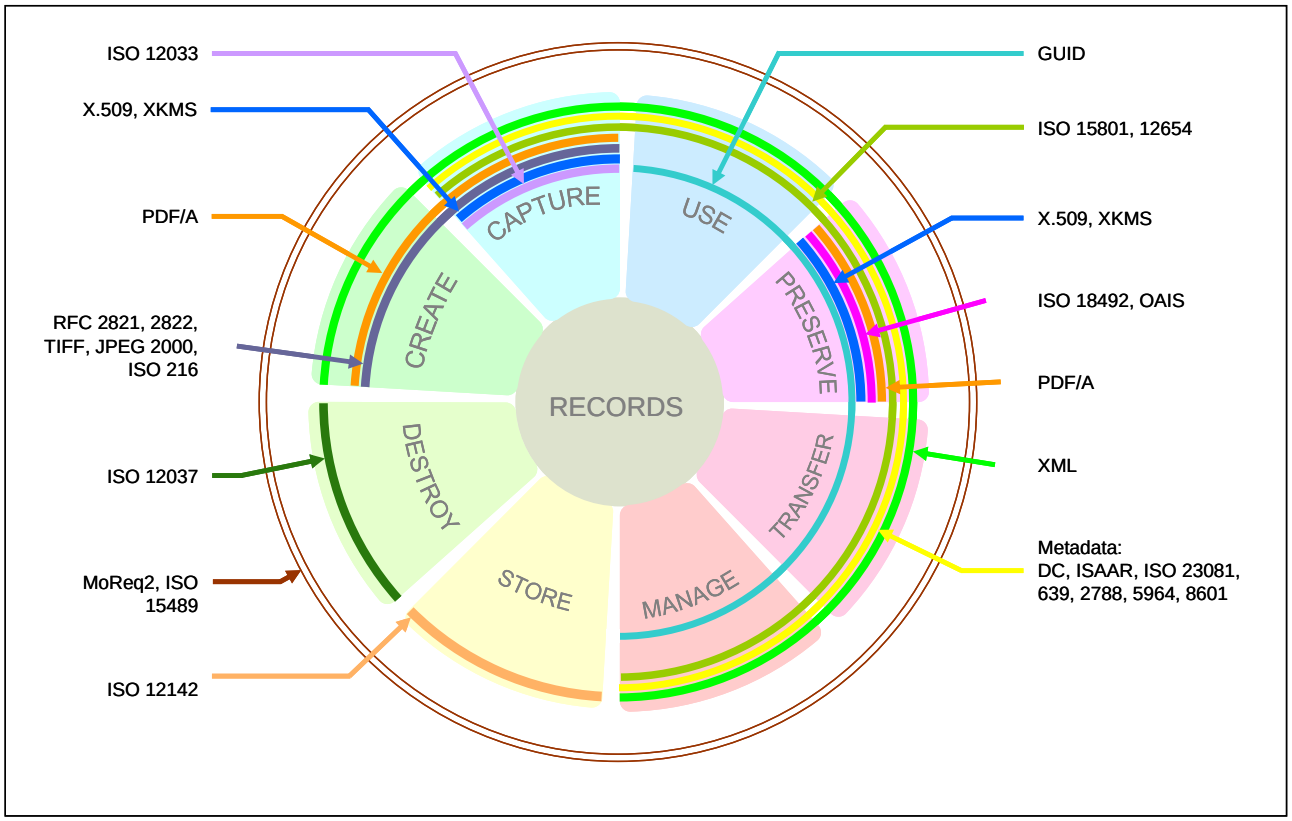

Figure A7.2

The relationship shown above between the standards and processes is repeated at the end of this section in a form that does not require colour.

This model traces the extent of the standards' relevance – very approximately – using coloured lines superimposed on the processes. Each coloured line represents one or several standards. Where possible, the standards are shown by their common name (e.g. PDF/A, OAIS) rather than by their less-descriptive standard number (e.g. ISO 19005, ISO 14721); refer to section 1 of this appendix for formal titles. Note that any model of this kind can only give a rough indication – it is not possible to show all details of all the interactions of processes and standards; to some extent the inclusions and omissions reflect a subjective view.

Only the standards that are considered as the most important are included in this diagram; others are excluded. The criteria for inclusion and exclusion are judgemental.

The model is explained below. Standards that are applicable across many processes are explained below, followed by the standards relevant to each of the processes.

ISO 15489 and MoReq2

ISO 15489 and MoReq2 both cover the entirety of the processes affecting electronic records. Accordingly, they are shown as encircling all the processes.

XML

XML is shown as being relevant to almost all processes – all except store and destroy. Its relevance varies greatly according to the environment. In principle however, it can influence the format of record creation then the way in which the metadata is stored and expressed at capture

and during later use; it is an important standard facilitating the interpretation of metadata and content in long term preservation; it can be used to provide a common schema for transfers between systems; it can be used to describe access and schemas and disposition authorities.

Metadata

Metadata standards are relevant to the processes of capture, use, preserve, transfer and manage. These include ISO 23081 (which covers all aspects of records management metadata), Dublin Core (which specifies a standard set of metadata for discovery), ISO 639 (controlled vocabulary for language codes), ISAD and ISAAR (approaches to the use of metadata for record keeping and archival description) and ISOs 2788 and 5984 (thesaurus standards).

Create

The major standards consideration in the process of records creation is the format of the record. Many format standards exist, including RFCs 2821/2822 (for e-mail), ISO 216, TIFF and JPEG (for scanned images), and PDF/A.

Capture

Metadata standards of all kinds apply strongly to the capture process. Some of the format standards affecting capture are also relevant from the point of view of extracting metadata values automatically. Standards affecting legal issues also apply to capture, namely ISO 15801 and ISO 12654.

Use

The standard governing GUIDs (globally unique ids), X.667, affects the way electronic records are used, as do standards relating to legal issues, ISO 15801 and ISO 12654.

Preserve

The key standard for digital preservation is OAIS; this provides a framework for the design and management of preservation activities. ISO 18492 also provides general guidance. Most preservation work relies greatly on the use of metadata standards; and a key standard is PDF/A, which describes a preservation format. Standards for electronic signatures, X.509 and XKMS also have a bearing on preservation issues.

Transfer

The use of metadata standards is essential to transfers between organisations or between systems.

Manage

Metadata standards can support the processes of managing access and retention. Also relevant are the legal standards, ISO 15801 and 12654.

Store

ISO 12142 addresses a small aspect of storage, related to storage on optical discs.

Destroy

ISO 12037 addresses a small aspect of destruction, namely expungement; this is only relevant is some environments.

The relationship between the standards and processes is shown below in a tabular form that does not require colour. This table shows the processes in columns, and the standards in rows. Where a tick (✓) is shown at the intersection of a row and a column it indicates that corresponding standard is related to the corresponding the process.

Standard		Create	Capture	Use	Preserve	Transfer	Manage	Store	Destroy
ISAAR(CPF)	International Standard Archival Authority Record for Corporate Bodies, Persons, and Families (International Council on Archives).		✓	✓	✓	✓	✓		
IETF RFC 2821	Simple Mail Transfer Protocol.	✓	✓						
IETF RFC 2822	Internet Message Format.	✓	✓						
ISO 216	Writing paper and certain classes of printed matter – Trimmed sizes – A and B series	✓	✓						
ISO 639	Codes for the representation of names of languages.		✓	✓	✓	✓	✓		
ISO 2788	Guidelines for the establishment and development of monolingual thesauri.		✓	✓	✓	✓	✓		
ISO 5964	Guidelines for the establishment and development of multilingual thesauri.		✓	✓	✓	✓	✓		
ISO 8601	Representation of dates and times.		✓	✓	✓	✓	✓		
ISO 9834-8	Generation and registration of Universally Unique Identifiers (UUIDs) and their use as ASN.1 Object Identifier components (see also ITU X.667).			✓			✓		
ISO 12033	Guidance for selection of document image compression methods.		✓						
ISO 12037	Recommendations for the expungement of information recorded on write-once optical media.								✓
ISO 12142	Media error monitoring and reporting techniques for verification of stored data on optical digital data disks.							✓	

Standard		Create	Capture	Use	Preserve	Transfer	Manage	Store	Destroy
ISO 12654	Recommendations for the management of electronic recording systems for the recording of documents that may be required as evidence, on WORM optical disk.		✓	✓			✓		
ISO 14721	Open archival information system – Reference model (OAIS).				✓				
ISO 15444	JPEG 2000 image coding system: Core coding system.	✓	✓						
ISO 15489	Records Management.	✓	✓	✓	✓	✓	✓	✓	✓
ISO 15801	Information stored electronically – Recommendations for trustworthiness and reliability.		✓	✓			✓		
ISO 15836	The Dublin Core metadata element set.		✓	✓	✓	✓	✓		
ISO 18492	Long-term preservation of electronic document-based information.				✓				
ISO 19005-1	Electronic document file format for long-term preservation.	✓	✓		✓				
ISO 23081	Metadata for records.		✓	✓	✓	✓	✓		
ITU X.667	Generation and registration of Universally Unique Identifiers (UUIDs) and their use as ASN.1 object identifier components.			✓			✓		
MoReq2	Update and extension of the Model Requirements for the management of electronic records	✓	✓	✓	✓	✓	✓	✓	✓
TIFF	Tagged Image File Format.	✓	✓						
X.509	ITU-T Recommendation X.509: Open systems interconnection – The Directory: Public-key and attribute certificate frameworks.		✓		✓				
XKMS	XML Key Management Specification.		✓		✓				
XML	W3C Extensible Markup Language.	✓	✓	✓	✓	✓	✓		

APPENDIX 8 – CHANGES FROM THE ORIGINAL MOREQ

8.1 Changes that are not Backwards-Compatible

MoReq2 has been written to ensure, as far as possible, compatibility with the original MoReq. The major innovation in this edition is the storage of records directly in classes, without the use of files. This is covered in section 3.2; it should be noted that the ERMS can be configured to disable this option.

The ability to create sub-files as well as volumes within files is also new to MoReq2 (see section 3.3). This does not raise any backwards compatibility issues, but is a major new requirement.

Also the definitions of presentation and rendition have reversed from the previous MoReq. See the glossary for a full definition of both terms.

8.2 Relationship between Sections

The structure of MoReq2 mirrors that of MoReq, subject to some changes and several additions. This section shows the correspondence between sections in MoReq and MoReq2.

MoReq		**MoReq2**	
Chap.	**Title**	**Chap.**	**Title**
	Preface		Preface: MoReq2
1	Introduction	1	Introduction
1.1	Background	1.1	Background
1.2	Purpose and Scope of this Specification	1.3	Purpose and Scope of this Specification
1.3	What is an ERMS?	1.4	What is an ERMS?
1.4	For What can this Specification be Used?	1.5	For what can this Specification be used?
1.5	Emphasis and Limitations of this Specification	1.7	Emphasis and Limitations of this Specification
1.6	Using this Specification	1.9	Customising this Specification
1.7	Organisation of this Specification	1.10	Organisation of this Specification
1.8	Mandatory and Desirable Requirements	1.12	Mandatory and Desirable Requirements
1.9	Intellectual Property	1.6	Intellectual property rights
2	Overview of ERMS Requirements	2	Overview of ERMS Requirements
2.1	Key Terminology	2.1	Key Terminology
2.2	Key Concepts	2.2	Key Concepts
2.3	Entity-Relationship Model	2.3	Entity-Relationship Model
3	Classification Scheme	3	Classification Scheme and File Organisation
3.1	Configuring the Classification Scheme	3.1	Configuring the Classification Scheme
3.2	Classes and Files	3.2	Classes and Files

MoReq		MoReq2	
Chap.	**Title**	**Chap.**	**Title**
3.3	Volumes	3.3	Volumes and Sub-Files
3.4	Maintaining the Classification Scheme	3.4	Maintaining the Classification Scheme
4	Controls and Security	4	Controls and Security
4.1	Access	4.1	Access
4.2	Audit trails	4.2	Audit trails
4.3	Backup and Recovery	4.3	Backup and Recovery
4.4	Tracking Record Movements		Deleted (covered in section 10.1)
4.5	Authenticity		Deleted (covered in chapter 2)
4.6	Security Categories	10.15	Security Categories
5	Retention and Disposal	5	Retention and Disposition
5.1	Retention Schedules	5.1	Retention and Disposition Schedules
5.2	Review	5.2	Review of Disposition Actions
5.3	Transfer, Export and Destruction	5.3	Transfer, Export and Destruction
6	Capturing Records	6	Capturing and Declaring Records
6.1	Capture	6.1	Capture
6.2	Bulk importing	6.2	Bulk Importing
6.3	Types of Document		Deleted (covered in section 6.1)
6.4	E-mail Management	6.3	e-Mail Management
7	Referencing	7	Referencing
8	Searching, Retrieval and Rendering	8	Searching, Retrieval and Presentation
8.1	Search and Retrieval	8.1	Search and Retrieval
8.2	Rendering: Displaying Records	8.2	Presentation: Displaying Records
8.3	Rendering: Printing	8.3	Presentation: Printing
8.4	Rendering: Other	8.4	Presentation: Other
9	Administrative Functions	9	Administrative Functions
9.1	General Administration	9.1	General Administration
9.2	Reporting	9.2	Reporting
9.3	Changing, Deleting and Redacting Records	9.3	Changing, Deleting and Redacting Records
10	Other Functionality	10	Optional Modules
10.1	Management of Non-electronic Records	10.1	Management of Physical (Non-electronic) Files and Records
10.2	Hybrid File Retention and Disposal	10.2	Disposition of Physical Records
10.3	Document Management	10.3	Document Management and Collaborative Working
10.4	Workflow	10.4	Workflow

MoReq		MoReq2	
Chap.	**Title**	**Chap.**	**Title**
10.5	Digital Signatures	10.7	Electronic Signatures
10.6	Encryption	10.8	Encryption
10.7	Electronic Watermarks etc.	10.9	Digital Rights Management
10.8	Interoperability and Openness		Deleted (covered in chapters 5, 6 and 10.6)
11	Non-Functional Requirements	11	Non-Functional Requirements
11.1	Ease of Use	11.1	Ease of Use
11.2	Performance and Scalability	11.2	Performance and Scalability
11.3	System Availability	11.3	System Availability
11.4	Technical Standards	11.4	Technical Standards
11.5	Legislative and Regulatory Requirements	11.5	Legislative and Regulatory Requirements
11.6	Outsourcing and Third Party Management of Data	11.6	Outsourcing and Third Party Management of Data
11.7	Long Term Preservation and Technology Obsolescence	11.7	Long Term Preservation and Technology Obsolescence
12	Metadata Requirements	12	Metadata Requirements
12.1	Principles	12.1	Principles
12.2	Organisation of the Remainder of this Chapter	App. 9.1	Introduction
12.3	Classification Scheme Metadata Elements	App. 9.7.1	Classification Schemes
12.4	Class and Electronic File Metadata Elements	App. 9.7.2	Classes, Files, Sub-Files, Volumes
12.5	Metadata Elements for Electronic File or Electronic File Volume	App. 9.7.2	Classes, Files, Sub-Files, Volumes
12.6	Electronic Volume Metadata Elements	App. 9.7.2	Classes, Files, Sub-Files, Volumes
12.7	Record Metadata Elements	App. 9.7.2	Classes, Files, Sub-Files, Volumes
12.8	Record Extract Metadata Elements	App. 9.7.3	Record Redactions
12.9	User Metadata Elements	App. 9.7.8	Agents (Users, Groups and Roles)
12.10	Role Metadata Elements	App. 9.7.8	Agents (Users, Groups and Roles)
12.11	Customisation Notes for Metadata Requirements	App. 9.9	Customisation Notes for Metadata Requirements
13	Reference Model	13	Reference Model
13.1	Glossary	13.1	Glossary

MoReq		MoReq2	
Chap.	**Title**	**Chap.**	**Title**
13.2	Entity-Relationship Model	13.2	Entity-Relationship Model
13.3	Entity-Relationship Diagram Narrative	13.3	Entity-Relationship Narrative
13.4	Access Control Model	13.4	Access Control Model
	ANNEXES		APPENDICES
Ann. 1	Reference Publications	App. 1	Reference Publications
Ann. 2	Development of this Specification	App. 2	Development of this Specification
Ann. 3	Use of this Specification in Electronic Form	App. 3	Use of this Specification in Electronic Form
Ann. 4	Acknowledgements	App. 4	Acknowledgements
1	Project Team	App. 4.1 App. 4.2	Project Team Editorial Board
2	Validation Organisations	App. 4.3 App. 4.4	DLM Forum Review Panellists
3	Trademarks	App. 4.5	Trademarks
Ann. 5	Correspondence to Other Models	App. 5	Correspondence to Other Models
1	Correspondence to Dublin Core Metadata Model	App. 5.2	ISO15836 – The Dublin Core metadata standard
2	Correspondence to Pittsburgh metadata model	-	-
Ann. 6	Date Processing	App. 6	Date Processing
Ann. 7	Standards and Other Guidelines	App. 7	Standards and Other Guidelines
1	Standards	App. 7.1	Standards
2	Other Guidelines	App. 7.2	Other Guidance
3	Accessibility Guidelines	App. 7.3	Accessibility Guidelines and resources
4	Long Term Preservation Guidelines	App. 7.4	Digital Preservation Guidelines

APPENDIX 9 – METADATA MODEL

Appendix 9 contains the MoReq2 metadata model. Because of its length, and to ease cross referencing, it is published in electronic form only, at www.dlm-network.org/moreq2 .