

Test Script TS-X03

Scope:

Metadata

Export (via Disposal)

Bulk Import

Version 1.0
Consultation Draft
XML Schema Project

February 2008
The National Archives

The National Archives
Kew
Richmond
Surrey
TW9 4DU
United Kingdom

email: rmadvisory@nationalarchives.gov.uk

Website: www.nationalarchives.gov.uk/electronicrecords

© Crown copyright 2008

Test Script TS-X03

Contents

Test Script TS-X03 Preparation	5
1.1. Coverage	5
1.2. Assumptions	5
1.3. Checksheets and worksheets	6
1.3.1. Before running the Test Scripts	6
1.3.2. During the Test Script	8
1.4. Test preconditions	9
1.5. Hierarchy model - Start	10
1.6. Metadata values - Start	10
1.7. Test data	11
1.7.1. Test data - Format	12
1.8. Individual user profiles	13
1.9. Access control groups	13
1.10. Disposal schedules	14
1.11. General Test Script notes	15
1.11.1. Fileplan hierarchy	15
1.11.2. Metadata shell	15
1.11.3. Date and time	15
1.11.4. Inheritance	16
1.11.5. Disposal	16
1.11.6. Language	16
1.12. XML files	17
1.12.1. Historic metadata	18
1.12.2. Custom metadata	18
1.13. Core metadata on export	21
1.13.1. 1. Identifier	21
1.13.2. 2. Title	21
1.13.3. 3. Subject	21
1.13.4. 4. Description	21
1.13.5. 5. Creator	21
1.13.6. 6. Date	21
1.13.7. 7. Correspondence Data	22
1.13.8. 8. Type	22
1.13.9. 9. Relation	22
1.13.10. 10. Aggregation	22
1.13.11. 11. Language	22
1.13.12. 12. Location	22
1.13.13. 13. Rights	23
1.13.14. 14. Disposal	23
1.13.15. 15. Digital Signature	24
1.13.16. 16. Preservation	24
1.13.17. 17. Mandate	24
1.13.18. 18. Format	24
1.13.19. 19. Function	24
1.13.20. 20. Coverage	24
1.14. Core metadata on bulk import	25
1.14.1. 1. Identifier	25

1.14.2.	2. Title	25
1.14.3.	3. Subject	26
1.14.4.	4. Description	26
1.14.5.	5. Creator	26
1.14.6.	6. Date.....	26
1.14.7.	7. Correspondence Data	27
1.14.8.	8. Type	27
1.14.9.	9. Relation	28
1.14.10.	10. Aggregation	28
1.14.11.	11. Language.....	28
1.14.12.	12. Location	29
1.14.13.	13. Rights.....	29
1.14.14.	14. Disposal	30
1.14.15.	15. Digital Signature	30
1.14.16.	16. Preservation.....	30
1.14.17.	17. Mandate.....	31
1.14.18.	18. Format	31
1.14.19.	19. Function	31
1.14.20.	20. Coverage	31
2.	Test Script TS-XS03 Steps.....	32
2.1.	Test execution formal details - Start	32
2.2.	Test steps.....	33
2.3.	Test execution formal details - End	61
2.4.	Hierarchy model - End.....	62
2.5.	Metadata values - End.....	63

Test Script TS-X03 Preparation

1.1. Coverage

This Test Script:

- checks metadata values
- checks for metadata inheritance where appropriate
- executes a disposal schedule that implements an export action
- executes a bulk import action

This main focus of this Test Script is the export and import of fileplan object metadata and record content.

All of the steps in this Test Script should be performed in a single day - i.e. the system clock should not be changed, and will not cross over into another day.

1.2. Assumptions

It is assumed that the tester(s)

- will be familiar with the ERMS functionality (or will be given appropriate guidance as required)
- will be familiar with the fileplan hierarchy model supported by both the source ERMS and the destination ERMS (or will be given appropriate guidance as required)
- will be familiar with metadata management within the ERMS
- will provide test comments in the text box provided at the bottom of each page as and when required.

Test Comments

1.3. Checksheets and worksheets

Various metadata checksheets and worksheets should be completed and referenced as appropriate throughout the testing process:

- Checksheets to be completed by the tester(s) before Test Scripts are run.
- Checksheets to be completed by the tester(s) while a Test Script is being run.
- Worksheets to be referenced by the tester(s) while a Test Script is being run.

Checksheets and worksheets are available in the following files:

- Pre-Test Script Metadata Mapping Checksheets.xls
- Test Script TS-X01 Checksheets and Worksheets.xls

It is strongly recommended that the checksheets will be completed by directly editing the spreadsheet file within MS Excel or compatible software application. Of course, there is nothing to prevent testers completing the checksheets in hard copy, but it is likely to be more beneficial to retain the details in electronic format.

1.3.1. *Before running the Test Scripts*

Before the running any of the Test Scripts, there is a general assumption that the tester(s) will be familiar with the ERMS functionality and also have an awareness of metadata management within the ERMS environment.

Testers will be aware that many of the metadata labels displayed via the user interface, and also at the underlying database level, will often be similar to the core metadata element and sub-elements names. However, there may also be instances where the field names are quite different. In order to avoid confusion at any point of the test process, checksheets are provided so that field labels (and database fields if necessary) for the source and destination ERMS under test can be clearly identified by the tester(s) alongside their corresponding core metadata names.

This approach ensures that the tester knows exactly which ERMS fields correspond to the core metadata, and will also identify any core fields that are not currently supported by the ERMS. Furthermore, the information held in these checksheets may aid the resolution of metadata issues that may arise when later running the Test Scripts.

In addition to the checksheets for core metadata, there are also checksheets for custom metadata. While the main focus of the Phase 1 testing is core metadata, it is acknowledged that some systems may automatically include system-specific custom metadata for one or more fileplan object, which is included when fileplan objects are exported. In order to ensure that the tester is aware of all export and import metadata fields, custom metadata details should be provided where applicable.

Test Comments

1.3.2. ***During the Test Script***

Throughout this Test Script, the tester(s) should take note of the actual metadata values in either "TS-X01 Checksheet 1A" or "TS-X01 Checksheet 1B" (the only difference is the preferred sort order of the metadata fields).

The corresponding worksheets ("TS-X01 Worksheet 1A" or "TS-X01 Worksheet 1B") contain example metadata values, as specified in the test steps of the Script. As such, the worksheets effectively provide a baseline reference point for the expected core metadata values at the end of the Test Script.

Many of the actual metadata values will vary from the example metadata values but there should also be an acceptable similarity in evidence.

For example:

- datestamp values should present the same date (e.g. 2008-02-01), but may apply different times depending on when the test step was completed
- ERMS terms for aggregation levels may not be an exact match for "class", "folder", "part", "record" and "component"; instead they may be similar strings, such as "Class object", "Folder object", "Part object", "Record object" and "Component object".

By monitoring and logging the actual metadata values in one of the checksheets during the Test Script, testers will be able to readily compare the actual values with the example values in the last step of the Test Script. This comparison will allow testers to check factors such as, whether or not :

- metadata has been populated in the expected manner during the Test Script
- actual metadata values match or demonstrate an acceptable similarity to the corresponding example metadata values
- additional data conversion functionality may be required from the ERMS for export and import actions.

In order to facilitate the completion of the checksheets, some common, fixed values have already been entered. If the actual values do not match these pre-entered values, the tester should update the checksheet accordingly and an appropriate note should be made.

Test Comments

The file "Test Script TS-X01 Checksheets and Worksheets.xls" includes 2 checksheets and 2 worksheets:

Checksheet	Details
TS-X01 Checksheet 1A	At least one of these checksheets should be completed by the tester(s) as Test Script TS-X01 is run.
TS-X01 Checksheet 1B	
<i>Note: The only difference between the two checksheets is the preferred sort order of metadata fields.</i>	

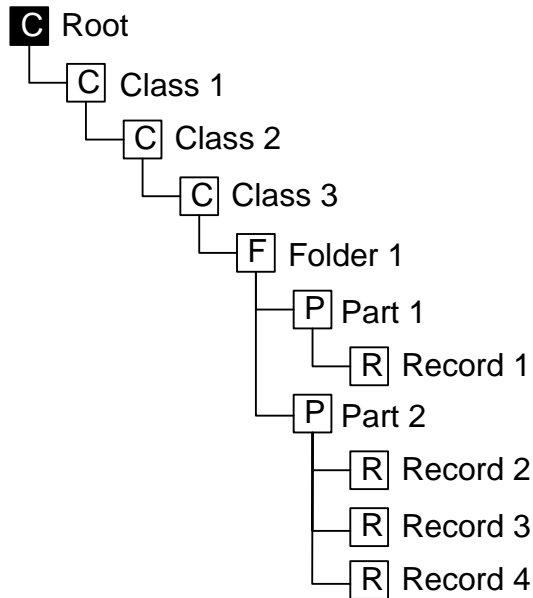
Worksheet	Details
TS-X01 Worksheet 1A	At least one of these worksheets should be referenced by the tester(s) in Test Script TS-X01.
TS-X01 Worksheet 1B	
<i>Note: The only difference between the two worksheets is the preferred sort order of metadata fields.</i>	

1.4. Test preconditions

No.	Precondition
1	The ERMS must be able to support the complete revised eGMS core metadata shell. All record_types used in this Test Script must provide the full metadata shell.
2	Successful completion of Test Script TS-X02.
3	Must be able to change the system date and time during the test period.

Test Comments	
----------------------	--

1.5. Hierarchy model - Start



1.6. Metadata values - Start

Example metadata values that match the end of Test Script TS-X02, and therefore the start of Test Script TS-X03, are provided in the worksheets in the file: Test Script TS-X02 Checksheets and Worksheets.xls". This file also contains template checksheets, which the tester(s) should have completed with actual metadata values throughout Test Script TS-X02.

Test Comments

1.7. Test data

The following record content file is involved in this Test Script:

Record	Source Filename	Date Created	Creator
Record 1	Record 1.doc	2008-01-31T14:59:14	Ramona Black
Record 2	Record 2.rtf	2008-01-31T15:12:48	Ramona Black
Record 3	Record 3.xls	2008-01-31T15:25:52	Ramona Black
Record 4	<i>Record 4 content.rtf</i>	<i>Variable</i>	<i>Variable</i>

Record content source files are held in the file: "Test Case A Record Content Files.zip".

"Record 1" was declared in Test Script TS-X01.

"Record 2", "Record 3" and "Record 4" were all declared in Test Script TS-X02.

Test Comments

1.7.1. Test data - Format

Record	Source Filename	Format			
		Application	Extent	Medium	Operating System
Record 1	Record 1.doc	Microsoft Office Word 2003	19,456 bytes	CD-ROM	Microsoft Windows XP
Record 2	Record 2.rtf	Microsoft Office Word 2003	253 bytes *	CD-ROM	Microsoft Windows XP
Record 3	Record 3.xls	Microsoft Office Excel 2003	13,824 bytes	Hard Drive	Microsoft Windows XP
<i>Record 4 **</i>	<i>Record 4 content.rtf</i>	<i>Microsoft Outlook 2000</i>	<i>524,244 bytes</i>	<i>Email server</i>	<i>Microsoft Windows XP</i>

* The custom Extent value for Record 2 was entered as "253 bytes", however, the size of the file is 3,193 bytes. While originally a typing error, this allows the tester to readily establish whether the actual value has been determined automatically or taken from the custom properties.

** It must be emphasised that the Format metadata for the email Record 4 is strictly for example purposes only. It is well recognised that the actual metadata values are likely to vary for emails managed by different applications, with potentially different storage formats. In particular, the file size in the Extent sub-element is simply an example value and is not expected to be an exact match to the actual metadata value observed on test.

Record content files for "Record 1", "Record 2" and "Record 3" have been configured in such a way that custom property fields hold the specified values for Application, Extent, Medium and Operating System.

Test Comments	
----------------------	--

1.8. Individual user profiles

The following users are involved in this Test Script:

Name	Password	Role	Protective Marking	Descriptor - Informative	Descriptor - Functional
RecMgr	RecMgr\$	ERMS Records Manager	TBC	TBC	TBC
SysAdmin	SysAdmin\$	ERMS System Administrator	TBC	TBC	TBC
User 1	User1\$	ERMS Typical End User	TBC	TBC	TBC

Note: The ERMS System Administrator may be given "super user" permissions, allowing full access rights to the ERMS fileplan.

1.9. Access control groups

The following access control groups are involved in this Test Script:

Name	Members	Protective Marking	Descriptor - Informative	Descriptor - Functional
Team A	RecMgr SysAdmin * User 1	TBC	TBC	TBC

* Some systems may automatically add SysAdmin to every access control group by default so specific allocation may not be required.

Test Comments	
----------------------	--

1.10. Disposal schedules

Some systems require a disposal schedule to be applied upon object creation. Therefore the following global disposal schedules definitions are actively involved in this Test Script:

ID	Name	Disposal Action	Disposal Event	Retention Period
DSR1	Schedule R1 - Global class review	review	Class opened	immediate
DSR1a	Schedule R1 - Global folder review	review	Folder opened	immediate

It is expected that disposal schedule entitled "Schedule R1 - Global class review" will be allocated to a high level class and then inherited by default down through the entire test fileplan hierarchy.

If the ERMS is not capable of basing disposal schedules on class events, the disposal schedule entitled "Schedule R1 - Global folder review" (which uses a folder event) should be allocated instead of "Schedule R1 - Global class review".

In addition, the following export disposal schedules are actively involved in this Test Script:

ID	Name	Disposal Action	Disposal Event	Retention Period
DSE2	Schedule E2 - Export folder 1 month after folder closed	export	Folder closed	0 years 1 month
DSE4	Schedule E4 - Export record 1 month after record declaration	export	Record declared	0 years 1 month

It is anticipated that default record_type records will implicitly inherit the disposal settings from the parent hierarchy rather than via the record_type configuration.

In order to avoid potential record_type definition management issues (in particular, where systems may not support the retrospective application of updated disposal schedules), the export disposal schedule entitled "Schedule E4 - Export record 1 month after record declaration" will be immediately allocated to any non-default record_type record in this Test Script (as opposed to a review as per "Schedule R1 - Global class review" and "Schedule R1 - Global folder review" for container objects).

Test Comments

It is envisaged that disposal schedule "Schedule E4 - Export record 1 month after record declaration" will be only used if the email record "Record 4" is not declared using the standard default record_type (see Test Script TS-X02, step 210). However, if the system under test requires all record_types to have a specific disposal schedule allocation, "Schedule E4 - Export record 1 month after record declaration" should be applied.

1.11. General Test Script notes

1.11.1. Fileplan hierarchy

The Test Script generally assumes that the ERMS is capable of supporting the fileplan model presented in TNA's 2002 functional requirements - i.e.

- a class may contain one or more child classes OR one or more folders
- a folder is segmented into one or more parts
- a record is declared into the most recent part of a folder

Where the ERMS deviates from this model, the test steps may still be run but users must actively monitor any variations and understand that additional manual effort may be required during export and import processes.

In particular if parts are not supported, when a folder is exported it may be feasible to export a "virtual" part, where the majority of metadata values are copied from the "parent" folder, and the part title is given the default value "Part 1". This approach may make import into another system much simpler.

1.11.2. Metadata shell

Source and destination systems may support different metadata shells - i.e. the metadata fields used by one system may be different to another. It is recommended that the complete core metadata set is always present. If a core metadata element is related to a non-mandatory function that is not available within the system, it is recommended that the field still be present in readiness to hold potential imported data values for information purposes.

If a system under test does not support the complete metadata shell, appropriate test notes should be taken and the impact on the test as a whole should be monitored.

Also see section [1.3 Checksheets and worksheets](#).

1.11.3. Date and time

Metadata fields for dates are expected to hold both date and time details. While it may be the case that only the date is displayed at the user interface, the full date and time details will normally be recorded at the underlying database level and held within the audit log.

Test Comments

All the actual date values (e.g. 2008-02-01) should be as directed by the Test Script. The actual time values associated with Test Script actions may vary depending on the actual time each action is performed, with the exception of the "Date.Created" metadata. The full datestamp associated with "Date.Created" record metadata is fixed and should match the value given in the Test Script.

There may be occasions when the user is required to manually enter date and time metadata. Date details should be applied as directed in the test step. Where the user is able to enter time details, the hour and minutes should closely match the current time, while the seconds may be set as "00" unless otherwise directed.

The way in which the date fields are managed may vary between systems. However, it must be possible to export and import values in the exact form: YYYY-MM-DDThh:mm:ss.

Upon export, it may be feasible to automatically include a static time of 00:00:00 in order to provide the expected data type.

When an exported XML file is validated, all date metadata values are expected to follow the YYYY-MM-DDThh:mm:ss format. Values that are not in this format will return a validation error, and additional manual effort may be required before later importing the data.

1.11.4. Inheritance

By default, there is an expectation that all child objects held at lower levels of the hierarchy line will inherit many of the metadata field values either upon creation or retrospectively.

1.11.5. Disposal

Disposal schedules are often allocated at high levels of the fileplan and then inherited down in order to quickly and efficiently manage a fileplan and its contents. This is how the Test Script expects the ERMS to manage disposal schedules.

It is, however, acknowledged that some systems require specific disposal allocations and / or have limited inheritance rules (particularly for retrospective inheritance). Where this is the case appropriate notes should be made in the Test Comments, so that potential impact on disposal management can be properly considered.

1.11.6. Language

The way in which the language options are displayed at the ERMS user interface may vary. However, it must be possible to export and import values using the format presented by ISO 639. For example, English is represented by the value: ISO 639-2/B [Eng].

Test Comments

1.12. XML files

When fileplan objects are exported, it is expected that the metadata exported in this Test Script will be held within a **single XML file** (with the possible exception of "Record 4", as explained in the Test Script steps.).

It is acknowledged that there may be instances where there is a need to export separate XML files (e.g. one for each folder, part and record, etc.), however, at this first stage in the assessment process, the ERMS must demonstrate the ability to export all fileplan object metadata within one file.

Furthermore, it is acknowledged that some systems may allow authorised users to configure which metadata is exported - e.g. for administrative tasks. Where this is the case, it is expected that the configuration under test will export the full metadata set.

In order to demonstrate proof of concept, every metadata field for each exported fileplan object is expected to be present in the single exported XML file. Empty metadata fields should be included in the XML file within standard XML comment tags.

For example an empty "Subject" field would be represented within an exported XML file as either:

<pre><!-- <Subject /> --></pre>
or
<pre><!-- <Subject></Subject> --></pre>

Each exported XML file will support metadata profiles for classes, folders, parts, records and components, with the metadata presented in a recognisable, structured hierarchy, with each metadata element or sub-element in the core metadata shell mapped to an intuitively named XML field. Additional XML block tags are used to aid the data presentation and management.

The structural field layout and field values within the exported XML file are validated and governed by the revised XML Schema model.

It should be noted that where fields expect one value from a pre-defined list (e.g. the list of possible Protective Marking values: "Unclassified", "Restricted", "Confidential", "Secret" and "Top Secret"), the validation process is very strict. Only exact match values will be accepted; the use of different letter cases and even the inclusion of extra space characters will result in a validation error message.

Test Comments

1.12.1. Historic metadata

It should be noted that while the ERMS is expected to support the complete core metadata shell, not all of the core metadata fields have a 1:1 mapping to a field tag in the exported XML file. A number of fields hold metadata that is only active and relevant within the source ERMS; when this metadata is migrated to another ERMS, the values are generally useful only for reference or historic purposes.

The use of "active" and "historic" metadata ensures that the values are always controlled in a standard manner, without the potential loss of data or the need to create different custom fields for each system involved in the data migration.

Core metadata checksheets in the file: "Pre-Test Script Metadata Mapping Checksheets.xls" provide mapped metadata columns for the core metadata and exported XML field tags.

Further information and examples of "historic" metadata is provided in the "Export and Import Manual" that accompanies this body of work.

1.12.2. Custom metadata

Custom metadata is any metadata that is not part of the core metadata set. While Test Scripts TS-X01, TS-X02 and TS-X03 only focus on the core metadata, it is acknowledged that some systems may apply other custom metadata to one or more fileplan objects.

Test Comments

As all fileplan object metadata should be exported, where custom metadata is used, it should be held within the "Dataset" XML block tag.

For example:

```
<?xml version="1.0" encoding="UTF-8"?>
<Folder SchemaVersion="3" xsi:schemaLocation="http://www.govtalk.gov.uk/CM/
gms ERMS-folder-v31.xsd" xmlns:gms="http://www.govtalk.gov.uk/CM/gms"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
</Folder>

<Aggregation>Folder</Aggregation>

...

<Dataset>
  <CustomField>Custom metadata field value here.</CustomField>
</Dataset>

...

<Subject>XML Schema</Subject>
</Folder>
```

Intuitive names should be given to each custom XML field tag; these details should be entered into the "Custom Metadata - Source" checksheet in the file: "Pre-Test Script Metadata Mapping Checksheets.xls".

If different source and destination systems are being used, details should also be entered into the "Custom Metadata - Destination" checksheet.

Test Comments

At present, the management of exported custom metadata has not been thoroughly investigated. However, within the scope of this Test Script, the following assumptions should be made about custom metadata fields:

Property	Details
Data type	Text
Field length *	256 characters
Obligation status	Optional
Repeatability	0, 1 or many

* Guidance on core metadata field length will be covered in the further work of Phase 1 of the XML Schema project.

Further guidance on custom metadata will be further developed in Phase 2 of the XML Schema project.

Test Comments

1.13. Core metadata on export

This section provides brief guidance and expectations for the export of the core metadata elements.

1.13.1. 1. Identifier

System ID and Fileplan ID values are exported to corresponding historic metadata fields. The Case ID value in the source ERMS will match the value in the XML file.

1.13.2. 2. Title

The Title value in the source ERMS will match the value in the XML file.

1.13.3. 3. Subject

The Subject value in the source ERMS will match the value in the XML file.

1.13.4. 4. Description

The Description value in the source ERMS will match the value in the XML file.

1.13.5. 5. Creator

The Creator value in the source ERMS will match the value in the XML file.

1.13.6. 6. Date

The Date metadata values in the source ERMS will match or reflect the values in the XML file.

While the date and time details may be presented differently within the source ERMS, this should not impact on the actual value format as expected by the XML Schema - i.e. YYYY-MM-DDThh:mm:ss.

When records are exported, it may often be the case that the record content file (e.g. Record 1.doc) is simply copied to the tester's desktop. This action results in a new copy of the record content file, which also means a new Date.Created timestamp value for the new file.

This could lead to metadata conflicts and potential record integrity issues. Therefore testers should pay particular note of what happens with respect to record and component metadata and the record content during the export process.

TNA will provide further guidance on the general expectations at a later stage.

Test Comments

1.13.7. 7. Correspondence Data

The email transmission metadata values in the source ERMS will match or reflect the values in the XML file.

While the date and time details may be presented differently within the source ERMS, this should not impact on the actual value format as expected by the XML Schema - i.e. YYYY-MM-DDThh:mm:ss.

1.13.8. 8. Type

The class_type, folder_type, part_type and record_type values in the source ERMS will match or reflect the values in the XML file.

Note: The core metadata set only includes the name of the Type, not the full definition. This issue and more will be covered during Phase 3 of this project.

1.13.9. 9. Relation

When relational metadata is exported, many of the active values become historic. Core metadata checksheets in the file: "Pre-Test Script Metadata Mapping Checksheets.xls" provide mapped metadata columns for the core metadata and exported XML field tags. These checksheets can be used to readily identify which active metadata fields in the source ERMS map to historic fields in the exported XML file.

1.13.10. 10. Aggregation

The Aggregation values in the source ERMS will match or reflect the values in the XML file. While the Aggregation details may be presented differently within the source ERMS, this should not impact on the actual value format as expected by the XML schema - i.e. "class", "folder", "part", "record" and "component".

1.13.11. 11. Language

The Language value in the source ERMS will match or reflect the value in the XML file. While the Language details may be presented differently within the destination ERMS, this should not impact on the actual value format as expected by the XML Schema - e.g. "ISO-639-2 [Eng].

1.13.12. 12. Location

The Location values in the destination ERMS will match the values in the XML file.

Test Comments

1.13.13. 13. Rights

The Rights values in the source ERMS will match or reflect the values in the XML file. While some of the Rights details (e.g. Protective Marking, Protective Marking Change Date, FoI Disclosability Indicator, etc.) may be presented differently within the source ERMS, this should not impact on the actual value format as expected by the XML Schema.

For example, it is expected that the Protective Marking metadata field will hold 1 of 5 possible values: "Unclassified" (default), "Restricted", "Confidential", "Secret" and "Top Secret". Similarly, it is expected that the "FoI Disclosability Indicator" metadata field will hold 1 of 3 possible values: "not specified" (default), "no" and "yes". Also, date and time details should be exported in the format: YYYY-MM-DDThh:mm:ss.

Descriptor (Informative and Functional) metadata values in the source ERMS will match the value in the XML file.

Some systems support Functional Descriptors using the workaround of applying access control groups rather than a specific descriptor field. Where this is the case, in order to standardise the use of the descriptor metadata, these access control details should be automatically exported into the "FunctionalDescriptor" field tag of the XML file wherever possible.

Note: The core metadata set only includes the name of individual users and access control groups, not the full user profiles or group membership information. This issue and more will be covered during Phase 3 of this project.

1.13.14. 14. Disposal

When disposal metadata is exported, all of the active values become historic. All disposal values in the source ERMS will map to the appropriate historic fields in the XML file. Core metadata checksheets in the file: "Pre-Test Script Metadata Mapping Checksheets.xls" provide mapped metadata columns for the core metadata and exported XML field tags. These checksheets can be used to readily identify which active metadata fields in the source ERMS map to historic fields in the exported XML file.

While the date and time details may be presented differently within the source ERMS, this should not impact on the actual value format as expected by the XML Schema - i.e. YYYY-MM-DDThh:mm:ss.

Test Comments

Note: The core metadata set includes a number of disposal fields that provide the title, action, retention period and trigger for the disposal. However, as the definition of disposal schedules is not limited to a fixed number of fields, it is possible that additional information must also be carried over when migrating the data into another ERMS. This issue and more will be covered during Phase 3 of this project.

1.13.15. 15. Digital Signature

TBC

1.13.16. 16. Preservation

TBC

1.13.17. 17. Mandate

The Mandate values in the source ERMS will match the values in the XML file.

1.13.18. 18. Format

Format values in the source ERMS will match the values in the XML file.

1.13.19. 19. Function

The Function values in the source ERMS will match the values in the XML file.

1.13.20. 20. Coverage

The Coverage values in the source ERMS will match the values in the XML file.

Test Comments

1.14. Core metadata on bulk import

This section provides brief guidance and expectations for the bulk import of the core metadata elements.

At present, it is assumed that there will be no metadata field length issues - i.e. all exported metadata values from the source ERMS are expected to fit into corresponding fields in the destination ERMS.

It is strongly recommended that the bulk import functionality offered by an ERMS be capable of performing a "trial run" through of all the import data to check for / resolve conflicts BEFORE starting the actual import of any data. This approach will ensure that the import is either completed in full or abandoned altogether, thus avoiding potential complications should only some of the data be imported.

1.14.1. 1. Identifier

New System ID and Fileplan ID values will be applied automatically upon import of each fileplan object.

Historic metadata values will match the values in the XML file.

The Case ID value in the destination ERMS will match the value in the XML file.

Assumptions

Case ID values are specific to different pieces of casework, and it is essential that the Case ID value provide proper representation. Therefore, it is assumed that appropriate system checks will be performed automatically to establish whether or not the same Case ID value already exists in the destination ERMS in order to ensure duplication of Case ID values does not occur for different casework material.

If duplicate values were found, the bulk import process should prompt the user to either continue (informed decision) or stop the import process.

1.14.2. 2. Title

The Title value in the destination ERMS will match the value in the XML file.

Assumptions

It is assumed that appropriate system checks will be performed automatically to establish whether or not the same Title value already exists in the destination fileplan in order to identify potential hierarchical naming conflicts.

If a conflict is found, the bulk import process should be stopped.

Test Comments

1.14.3. 3. Subject

Subject metadata values are often selected from a controlled vocabulary list or thesaurus. Where this is the case, if the Subject value in the XML file is a member of the list of controlled terms, then the Subject value in the destination ERMS will match the value in the XML file. If the value being imported is not included in the list of controlled terms, an authorised user must decide whether or not to edit the list or abandon the import process.

If the Subject metadata does not use a list of controlled terms, or allows the additional use of free text entries, then the Subject value in the destination ERMS will match the XML file.

Assumptions

Manual and / or automated system checks will be performed to determine the management of the Subject metadata values.

1.14.4. 4. Description

The Description value in the destination ERMS will match the value in the XML file.

Assumptions

None.

1.14.5. 5. Creator

The Creator value in the destination ERMS will match the value in the XML file.

Assumptions

None.

1.14.6. 6. Date

The Date metadata values in the destination ERMS will match or reflect the values in the XML file.

While the date and time details may be presented differently within the destination ERMS, this should not impact on the actual values.

When records are exported, this may result in the creation of new files (copies of the original held in the source ERMS), with new Date.Created values for the new record content files.

This could lead to metadata conflicts and potential record integrity issues. Therefore testers should pay particular note of what happens with respect to record and component metadata and the record content during the bulk import process.

TNA will provide further guidance on the general expectations at a later stage.

Test Comments

Assumptions

If the date and time format in the destination ERMS is different to the XML file, the ERMS should be able to translate the value as appropriate. Ideally the original metadata value will not be affected.

1.14.7. 7. Correspondence Data

The email transmission metadata values in the destination ERMS will match or reflect the values in the XML file.

While the date and time details may be presented differently within the destination ERMS, this should not impact on the actual values.

Assumptions

If the date and time format in the destination ERMS is different to the XML file, the ERMS should be able to translate the value as appropriate. Ideally the original metadata value will not be affected.

It is also assumed that the mechanism used by the destination ERMS to view email records is capable of implementing the imported values in an appropriate manner. This may be of particular note where different email clients are used in the source and destination systems.

1.14.8. 8. Type

The class_type, folder_type, part_type and record_type values in the destination ERMS will match or reflect the values in the XML file.

Assumptions

Type values provide the name of a set of configuration rules that define the properties of a class, folder, part or record. The core metadata set only includes the name of the Type, not the full definition. Therefore, it is assumed that appropriate manual and system checks will be performed as necessary to establish whether or not the same type value already exists in the destination ERMS, or whether a new definition must be added.

Furthermore, if the same Type name exists in the destination ERMS but the definition is different, an authorised user must make a decision as to how best to proceed with the import. For example, it may be possible to automatically append the imported Type details with some distinguishing text such as "_import_2008-03-05". It is strongly recommended that any controlled changes be applied and audited by the ERMS rather than editing the XML file.

It is recommended that manual checks be performed BEFORE beginning the bulk import process wherever possible.

Test Comments

1.14.9. 9. Relation

When relational metadata is exported, many of the active values become historic. By default, the values in the destination ERMS will match the values in the XML file.

Assumptions

If the destination system offers a separate import area, the ERMS should automatically add the appropriate prefix to the existing child and parent object fileplan paths of the imported metadata values. The original XML data should not be edited.

While this Test Script does not include pointers, controlled copies, extracts, etc. management of relation metadata will require manual and systems checks to ensure the imported metadata is performed in the expected manner.

An additional consideration is that the destination ERMS may allow an authorised user to map specified relational metadata during the import process. This may be of use where it is known that related records are being imported together. For example, if a record and its related pointer or extract was being imported, the ability to map the exported historic metadata to active metadata in the destination ERMS would avoid the need to update metadata after import. However, strict caution must be used for such an approach because if any records are imported without their related pointers, extracts, etc. then this will result in broken links and could introduce conflicts and system errors.

1.14.10. 10. Aggregation

The Aggregation values in the destination ERMS will match or reflect the values in the XML file.

While the aggregation details may be presented differently within the destination ERMS, this should not impact on the actual values.

Assumptions

If the Aggregation name and / or format in the destination ERMS is different to the XML file, the ERMS should be able to translate the value as appropriate. Ideally the original metadata value will not be affected.

1.14.11. 11. Language

The Language value in the destination ERMS will match or reflect the value in the XML file.

While the Language details may be presented differently within the destination ERMS, this should not impact on the actual value.

Test Comments

Assumptions

If the Language value format in the destination ERMS is different to the XML file, the ERMS should be able to translate the value as appropriate. Ideally the original metadata value will not be affected.

1.14.12. 12. Location

The Location values in the destination ERMS will match the values in the XML file.

1.14.13. 13. Rights

The Rights values in the destination ERMS will match or reflect the values in the XML file. While some of the Rights details (e.g. Protective Marking, Protective Marking Change Date, FoI Disclosability Indicator, etc.) may be presented differently within the destination ERMS, this should not impact on the actual values.

Descriptor (Informative or Functional) metadata values may be selected from a controlled list. Where this is the case, if the Descriptor value in the XML file is a member of the list of controlled terms, then the value in the destination ERMS will match the value in the XML file. If the value being imported is not included in the list of controlled terms, an authorised user must decide whether or not to edit the list or abandon the import process.

If the Descriptor (Informative or Functional) metadata does not use a list of controlled terms, or allows the additional use of free text entries, then the value in the destination ERMS will match the XML file.

Assumptions

If any of the Rights value formats in the destination ERMS are different to the XML file, the ERMS should be able to translate each value as appropriate. Ideally the original metadata value will not be affected.

The core metadata set includes individual user and access control group names but not full user profile or group membership information. Therefore, it is assumed that appropriate manual and system checks will be performed as necessary to ensure the necessary access rights are established in the destination system.

Some systems support Functional Descriptors using the workaround of applying access control groups rather than a specific descriptor field. Where this is the case, it is assumed that data held in the "FunctionalDescriptor" field tag of the XML file will be automatically managed and mapped to achieve the workaround effect wherever possible as part of the bulk import process.

Test Comments

Manual and / or automated system checks will be performed to determine the management of the Descriptor (Informative or Functional) metadata values.

1.14.14. 14. Disposal

When disposal metadata is exported, all of the active values become historic. All disposal values in the destination ERMS will match the values in the XML file.

While the date and time details may be presented differently within the destination ERMS, this should not impact on the actual values.

Assumptions

If the date and time format in the destination ERMS is different to the XML file, the ERMS should be able to translate the value as appropriate. Ideally the original metadata value will not be affected.

The core metadata set includes a number of disposal fields that provide the title, action, retention period and trigger for the disposal. However, as the definition of disposal schedules is not limited to a fixed number of fields, it is possible that additional information must also be carried over into the destination ERMS when data is migrated.

Furthermore, even when all disposal metadata is present in the core metadata, appropriate system checks must be performed to establish whether or not the same Disposal Schedule Title value already exists in the destination ERMS in order to ensure duplication of Disposal Schedule Title values does not occur for different disposal schedules.

If duplicate values were found, the bulk import process should prompt the user to either continue (informed decision) or stop the import process.

1.14.15. 15. Digital Signature

TBC

Assumptions

None.

1.14.16. 16. Preservation

TBC

Assumptions

None.

Test Comments

1.14.17. 17. Mandate

The Mandate values in the destination ERMS will match the values in the XML file.

Assumptions

None.

1.14.18. 18. Format

In general, Format values in the destination ERMS will match the values in the XML file.

However, Application and / or Operating System metadata values may be managed using a controlled list. Where this is the case, a value in the XML file is a member of the list of controlled terms, then the value in the destination ERMS will match the value in the XML file. If the value being imported is not included in the list of controlled terms, an authorised user must decide whether or not to edit the list or abandon the import process.

If the Application and / or Operating System metadata does not use a list of controlled terms, or allows the additional use of free text entries, then the value in the destination ERMS will match the XML file.

Assumptions

Manual and / or automated system checks will be performed to determine the management of the Application and Operating System metadata values.

1.14.19. 19. Function

The Function values in the destination ERMS will match the values in the XML file.

Assumptions

None.

1.14.20. 20. Coverage

The Coverage values in the destination ERMS will match the values in the XML file.

Assumptions

None.

Test Comments

2. Test Script TS-XS03 Steps

2.1. Test execution formal details - Start

Start Date	
Start Time	
Tester(s)	
Pre-Test Note(s)	

Test Comments

2.2. Test steps

The following test steps must be performed in the specified order.

Step	Action	Expected Result	Actual Outcome	
			OK	Other
PREPARATION				
010	Login as SysAdmin .	User login details are recognised and authenticated.		
020	Confirm that the system date and time is: 2008-03-05T09:00:00 i.e. 9:00am on 5th March 2008	System datestamp details are set and verified.		
030	Confirm that Section 1: Test Script TS-X03 Preparation has been read.	Confirm that the Test Script preparation details have been read by the tester(s).		
040	Confirm that all of the specified test data files are available. See section 1.7. Test data .	Confirm that the Test Script preparation details have been applied.		
050	Confirm that all individual user profiles and access control groups have been created as specified. See sections 1.8. Individual user profiles and 1.9. Access control groups .	Otherwise, manually perform checks and definition tasks now.		
060	Confirm that all disposal schedules have been created as specified. See 1.10. Disposal schedules .			
070	*** CONDITIONAL STEP *** If any of the test data files, access and disposal settings have not been prepared, do so now.			

Test Comments

Step	Action	Expected Result	Actual Outcome	
			OK	Other
CLASS METADATA				
Note	<ul style="list-style-type: none"> This section of the Test Script covers metadata for the class: Class 1 \ Class 2 \ <u>Class 3</u> 			
080	<p>In the metadata sub-element: <u>Disposal.Disposal (Due / Effective) Date</u> for the class: Class 1 \ Class 2 \ <u>Class 3</u> manually check that the disposal schedule: Schedule E2 - Export folder 1 month after folder closed has been triggered, and that the due date value corresponds with the datestamp noted in Test Script TS-X02, step 1740. While the actual time may vary, the date will always be the same.</p>	<p>Confirm that when the folder was closed at the end of Test Script TS-X02, this triggered the retention period countdown for the disposal schedule. At this point, the full retention period has elapsed, and the export action is now due. <i>DISPOSAL SETTINGS</i> <i>Title: Schedule E2 - Export folder 1 month after folder closed</i> <i>Action: export</i> <i>External Event: N/A</i> <i>Internal Event: Folder closure.</i> <i>Retention Period: 1 month</i> <i>Due: 2008-03-03T00:00:01</i> The example value for the folder closure in Test Script TS-X02 was: 2008-02-03T23:20:00.</p>		
Note	<ul style="list-style-type: none"> Whenever the ERMS sets a datestamp value, the server clock (as opposed to the client pc clock) must be used. The client pc clock would normally be expected to mirror the server clock. This is of particular note with regards to disposal actions, as the current status is based solely on the server time - i.e. a disposal action cannot be "brought forward" by simply changing the local client time. The actual time of the due date may vary as some systems will apply a default daily time, while others may be quite specific and mark the time based on when the disposal schedule was triggered (e.g. when the folder was closed). 			

Test Comments

Step	Action	Expected Result	Actual Outcome	
			OK	Other
FOLDER METADATA				
Note	<ul style="list-style-type: none"> This section of the Test Script covers metadata for the folder: <u>Class 1 \ Class 2 \ Class 3 \ Folder 1</u> 			
090	<p>In the metadata sub-element: <u>Date.Closed</u> for the folder: <u>Class 1 \ Class 2 \ Class 3 \ Folder 1</u> manually check that the date and time details match the value noted in Test Script TS-X02, step 1740.</p>	<p>Date metadata is manually confirmed for the folder "Class 1 \ Class 2 \ Class 3 \ Folder 1". The example value for the folder closure in Test Script TS-X02 was: 2008-02-03T23:20:00.</p>		
100	<p>Manually check that the disposal metadata for folder: <u>Class 1 \ Class 2 \ Class 3 \ Folder 1</u> has been inherited from the parent class.</p>	<p>Disposal metadata is manually confirmed for the folder "Class 1 \ Class 2 \ Class 3 \ Folder 1". <i>DISPOSAL SETTINGS</i> <i>Title: Schedule E2 - Export folder 1 month after folder closed</i> <i>Action: export</i> <i>External Event: N/A</i> <i>Internal Event: Folder closure.</i> <i>Retention Period: 1 month</i> <i>Due: 2008-03-03T00:00:01</i> A brief summary of class metadata is also provided at step 080.</p>		

Test Comments

Step	Action	Expected Result	Actual Outcome	
			OK	Other
RECORD METADATA				
Note	<ul style="list-style-type: none"> This section of the Test Script covers metadata for the record: Class 1 \ Class 2 \ Class 3 \ Folder 1.Part 2 \ <u>Record 4</u> 			
110	<p>*** CONDITIONAL STEP ***</p> <p>If a non-default record_type was specified for: Class 1 \ Class 2 \ Class 3 \ Folder 1.Part 2 \ <u>Record 4</u> complete this test step. Otherwise, continue to the next step.</p> <p>Manually check that the disposal metadata for record: Class 1 \ Class 2 \ Class 3 \ Folder 1.Part 2 \ <u>Record 4</u> has been applied via the specific, non-default record_type definition.</p>	<p>Disposal metadata is manually confirmed for the record "Class 1 \ Class 2 \ Class 3 \ Folder 1.Part 2 \ Record 4".</p> <p><i>DISPOSAL SETTINGS</i></p> <p><i>Title: Schedule E4 - Export record 1 month after record declaration</i></p> <p><i>Action: export</i></p> <p><i>External Event: N/A</i></p> <p><i>Internal Event: Record declared</i></p> <p><i>Retention Period: 1 month</i></p> <p><i>Due: 2008-03-03T00:00:01</i></p>		
Note	<ul style="list-style-type: none"> The actual time of the due date may vary as some systems will apply a default daily time, while others may be quite specific and mark the time based on when the disposal schedule was triggered (e.g. when the record was declared). The date will always be the same: 2008-03-03. 			

Test Comments

Step	Action	Expected Result	Actual Outcome	
			OK	Other
EXPORT (via Disposal Schedule)				
120	<p>*** CONDITIONAL STEP ***</p> <p>This Test Script assumes that RecMgr has disposal management rights. If this is the case, complete this test step. Login as RecMgr.</p> <p>Otherwise, continue to the next test step and remain logged in as SysAdmin.</p>	User login details are recognised and authenticated.		
130	Initiate the disposal mechanism.	RecMgr is able to manage disposal actions via the user interface.		
Note	<ul style="list-style-type: none"> The way in which the disposal mechanism is initiated and governed may vary. However, each ERMS must allow authorised users to view and execute pending disposal actions as appropriate. 			
140	Confirm that the disposal event status is fully refreshed and up to date.	The status of all disposal events is manually confirmed as being fully up to date.		
Note	<ul style="list-style-type: none"> It is acknowledged that it may be inefficient to calculate and display disposal metadata in "real time". Where this is the case, the ERMS must provide a clear mechanism to update the data on request or as part of a periodic tracking / update function. 			

Test Comments

Step	Action	Expected Result	Actual Outcome	
			OK	Other
150	<p>Manually check that the disposal view specifically identifies the folder:</p> <p>Class 1 \ Class 2 \ Class 3 \ Folder 1</p> <p>as due for the disposal action:</p> <p>export</p>	<p>Disposal status is manually confirmed for the folder "Class 1 \ Class 2 \ Class 3 \ Folder 1".</p> <p>The disposal status is expected to also include all child objects except where other disposal schedules have been applied and may conflict.</p> <p>In this instance, the disposal of "Folder 1" will also include:</p> <p>"Part 1", "Record 1", "Part 2", "Record 2", and "Record 3".</p> <p>"Record 4" will also be included if it was declared using the default record_type.</p>		
Note	<ul style="list-style-type: none"> • <i>By default, the disposal mechanism should indicate that the export action for the folder applies to the folder object and all its contents (parts and records), with the exception of any records that have been allocated a separate and possibly conflicting disposal schedule via their specific (non-default) record_type definition.</i> • <i>As the folder was specified for export, it would not normally be possible to pick and choose which child objects to export. Other than the exception mentioned previously, the folder and its child objects would all be automatically exported as part of the same process.</i> 			

Test Comments

Step	Action	Expected Result	Actual Outcome	
			OK	Other
160	<p>*** CONDITIONAL STEP ***</p> <p>If a non-default record_type was specified for:</p> <p>Class 1 \ Class 2 \ Class 3 \ Folder 1.Part 2 \ <u>Record 4</u></p> <p>complete this test step. Otherwise, continue to the next step.</p> <p>Manually check that the disposal view specifically identifies the record:</p> <p>Class 1 \ Class 2 \ Class 3 \ Folder 1.Part 2 \ <u>Record 4</u></p> <p>as due for the disposal action:</p> <p>export</p>	<p>Disposal status is manually confirmed for the record "Class 1 \ Class 2 \ Class 3 \ Folder 1.Part 2 \ Record 4".</p>		
170	<p>Confirm that the ERMS can export metadata in accordance with the revised XML Schema.</p> <p>All core metadata fields used by the source ERMS should be mapped to corresponding XML field tags.</p> <p>Core and custom metadata check sheets in the file: "Pre-Test Script Metadata Mapping Checksheets.xls" provide mapped metadata columns for the ERMS metadata and exported XML field tags.</p>	<p>Support of the revised XML Schema model is confirmed.</p> <p>All core metadata elements and sub-elements have been mapped to corresponding XML field tags.</p> <p>Confirmation has been given that all metadata fields will be represented within exported XML files - whether populated with values or commented out if empty.</p>		

Test Comments

Step	Action	Expected Result	Actual Outcome	
			OK	Other
180	<p>*** CONDITIONAL STEP ***</p> <p>If there is any non-core (i.e. custom) metadata applied for the fileplan objects:</p> <p>Folder 1</p> <p> Part 1</p> <p> Record 1</p> <p> Part 2</p> <p> Record 2</p> <p> Record 3</p> <p> Record 4</p> <p>appropriate notes should be taken so that later checks (see step 420) can be made to ensure all metadata is managed as expected. Confirm that all custom metadata have been mapped to appropriate XML field tags.</p>	<p>Appropriate notes should be made to identify non-core (i.e. custom) metadata elements / sub-elements, and their associated values within the ERMS environment.</p> <p>All core metadata elements and sub-elements have been mapped to corresponding XML field tags.</p> <p>(While checksheets are available for core and custom metadata, reference worksheets are provided only for the core metadata data.)</p>		
Note	<ul style="list-style-type: none"> Where custom metadata is exported, these fields should be held within the "Dataset" XML block tag. Intuitive names should be given to each field, and the value specified as appropriate. Guidance on custom metadata will be further developed in Phase 2 of the XML Schema project. 			

Test Comments

Step	Action	Expected Result	Actual Outcome	
			OK	Other
190	<p>*** CONDITIONAL STEP A ***</p> <p>Confirm that the advanced model of the revised XML Schema is applied for exporting data in the current ERMS configuration.</p>	<p>The version of the revised XML Schema that is currently being applied for export actions is determined.</p> <p>The advanced model is preferred.</p>		
200	<p>*** CONDITIONAL STEP B ***</p> <p>Confirm that the basic model of the revised XML Schema is applied for exporting data in the current ERMS configuration.</p>	<p>Where only one ERMS is being tested, it is expected that this will match the bulk import setting (see steps 430 and 440).</p>		
210 export	<p>The export process is not complete until steps 210 - 260 have all been completed. Do not finalise the export action until these test steps have all been performed as appropriate.</p> <p>Initiate the function necessary to execute the export of the folder: Class 1 \ Class 2 \ Class 3 \ <u>Folder 1</u></p>	<p>The process of exporting the folder "Class 1 \ Class 2 \ Class 3 \ Folder 1" (and its child objects) using the disposal mechanism is entered into.</p>		

Test Comments

Step	Action	Expected Result	Actual Outcome	
			OK	Other
220 export	<p>*** CONDITIONAL STEP A ***</p> <p>If RecMgr is allowed to select an export directory, manually enter the destination: c:\TNA\TS-X03\export</p> <p>Manually create the directory on the local pc if this is not done automatically by the export function.</p>	<p>The location for the exported data is specified or determined.</p> <p>If user selection is allowed, the exported data will be held in the directory "c:\TNA\TS-X03\export" on the user's local pc.</p> <p>If a default system selection is applied, details will be noted accordingly.</p>		
230 export	<p>*** CONDITIONAL STEP B ***</p> <p>If a fixed location is used for exported data, take note of the destination details.</p>			
240 export	<p>*** CONDITIONAL STEP C ***</p> <p>If RecMgr is allowed to select a name for the exported XML file, manually enter the name: Export TS-X03-260.xml</p>	<p>The filename for the exported XML file is specified or determined.</p> <p>If user selection is allowed, the exported XML filename will be "Export TS-X03-260.xml".</p> <p>If a default or automated system filename is applied, details will be noted accordingly.</p>		
250 export	<p>*** CONDITIONAL STEP D ***</p> <p>If a fixed (or auto-generated) export filename is used for exported data, take note of the details.</p>			

Test Comments

Step	Action	Expected Result	Actual Outcome	
			OK	Other
260 export	<p>Manually enter the disposal comment: Folder 1 (and contents) exported at TS-X03, step 260. and complete the process of exporting the folder: Class 1 \ Class 2 \ Class 3 \ Folder 1</p> <p>The disposal action is authorised by: RecMgr.</p>	<p>The folder, and all the fileplan objects held within, is exported. A disposal comment was added and the disposal action was authorised by RecMgr. For example, the export directory "c:\TNA\TS-X03\export" holds:</p> <ul style="list-style-type: none"> • 1 XML file Export TS-X03-260.xml • 4 record content files: Record 1.doc Record 2.rtf Record 3.xls and an appropriate file format to represent the email "Record 4". * <p>* Where "Record 4" is declared using a non-default record_type, it may not be exported along with the parent folder. Instead a separate export will be performed (see step 320). The exported XML file contains complete metadata profiles for the exported folder, parts, records and components. Hierarchy details are also provided for the parent fileplan tree structure.</p>		
Note	<ul style="list-style-type: none"> • <i>Exporting record content may have an adverse effect on the Date.Created metadata. See section 1.13.6 6. Date for further details.</i> 			

Test Comments

Step	Action	Expected Result	Actual Outcome	
			OK	Other
270 export	<p>*** CONDITIONAL STEP ***</p> <p>If "Record 4" was created using a non-default record_type, which results in a disposal conflict where the record cannot be exported with the parent folder "Folder 1", complete this test step and the related conditional steps that follow.</p> <p>The export process is not complete until steps 270 - 320 have all been completed. Do not finalise the export action until these test steps have all been performed as appropriate.</p> <p>Initiate the function necessary to execute the export of the record: Class 1 \ Class 2 \ Class 3 \ Folder 1.Part 2 \ Record 4</p>	The process of exporting the record "Class 1 \ Class 2 \ Class 3 \ Folder 1.Part 2 \ Record 4" using the disposal mechanism is entered into.		
280 export	<p>*** CONDITIONAL STEP A ***</p> <p>If RecMgr is allowed to select an export directory, manually enter the destination: c:\TNA\TS-X03\export</p>	The location for the exported data is specified or determined. If user selection is allowed, the exported data will be held in the directory "c:\TNA\TS-X03\export" on the user's local pc.		
290 export	<p>*** CONDITIONAL STEP B ***</p> <p>If a fixed location is used for exported data, take note of the destination details.</p>	If a default system selection is applied, details will be noted accordingly.		

Test Comments

Step	Action	Expected Result	Actual Outcome	
			OK	Other
300 export	<p>*** CONDITIONAL STEP C ***</p> <p>If RecMgr is allowed to select a name for the exported XML file, manually enter the name: Export TS-X03-320.xml</p>	<p>The filename for the exported XML file is specified or determined.</p> <p>If user selection is allowed, the exported XML filename will be "Export TS-X03-320.xml".</p> <p>If a default or automated system filename is applied, details will be noted accordingly.</p>		
310 export	<p>*** CONDITIONAL STEP D ***</p> <p>If a fixed (or auto-generated) export filename is used for exported data, take note of the details.</p>			

Test Comments

Step	Action	Expected Result	Actual Outcome	
			OK	Other
320 export	<p>*** CONDITIONAL STEP ***</p> <p>Manually enter the disposal comment: Record 4 exported at TS-X03, step 320.</p> <p>and complete the process of exporting the record: Class 1 \ Class 2 \ Class 3 \ Folder 1.Part 2 \ <u>Record 4</u></p> <p>The disposal action is authorised by: RecMgr.</p>	<p>The record is exported.</p> <p>A disposal comment was added and the disposal action was authorised by RecMgr.</p> <p>For example, the export directory "c:\TNA\TS-X03\export" holds:</p> <ul style="list-style-type: none"> • 2 XML files Export TS-X03-260.xml * Export TS-X03-320.xml • 4 record content files: Record 1.doc * Record 2.rtf * Record 3.xls * <p>and an appropriate file format to represent the email "Record 4".</p> <p>* Entries marked with an asterisk denote the exported files associated with the export of "Folder 1" at step 260.</p> <p>The exported XML file contains complete metadata profiles for the exported record and component. Hierarchy details are also provided for the parent fileplan tree structure.</p>		
Note	<ul style="list-style-type: none"> • <i>Exporting record content may have an adverse effect on the Date.Created metadata. See section 1.13.6 6. Date for further details.</i> 			

Test Comments

Step	Action	Expected Result	Actual Outcome	
			OK	Other
330	<p>Confirm the existence of the exported XML file(s) and related record content files.</p> <p>Confirmation should be achieved by manual checks and also be reviewing the export log created by the ERMS.</p>	Confirmation is given to indicate that the export action is complete and that all of the files are in the expected location.		
Note	<ul style="list-style-type: none"> <i>The structural layout of exported data may vary between systems - e.g. some systems may hold all exported files in a single directory (e.g. the specified c:\TNA\TS-X03\export"), while others may separate record content files into one or more sub-directories, etc.</i> <p><i>Where the same source and destination ERMS is used, it is expected that the export and import functions will follow the same rules in terms of managing the output and input data respectively. However, if different source and destination systems follow different rules, the potential for additional manual or automated effort must be considered.</i></p> <p><i>As this Test Script only deals with a small amount of data, the simplest way to manage the exported data would be to hold all the fileplan object metadata and record content files within the same directory.</i></p>			
340	Confirm that ALL of the exported fileplan objects are still in the ERMS.	It is confirmed that the export action did not unexpectedly remove any of the fileplan objects from the ERMS.		
Note	<ul style="list-style-type: none"> <i>The way in which the ERMS updates disposal metadata after an export action has been performed may vary. For example, some systems may completely remove the export disposal settings, while others may leave them in place, ready to perform another export if required. Also, the export status text may differ.</i> 			

Test Comments

Step	Action	Expected Result	Actual Outcome	
			OK	Other
350	Validate the exported XML file(s).	The exported XML file(s) is validated against the schema rules.		
Note	<ul style="list-style-type: none"> • There are 15 xsd (XML Schema Definition) files that govern the validation of the exported XML files. These can be used with an XML validation application in order to establish whether or not there are any problems with the exported metadata. • The use of a separate Perl validation program will be investigated by TNA as part of further development work. 			
360	<p>*** CONDITIONAL STEP ***</p> <p>If there are any validation errors, complete steps 360 - 370.</p> <p>If any validation errors are due to user-error in following any test step from Test Script TS-X01, TS-X02 or TS-X03:</p> <ul style="list-style-type: none"> • apply the correction(s) via the ERMS user interface • remove the previously exported XML file(s) and related record content files (see steps 260 and 320) • perform the actions necessary to re-export the folder: Class 1 \ Class 2 \ Class 3 \ Folder 1 and its contents. <p>In addition, if necessary, also re-export the record:</p> <ul style="list-style-type: none"> • Class 1 \ Class 2 \ Class 3 \ Folder 1.Part 2 \ Record 4 	<p>If there are any validation errors in the exported XML file, first check for possible errors in performing previous test steps.</p> <p>If any errors are uncovered, the previous export data is discarded, and the complete export process is executed once more.</p>		

Test Comments

Step	Action	Expected Result	Actual Outcome	
			OK	Other
370	<p>*** CONDITIONAL STEP ***</p> <p>If any validation errors are due to unexpected metadata value issues, take note and investigate as required.</p> <p>Relevant feedback may also be sent to TNA: rmadvisory@nationalarchives.gov.uk</p>	<p>If there are any validation errors in the exported XML file, that do not correspond to user errors, or anomalies within the ERMS, appropriate notes should be taken and where possible, feedback given to TNA for follow-up.</p>		
380	<p>Compare the exported XML file with the appropriate example XML file (i.e. basic or advanced):</p> <p>Test Script TS-X03 Example XML File - Basic.xml</p> <p>Test Script TS-X03 Example XML File - Advanced.xml</p> <p>Take note and investigate any unexpected metadata value issues. Relevant feedback may also be sent to TNA: rmadvisory@nationalarchives.gov.uk</p>	<p>It is determined whether or not there are any metadata issues in the exported XML file.</p> <p>A comparison is made between the actual exported XML file and the corresponding example XML file provided in the test materials.</p> <p>The exported metadata should closely match the values already held in the checksheets and worksheets in the file "Test Script TS-X02 Checksheets and Worksheets.xls". However, it should be noted that the disposal metadata values may vary, depending on whether the ERMS updates the disposal status before or after the export is executed.</p> <p>Furthermore, checks on the live data held within the ERMS may provide an alternative means of checking any issues that may arise.</p>		

Test Comments

Step	Action	Expected Result	Actual Outcome	
			OK	Other
390	<p>Confirm that the exported record content files match those in the ERMS.</p> <p>In particular, check the metadata properties as specified in section 1.7. Test data.</p> <p>Also compare content by opening each file on the local pc and in the ERMS.</p>	<p>The exported record content files are manually checked.</p> <p>Record content should be exported in the appropriate source application format.</p>		
Note	<ul style="list-style-type: none"> Exporting record content may have an adverse effect on the Date.Created metadata. See section 1.13.6 6. Date for further details. 			

Test Comments

Step	Action	Expected Result	Actual Outcome	
			OK	Other
BULK IMPORT				
Note	<ul style="list-style-type: none"> <i>This section of the Test Script covers bulk import actions where data is imported using the original fileplan hierarchy "backbone" structure.</i> <i>The functionality to bulk import sections of isolated data at specified levels of the fileplan or the import of individual fileplan objects may be offered by the ERMS but they are not covered by this Test Script.</i> <i>If the source and destination systems support different fileplan hierarchy controls serious consideration must be given to the potential impact on the export and import of data. For example, it should be determined beforehand that both systems support the necessary number of class levels, etc. Automated conflict controls should ensure that most, if not all, hierarchy issues are addressed as part of the import process.</i> 			
400	<p>*** CONDITIONAL STEP ***</p> <p>Initiate the import mechanism.</p>	<p>RecMgr is able to manage bulk import actions via the user interface.</p> <p>If RecMgr does not have disposal management rights, SysAdmin will currently be logged on and will complete all of the bulk import steps (see step 120).</p>		
Note	<ul style="list-style-type: none"> <i>Some systems may use a specific area and mechanism to manage the imports, while others may simply activate functions.</i> 			

Test Comments

Step	Action	Expected Result	Actual Outcome	
			OK	Other
410	<p>Confirm that the ERMS can bulk import metadata in accordance with the revised XML Schema.</p> <p>All core XML fields should be mapped to corresponding core metadata fields within the destination ERMS.</p> <p>Core and custom metadata checksheets in the file: "Pre-Test Script Metadata Mapping Checksheets.xls" provide mapped metadata columns for the ERMS metadata and exported XML field tags.</p>	<p>Support of the revised XML Schema model is confirmed.</p> <p>All core metadata elements and sub-elements are supported in the destination ERMS and have been mapped to corresponding XML field tags.</p>		
420	<p>*** CONDITIONAL STEP ***</p> <p>If there is any non-core (i.e. custom) metadata applied for the exported fileplan objects:</p> <ul style="list-style-type: none"> Folder 1 <ul style="list-style-type: none"> Part 1 <ul style="list-style-type: none"> Record 1 Part 2 <ul style="list-style-type: none"> Record 2 Record 3 Record 4 <p>refer to notes taken at step 180.</p> <p>Confirm that all custom XML fields have been mapped to corresponding custom metadata fields within the destination ERMS.</p>	<p>Where the same source and destination ERMS is used, there is an assumption that the same custom fields will be supported and as long as the metadata fields are mapped in the correct way, the import action will process all the metadata in the expected manner.</p> <p>However, if different systems are used, the tester must ensure that the destination ERMS supports all of the exported custom metadata fields, and produce a full metadata mapping before import.</p>		

Test Comments

Step	Action	Expected Result	Actual Outcome	
			OK	Other
430	<p>*** CONDITIONAL STEP A ***</p> <p>Confirm that the advanced model of the revised XML Schema is applied for importing data in the ERMS current configuration.</p>	<p>The version of the revised XML Schema that is currently being applied for import actions is determined.</p> <p>The advanced model is preferred.</p> <p>Where only one ERMS is being tested, it is expected that this will match the export setting (see steps 190 and 200).</p>		
440	<p>*** CONDITIONAL STEP B ***</p> <p>Confirm that the basic model of the revised XML Schema is applied for importing data in the ERMS current configuration.</p>			
Note	<ul style="list-style-type: none"> <i>If the source and destination systems support different versions of the XML schema, serious consideration must be given to the potential impact on the export and import of data.</i> 			
450	<p>Initiate the function necessary to perform the bulk import of the folder:</p> <p><u>Class 1 \ Class 2 \ Class 3 \ Folder 1</u></p> <p>and all its child objects.</p>	<p>The process of bulk importing the folder "Class 1 \ Class 2 \ Class 3 \ Folder 1" (and all its child objects) is entered into.</p>		

Test Comments

Step	Action	Expected Result	Actual Outcome	
			OK	Other
460	<p>By default, the imported data should follow the original fileplan hierarchy structure.</p> <p>Where the source and destination ERMS are the same, an attempt to import complete fileplan objects with the same Titles in the same locations should return a conflict message.</p> <p>Conflicts are not expected where only minimum metadata is present to form the "backbone" of the original structure.</p> <p>When such a conflict message is presented, the appropriate action should be taken so that no import is performed.</p>	<p>Automated conflict controls for the bulk import process are invoked.</p> <p>The presence of minimum metadata for "Class 1", "Class 2" and "Class 3" does not result in a conflict.</p> <p>The presence of full metadata for "Folder 1" (and its child objects) does result in conflicts.</p> <p>No import is performed.</p>		
Note	<ul style="list-style-type: none"> <i>In theory, there may be instances where system administrators wish to import data that will overwrite existing fileplan data. Therefore, some systems may present suitably authorised users with the option to continue once the conflict message has been displayed. Other restrictions may apply (e.g. they may only be able to import a single object at a time, with individual confirmations) but such actions must always be properly governed and automatically audited.</i> 			

Test Comments

Step	Action	Expected Result	Actual Outcome	
			OK	Other
470	<p>*** CONDITIONAL STEP ***</p> <p>While it is strongly recommended that automated conflict controls be in place, some systems may not apply such rules.</p> <p>Therefore, if no conflict message is presented, and there is no option to prevent the import action, examine the affected area of the fileplan and determine the overall impact of the action.</p>	<p>If automated conflict controls for the bulk import process are not implemented, and the import action cannot be stopped once initiated, the import will be performed.</p> <p>This is not a desirable outcome.</p> <p>Where the import does take place, the tester is responsible for checking the fileplan and determining the overall impact of the import action.</p>		

Test Comments

Step	Action	Expected Result	Actual Outcome	
			OK	Other
480	<p>*** CONDITIONAL STEP A ***</p> <p>The ERMS may provide a bulk import option for the authorised user to specify a new top level class to hold the imported fileplan tree.</p> <p>This top level class would be expected directly under the main root class, and would effectively provide a separate classification scheme in addition to the classification scheme already in existence within the destination ERMS.</p> <p>If RecMgr is allowed to select an import class name, manually specify the new class "Import Hierarchy".</p> <p>Continue to step 500.</p>	<p>The import parent class for the imported data is specified or determined.</p> <p>If user selection is allowed, the imported data will be held under the top level class "Import Hierarchy".</p> <p>If a default class is applied, details will be noted accordingly.</p>		
490	<p>*** CONDITIONAL STEP B ***</p> <p>If a fixed class is used for bulk import data, take note of the class details.</p> <p>Continue to step 500.</p>			

Test Comments

Step	Action	Expected Result	Actual Outcome	
			OK	Other
500	<p>*** CONDITIONAL STEP A/B ***</p> <p>Import the previously exported metadata and record content under the "Import Hierarchy" class.</p>	<p>The previously exported data is imported en masse under the class "Import Hierarchy".</p> <p>Fileplan object metadata and record content is imported.</p> <p>The resulting fileplan branch for the "Import Hierarchy" class can be seen in section 2.4: "Hierarchy model - End.</p>		
Note	<ul style="list-style-type: none"> Exporting record content may have an adverse effect on the Date.Created metadata. See section 1.14.6 6. Date for further details. 			
510	<p>*** CONDITIONAL STEP A/B ***</p> <p>Confirm that ALL of the imported fileplan objects are now in the ERMS.</p> <p>Confirmation should be achieved by manual checks and also be reviewing the import log created by the ERMS.</p>	<p>Confirmation is given to indicate that the bulk import action is complete and that all of the files are in the expected location.</p> <p>The bulk import must not have affected the integrity of the data held within the fileplan hierarchy.</p>		
520	<p>*** CONDITIONAL STEP A/B ***</p> <p>Confirm that the imported record content files correspond to those on the local pc.</p> <p>In particular, check the metadata properties as specified in section 1.7. Test data.</p> <p>Also compare content by opening each file on the local pc and in the ERMS.</p>	<p>The imported record content files are manually checked.</p> <p>Record content should be imported in the appropriate source application format.</p>		

Test Comments

Step	Action	Expected Result	Actual Outcome	
			OK	Other
530	In order to provide a minimum, common proof of bulk import capability, make a copy of the exported XML file (generated at step 260): c:\TNA\TS-X03\export\ Exported File TS-X03-260.xml to: c:\TNA\TS-X03\export\ Copy Export TS-X03-260.xml	A copy of the previously exported XML file is made. Additional copies of the record content are not needed; the originals can be used.		
Note	<ul style="list-style-type: none"> The content of an exported XML file would not normally be manually edited. A controlled edit is performed here for test purposes only. Any change to the exported XML file may compromise the integrity of the data and it is strongly recommended that file edits be avoided. 			
540	Edit the copied XML file: c:\TNA\TS-X03\export\ Copy Export TS-X03-260.xml so that the folder title text: "Folder 1" is replaced by "Folder 1A" throughout.	The copied XML file is edited so that there will no longer be a conflict in the fileplan with regards to the folder that will be imported.		

Test Comments

Step	Action	Expected Result	Actual Outcome	
			OK	Other
550	Perform the actions necessary to execute the bulk import of the "Folder 1A" data.	The previously exported data is imported en masse under the original fileplan hierarchy. "Folder 1" and "Folder 1A" are sibling folders within the class "Class 1 \ Class 2 \ Class 3". Fileplan object metadata and record content is imported. The resulting fileplan branch can be seen in section 2.4: "Hierarchy model - End.		
Note	<ul style="list-style-type: none"> Exporting record content may have an adverse effect on the Date.Created metadata. See section 1.14.6 6. Date for further details. 			
560	Confirm that ALL of the imported fileplan objects are now in the ERMS. Confirmation should be achieved by manual checks and also be reviewing the import log created by the ERMS.	Confirmation is given to indicate that the bulk import action is complete and that all of the files are in the expected location. The bulk import must not have affected the integrity of the data held within the fileplan hierarchy.		
570	Confirm that the imported record content files correspond to those on the local pc. In particular, check the metadata properties as specified in section 1.7. Test data. Also compare content by opening each file on the local pc and in the ERMS.	The imported record content files are manually checked. Record content should be imported in the appropriate source application format.		
Note	<ul style="list-style-type: none"> "Record 1", "Record 2" and "Record 3" will always be included in this bulk import of previously exported data. However, depending on how it was managed, "Record 4" may not be included. 			

Test Comments

Step	Action	Expected Result	Actual Outcome	
			OK	Other
FINAL METADATA CHECKS				
580	<p>In order to confirm that the metadata settings have been applied as expected throughout, and also confirm any known issues, checksheets and example worksheets in the file:</p> <p style="text-align: center;">Test Script TS-X03 Checksheets and Worksheets.xls</p> <p>should be compared.</p> <p>The actual metadata values entered by the tester(s) in either:</p> <p style="text-align: center;">TS-X03 Checksheet 3A</p> <p>or</p> <p style="text-align: center;">TS-X03 Checksheet 3B</p> <p>should be compared against the values in the corresponding example worksheet:</p> <p style="text-align: center;">TS-X02 Worksheet 3A</p> <p>or</p> <p style="text-align: center;">TS-X02 Worksheet 3B</p> <p>The metadata values should generally match or demonstrate an acceptable similarity. If there are distinct differences, these should correspond to Test Comments made during the Test Script.</p> <p>The tester(s) may also compare actual values via the ERMS interface.</p>	<p>All metadata values are manually confirmed for all of the imported fileplan objects (class, folder, part, record and component) in this Test Script.</p> <p>In terms of the original fileplan branch, only the disposal metadata is likely to have changed during this Test Script.</p> <p>Any imported metadata should closely match the metadata values for the original branch. Depending on the way in which the ERMS manages the disposal metadata, the disposal details will either reflect the values before the export was executed (see checksheets and worksheets for TS-X02) or after the export was executed (see checksheets and worksheets for TS-X03).</p> <p>The checksheets and worksheets for TS_X03 only reflect the metadata status for the original fileplan branch.</p>		
END OF TEST SCRIPT TS-X03				

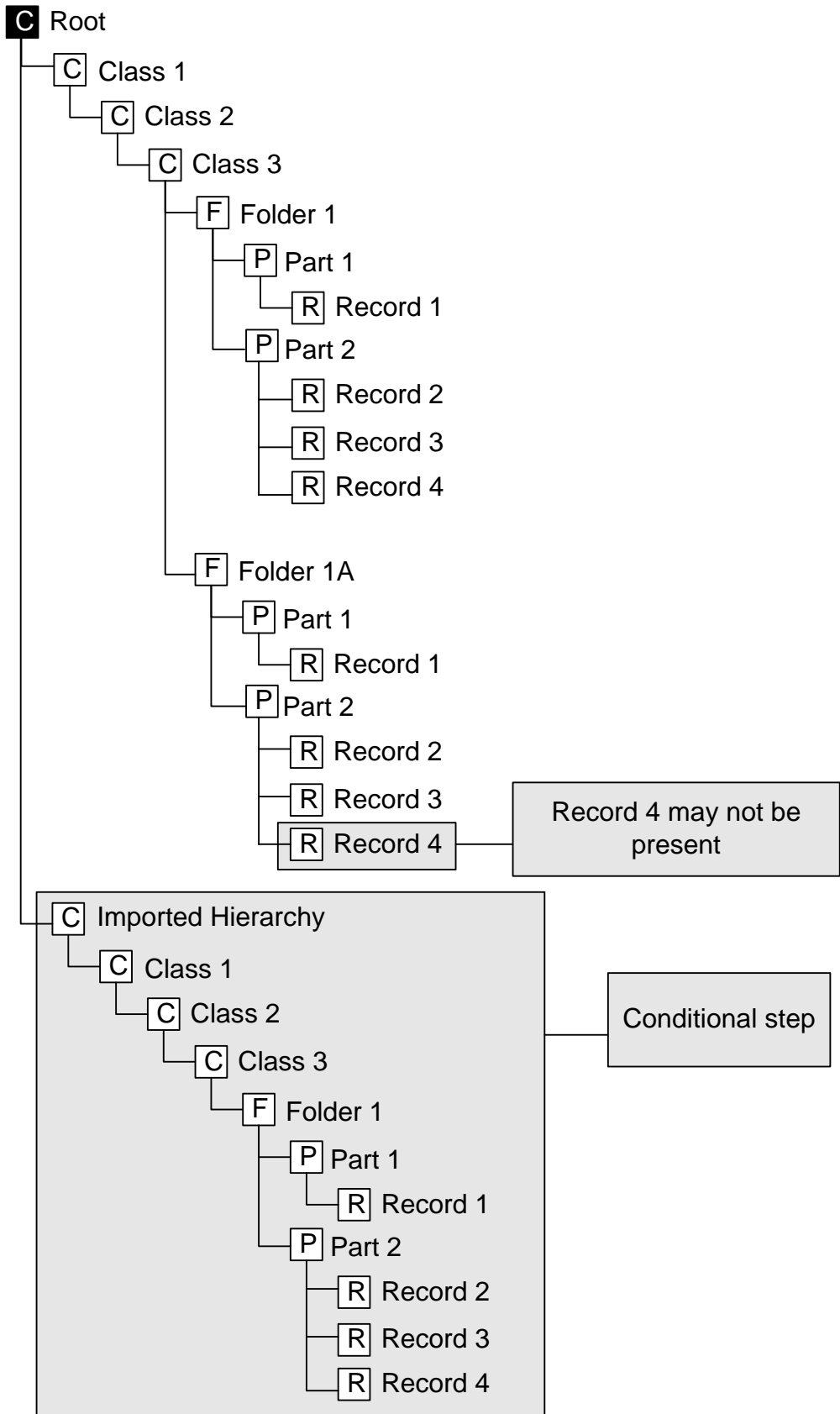
Test Comments

2.3. Test execution formal details - End

End Date	
End Time	
Post-Test Note(s)	

Test Comments

2.4. Hierarchy model - End



Note: The "root" is not normally included as part of the fileplan path details.

2.5. Metadata values - End

Example metadata values are provided in the worksheets in the file: Test Script TS-X03 Checksheets and Worksheets.xls". This file also contains template checksheets, which the tester(s) should have completed with actual metadata values throughout this Test Script.

END OF TEST SCRIPT TS-X03

Test Comments