Cairo use cases

a survey of user scenarios applicable to the Cairo ingest tool

21 May 2007

Version 1.0

WELLCOME LIBRARY

Cairo Project team

Susan Thomas,
Project Manager/Digital Archivist,
Oxford University Library Services

Fran Baker,
Digital Archivist, John Rylands University Library

Renhart Gittens,
Software Engineer,
Oxford University Library Services

Dave Thompson,
Digital Curator, Wellcome Library

Cairo is funded under the 'Tools and innovations' strand of JISC's Repositories and Preservation programme.

JISC

Table of Contents

Introduction to the Cairo tool

The aim of the Cairo project is to create a tool which provides an interface for the ingest workflow, which brings together ingest tools, especially metadata creation tools, into a single coherent, usable and documented tool, which is suitable for use by professional archivists with limited technical competencies. The tool should be capable of processing formats commonly found in personal digital archives and be extensible, so that support for other formats and the metadata they need can be added as necessary. The tool's output should be digital archives that have been subject to ingest processes, together with repository-independent metadata packages in the form of METS files, which document that workflow and record metadata that will provide the basis for long term lifecycle management.

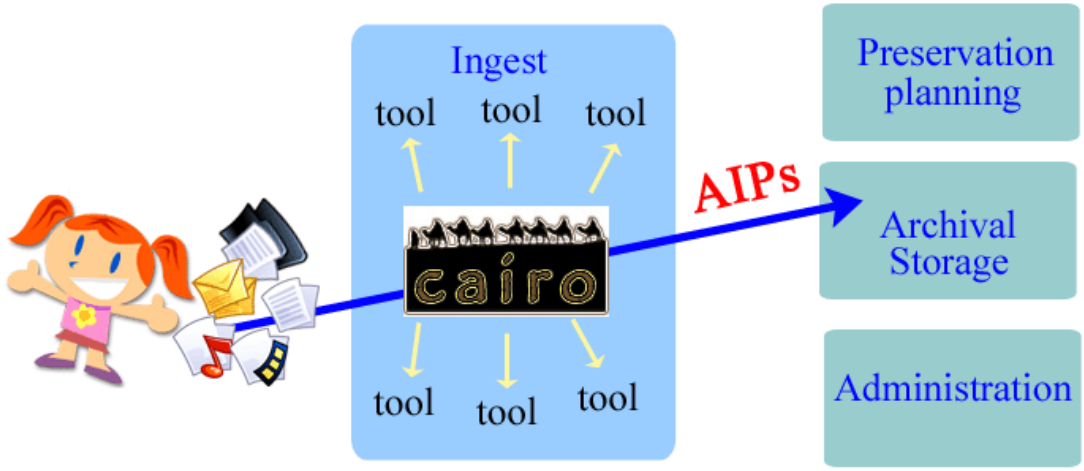

Illustration 1: Archivist presents personal digital archive to Cario, which creates Archival Information Packages (AIPs) for the lifecycle management of the archive and adds them to archival storage.

The principal user of the Cairo tool will be an archivist performing the everyday tasks of receiving archival material, preparing it for placement in long-term storage, and running queries or generating reports on work processed by the Cairo tool. The archivist will present an arrangement of digital archives to Cairo whereupon the tool will coordinate an ingest workflow, with a minimum of input from the archivist, resulting in the metadata packages needed for the lifecycle management of the digital archives.

The digital archives and their metadata may then be presented to some kind of archival storage, perhaps a digital repository system, as Archival Information Packages (AIPs).

Currently, the process of preparing digital archives for ingest requires knowledge of many file formats, ingest-related tools and metadata standards:

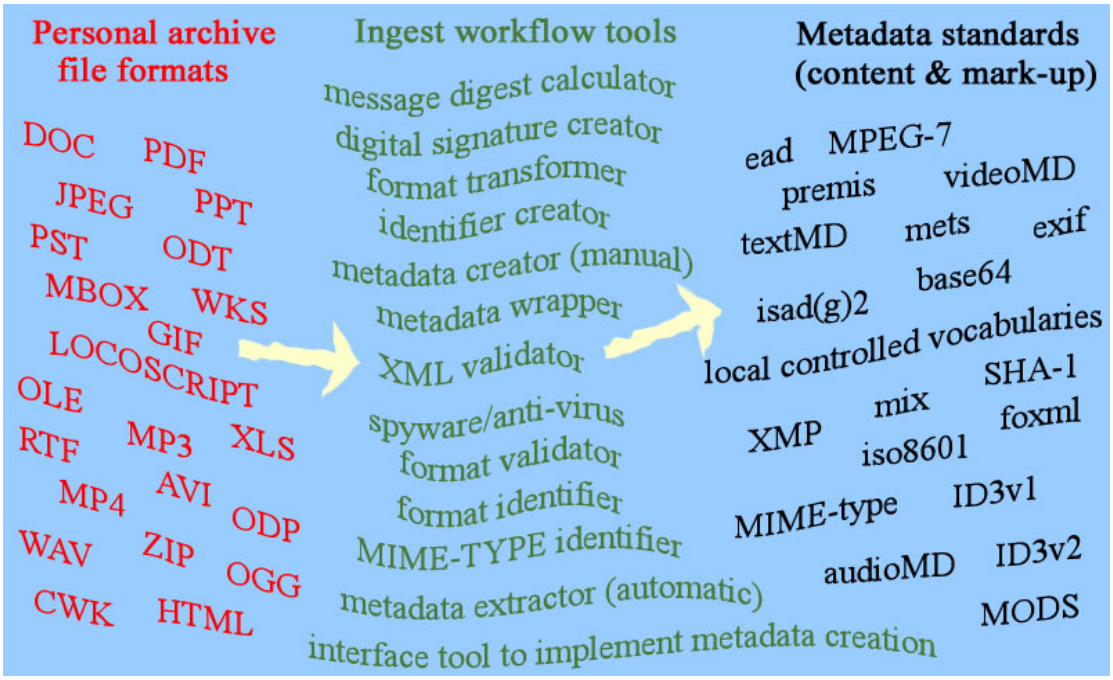

Illustration 2: The range of file formats, ingest tools and metadata standards applicable to the preparation of personal digital archives for preservation.

The quantity and complexity of new knowledge required presents a significant barrier to engaging professional archivists in digital preservation.

The Cairo tool will alleviate this complexity by orchestrating an ingest workflow, composed of several discrete components and standards, from a single user interface. The archivist is therefore not expected to be an expert user of XML, command-line metadata utilities or the means by which the output of individual tool components is aggregated into a standards-based metadata package. Nonetheless, it is advisable for the archivist to learn something about the nature of the preservation metadata that the Cairo tool is creating. It is hoped that Cairo can help in this by providing a gentler introduction to the creation of such metadata than is presently available: users of the Cairo tool will interact with the ingest workflow through selecting options on 'radio buttons', drop-down menus or from pre-determined options taking effect as a consequence of user selections.

The archivist is expected to understand the basic principles of archival practice and especially those principles that relate to authenticity, context and provenance, such as the creation of meaningful hierarchical arrangements of material. These principles are derived from the practice of working with physical materials and must also be applied to their digital equivalents. While the Cairo tool will offer an easy-to-use interface it will not do away with the need for professional archivists to apply specialist skills and understandings.

Further information about the roles of Cairo users is available in the Cairo Tool Review document.

Introduction to the Cairo use cases

This document outlines a set of use cases describing the different interactions users of the Cairo tool have with that tool. The use cases also describe the behaviour of the tool in response to those user interactions.

These use cases model both system and user behaviour in a range of scenarios. Use cases allow behaviour to be proposed, examined by key stakeholders and modified as a result of their feedback. Use cases are also used to examine user expectations of the system in terms of its behaviour, interfaces and feedback. Management of expectation and the design of uses/system interaction at an early stage of the Cairo tool's development allows the project team the opportunity to easily modify the tool's basic design.

These use cases are not designed to be an exhaustive road map of all potential interactions at a micro level. There will still be a need to create a clear and specific set of Cairo tool requirements prior to commencing tool construction and, potentially, to make alterations as the development of the tool progresses.

It is intended that these high level use cases inform Cairo tool design, but do not themselves strictly define how the tool should look and feel or provide the functionality discussed below.

Associated documents

This Tools Survey document should be read in conjunction with other documents designed to inform the development of the Cairo tool:

- Cairo Content Typology Model
- Cairo Use Cases.

Roles associated with the Cairo tool

Several roles will be involved with the Cairo tool from developing to deploying using, administering and enhancing the tool. The following use cases describe scenarios which might be applicable to five types of role:

1. **Administrator** – the User responsible for day-to-day system administration, typically an archivist
2. **Processor** – a routine User of the system, typically an archivist
3. **Sysadmin** – technician responsible for infrastructure administration
4. **Developer** – responsible for creating new or additional system functionality, may be internal or external
5. **The system itself** – described in the use cases through its feedback role in monitoring activity, reporting activity and performing automated functions, etc.

There are only two kinds of User that will interact with the system regularly. The most common User will be the archivists responsible for ingesting digital materials into repositories (described below as 'Processor') followed by the Administrator. The Administrator will be drawn from the archivist group and will probably also be a Processor; this User will therefore have two roles - as a Processor of digital archives and as an Administrator with system authority to

effect certain changes to the Cairo tool and to manage the Cairo tool user base. It is likely that the Administrator will have greater expertise in digital curation and preservation in addition to archival training.

The role of the **User Administrator** is to:

- set general system operating parameters.
- configure tool settings at a high level.
- maintain user accounts.
- act as first level support in the event of problems.
- report problems to the sysadmin, especially network access problems.
- ensure that the activities of the sysadmin and developer are not in conflict with the basic archival principles.
- identify the need for new system functionality and develop related functional requirements.
- where necessary, run queries or generate reports on work processed by the Cairo tool.

The role of the **User Processor** is to:

- use the system to create Archival Information Packages (AIP) from complex objects.
- where necessary, run queries or generate reports on work processed by the Cairo tool.

The role of the **Sysadmin** is to:

- install the tool
- configure access to basic network services.
- maintain the infrastructure on which the tool resides and in which it operates. The sysadmin will be responsible for maintaining hardware, software and access to all necessary networked resources.
- provide technical input into the development of new or additional system functionality in support of the administrator.

The role of the **Developer** is to:

- create new or additional system functionality.
- further develop the Cairo tool interface.
- design user interface.

It is not the role of the Processors to maintain the system, or to modify operating parameters, etc. In their routine work the Processor will select from options presented by the system. Ideally, the need to select from these options will be minimal with the ‘default’ options sufficing for most material, in most cases, most of the time. The Processor should not be permitted any interaction with the system that permanently modifies default operating parameters. There should be no ‘level’ of User without archival training.

The system will have some automated functionality by which it passes relevant information back to the different Processors and Administrators. This

includes such things as notification of error conditions, completion of requested tasks, etc.

Whilst the Processor will have minimal need to change default system parameters, the system will not be allowed to act wholly autonomously or independently of Processor instruction. This may change in the future as confidence with the system increases over time and system functionality and reliability are improved.

Use cases

Use cases have been tabulated to the following format,

Use case number	The number of the use case
Use case class	The class of the use case, i.e. Administrator Processor Sysadmin Developer System
Use case description	Free text description of the use case scenario
Priority	Terminology adopted from RFC 2119: 1. MUST - This word, or the terms "REQUIRED" or "SHALL", mean that the definition is an absolute requirement of the specification. 2. MUST NOT - This phrase, or the phrase "SHALL NOT", mean that the definition is an absolute prohibition of the specification. 3. SHOULD This word, or the adjective "RECOMMENDED", mean that there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course. 4. SHOULD NOT - This phrase, or the phrase "NOT RECOMMENDED" mean that there may exist valid reasons in particular circumstances when the particular behaviour is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behaviour described with this label.
Primary actor	The role most responsible for the actions described in the use case
Secondary actor	Any other role associated or involved in the use case
Steps	The individual actions of the use case (Steps enclosed in parentheses may be conditional on some other action, intervention or condition)
Notes	Anything not covered above but adding further information

Usecases - Processor

The Processor will normally be an archivist who has been assigned responsibility for ingesting a series of folders and files into the digital repository; an Administrator might also act as a Processor.

Summary

Use Case	Processor Use case description.
P1	Processor wants to process the first accession of a given archive
P2	Processor wants to manually enter details of new collection
P3	Processor wants to add a new accession to an existing archive
P4	Processor wishes to cancel current process
P5	Processor wishes to continue working with a saved process
P6	Processor wants to suspend work on processing an accession and continue working on an accession started earlier
P7	Processor wants to suspend work on processing an accession and log out
P8	Processor wishes to the change their level of system permission
P9	Processor wishes to generate report of their work processed in a given time period
P10	Processor wishes to change their password
P11	User wishes to ‘suspend’ their login whilst Cairo tool performs a long task
P12	Processor wishes to view a list of their work saved by the Cairo tool
P13	Processor wishes to view a list of all work saved by the Cairo tool
P14	Processor wishes to view a list of all the accessions to a specified archive or collection
P15	Processor wishes to view a list of all the accessions of material which have come from a specified donor or depositor
P16	Processor wishes to view a list of work processed in a given time period

Detail

Use case number	P1
Use case class	Processor
Use case description	Processor wants to process the first accession of a given archive
Priority	Must
Primary actor	Processor
Secondary actor	N/A
Steps	1. User logs in to Cairo as a Processor and is authenticated 2. Cairo asks User if they want to a. Process a new collection b. Add a new accession to an existing collection c. Continue working on a saved process d. Save work at current point e. View 'personal' workspace f. Generate reports g. Lock session/log out 3. User selects 'Process a new collection' and fills in a few high-level details (see usecase P2 below) 4. System verifies location of network services and confirms that media/files are available at that location 5. Cairo processes work, calling upon the various tools to process the archives and to generate the metadata specified by the tool's configuration settings. The result is METS files for the collection object; a child accession object; for folder and file objects; event objects; agent objects; and rights objects. 6. System reports back to User when process successfully completed and offers the User an opportunity to view a log of the process. 7. The METS files and associated digital objects (where relevant) can now be ingested into a digital repository, or simply stored in a file system.
Notes	

Use case number	P2
Use case class	Processor
Use case description	Processor wants to manually enter details of new collection
Priority	Must
Primary actor	Processor
Secondary actor	N/A
Steps	1. User logs in to Cairo as a Processor and is authenticated 2. User has already selected 'Process a new collection' 3. User is presented with on screen form 4. User enters details of the new collection such as, a. Name of collection b. Whether collection has analogue component c. Local collection identifier/link to local descriptive system record d. Date of accession e. Location of accession on disk f. Own name 5. Cairo checks that mandatory fields are completed 6. Cairo forces completion of mandatory fields 7. Cairo checks validity of field data, e.g. date field must be numeric and in ISO format 8. Cairo forces validity of field data 9. Cairo accepts User input 10. Cairo allows User to progress to next stage (usecase P1)
Notes	Notes: Cairo should not halt if a Processor fails to input data in the correct format. The system should continue to seek input in the appropriate form from the archivist and offer them an option to 'Cancel' if they cannot supply the correct input. Proper and full documentation and/or contextual help will assist here

Use case number	P3
Use case class	Processor
Use case description	Processor wants to add a new accession to an existing archive
Priority	Must
Primary actor	Processor
Secondary actor	N/A
Steps	1. User logs in to Cairo as a Processor and is authenticated 2. Cairo asks User if they want to a. Process a new collection b. Add a new accession to an existing collection c. Continue working on a saved process d. Save work at current point e. View 'personal' workspace f. Generate reports g. lock session/log out 3. User selects 'Add a new accession to an existing collection' 4. Cairo asks User to identify existing archive 5. User supplies accession specific metadata such as date of accession and location of accession on the disk and submits 6. Cairo creates METS files for the new accession consisting of an accession object which will be the child of an existing collection object and which records the structure of the accession as represented on the disk using METS structmap; folder and file objects; event objects; agent objects; and rights objects.
Notes	

Use case number	P4
Use case class	Processor
Use case description	Processor wishes to cancel current process
Priority	Must
Primary actor	Processor Administrator
Secondary actor	N/A
Steps	1. User logs in to Cairo as a Processor and is authenticated 2. Cairo asks User if they want to a. Process a new collection b. Add a new accession to an existing collection c. Continue working on a saved process d. Save work at current point e. View 'personal' workspace f. Generate reports g. Lock session/log out 3. User selects 'Add a new accession to an existing collection' 4. User selects option and begins process 5. User wishes to cancel current process 6. User selects one of two options a. Cancel current stage of process b. Cancel entire process for this object 7. User selects preferred option 8. Cairo presents User with either start screen for current process or options from step 2
Notes	Depending upon which degree of 'cancel' is selected some or all work on the object should be completely deleted. There should be no widows or orphans existing in any system. However, the original files must not be deleted.

Use case number	P5
Use case class	Processor
Use case description	Processor wishes to continue working with a saved process
Priority	Must
Primary actor	Processor
Secondary actor	N/A
Steps	1. User logs in to Cairo as a Processor and is authenticated 2. Cairo asks User if they want to a. Process a new collection b. Add a new accession to an existing collection c. Continue working on a saved process d. Save work at current point e. View ‘personal’ workspace f. Generate reports g. Lock session/log out 3. User selects ‘Continue working on a saved process’ option, is prompted for details of the relevant saved process, and continues process
Notes	

Use case number	P6
Use case class	Processor
Use case description	Processor wants to suspend work on processing an accession and continue working on an accession started earlier
Priority	Must
Primary actor	Processor Administrator
Secondary actor	N/A
Steps	1. User is logged into Cairo as a Processor and is authenticated 2. User selects ‘Save work at current point’ 3. Cairo forces User to complete current step 4. Cairo asks for conformation 5. Cairo saves work completed up to this point 6. Cairo logs date and time of save action 7. Cairo asks if User wishes to a. Process a new collection b. Add a new accession to an existing collection c. Continue working on a saved process d. View ‘personal’ workspace e. Generate reports f. Lock session/log out 8. User selects relevant option. 9. Cairo adds events to instance process log file
Notes	This scenario might arise when the archivist must work on ingesting another collection or accession, etc. Notes: The Administrator can configure the options offered to a Processor when selecting where to store saved work; it should also offer a unique ‘logical’ file name for the saved work. The system should store internal information about saved work so that it can report on saved work that has not been actioned after *n* period of time.

Use case number	P7
Use case class	Processor
Use case description	Processor wants to suspend work on processing an accession and log out
Priority	Must
Primary actor	Processor
Secondary actor	N/A
Steps	1. User is logged into Cairo as a Processor and is authenticated 2. User selects ‘Save work at current point’ 3. Cairo forces User to complete current step 4. Cairo asks for conformation 5. Cairo saves work completed up to this point 6. Cairo logs date and time of save action 7. Cairo asks if User wishes to a. Process a new collection b. Add a new accession to an existing collection c. Continue working on a saved process d. View ‘personal’ workspace e. Generate reports f. Lock session/log out 10. User selects Log out
Notes	Notes This scenario might arise when the archivist must finish work at the end of the day, or must go to a meeting. The Administrator can configure the options offered to a Processor when selecting where to store saved work; it should also offer a unique ‘logical’ file name for the saved work. The system should store internal information about saved work so that it can report on saved work that has not been actioned after *n* period of time.

Use case number	P8
Use case class	Processor
Use case description	Processor wishes to the change their level of system permission
Priority	Must
Primary actor	Processor Administrator
Secondary actor	N/A
Steps	1. User logs in to Cairo as a Processor and is authenticated 2. Cairo asks User if they want to a. Process a new collection b. Add a new accession to an existing collection c. Continue working on an saved process d. Save work at current point e. View ‘personal’ workspace f. Generate reports g. Lock session/log out 3. User selects view personal workspace 4. User is presented with their ‘personal’ workspace page and options to: a. Change role b. Change password c. Generate reports d. View saved work e. Specify location of preferred XML editor f. Autosave options g. Others? 5. User selects ‘Change role’ option 6. User is presented with list of different roles, e.g. Administrator or Processor 7. User selects new role 8. Cairo asks for User password 9. If entitled and if password is correct User is allowed to change role 10. If not entitled User is informed so by system 11. If password incorrect User is invited to re-submit password
Notes	A User should not have to log out to switch between Processor and Administrator roles.

Use case number	P9
Use case class	Processor
Use case description	Processor wishes to generate report of their work processed in a given time period
Priority	Should
Primary actor	Processor
Secondary actor	N/A
Steps	1. User logs in to Cairo as a Processor and is authenticated 2. Cairo asks User if they want to a. Process a new collection b. Add a new accession to an existing collection c. Continue working on a saved process d. Save work at current point e. View ‘personal’ workspace f. Generate reports g. Lock session/log out 3. User selects view personal workspace 4. User is presented with their ‘personal’ workspace page and options to: a. Change role b. Change password c. Generate reports d. View saved work e. Others? 5. User selects ‘generate reports’ option 6. User supplies report parameters 7. Cairo generates report
Notes	This scenario may occur when a Processor is required to update a supervisor on the progress of processing a collection, or collections, or of work undertaken in a given time frame. Reports should be generated in a non-proprietary format, such as CSV.

Use case number	P10
Use case class	Processor
Use case description	Processor wishes to change their password
Priority	Must
Primary actor	Processor
Secondary actor	N/A
Steps	1. User logs in to Cairo as a Processor and is authenticated 2. Cairo asks User if they want to a. Process a new collection b. Add a new accession to an existing collection c. Continue working on a saved process d. Save work at current point e. View 'personal' workspace f. Generate reports g. Lock session/log out 3. User selects view personal workspace 4. User is presented with their 'personal' workspace page and options to: a. Change role b. Change password c. Generate reports d. View saved work e. Others? 5. User selects 'change password' option 6. User supplies password according to Cairo password rules 7. Cairo validates and confirms password change
Notes	New Users should change the password allocated them by an administrator, and other users may wish to change their password for security reasons.

Use case number	P11
Use case class	Processor
Use case description	User wishes to ‘suspend’ their login whilst Cairo tool performs a long task
Priority	Must
Primary actor	Processor Administrator
Secondary actor	N/A
Steps	1. A User logs in to Cairo and is authenticated 2. Administrator/Processor initiates Cairo tool process 3. Administrator/Processor wishes to leave process running but to ‘lock’ their account 4. Administrator/Processor invokes logout process 5. Administrator/Processor selects ‘Lock account’ option 6. Administrator/Processor account is locked 7. To unlock account the login process is invoked and authentication is required
Notes	Some Cairo process may take some time to run. Users should be able to let the system run in a secure but ‘unattended’ mode whilst they carry out other work. This could be a similar process to a system time out.

Use case number	P12
Use case class	Processor
Use case description	Processor wishes to view a list of their work saved by the Cairo tool
Priority	Must
Primary actor	Processor Administrator
Secondary actor	N/A
Steps	1. User logs in to Cairo and is authenticated 2. User is presented with Cairo tool ‘home page’ 3. User selects ‘View saved work’ option 4. Cairo presents list of all work saved by that User 5. User chooses to sort work by various criteria, e.g. date, work number, etc. 6. User clicks on individual work item 7. Cairo tool opens that work item for User
Notes	

Use case number	P13
Use case class	Processor
Use case description	Processor wishes to view a list of all work saved by the Cairo tool
Priority	Must
Primary actor	Processor Administrator
Secondary actor	N/A
Steps	1. User logs in to Cairo as a Processor and is authenticated 2. Cairo asks user if they want to a. Process a new collection b. Add a new accession to an existing collection c. Continue working on a saved process d. Save work at current point e. View ‘personal’ workspace f. Generate reports g. Lock session/log out 3. User changes their system privilege level from ‘Processor’ to Administrator’ (see usecase un ‘User wishes the change their level of system permission’) 4. Processor selects ‘generate reports’ 5. Processor supplies report parameters 6. Cairo generates report
Notes	

Use case number	P14
Use case class	Processor
Use case description	Processor wishes to view a list of all the accessions to a specified archive or collection
Priority	Should
Primary actor	Processor
Secondary actor	N/A
Steps	1. User logs in to Cairo as a Processor and is authenticated 2. Cairo asks User if they want to a. Process a new collection b. Add a new accession to an existing collection c. Continue working on a saved process d. Save work at current point e. View 'personal' workspace f. Generate reports g. Lock session/log out 3. User selects 'Generate reports' 4. User supplies report parameters 5. Cairo generates report.
Notes	The 'Generate reports' option should allow certain query/report parameters to be run by Processors, and others to be limited to Administrators as determined by the 'privileges' set for their level of use. This is verging on using Cairo as a means to query the repository and may be lower on the priority list than some ingest-related features.

Use case number	P15
Use case class	Processor
Use case description	Processor wishes to view a list of all the accessions of material which have come from a specified donor or depositor
Priority	Must
Primary actor	Processor
Secondary actor	N/A
Steps	1. User logs in to Cairo as a Processor and is authenticated 2. Cairo asks User if they want to a. Process a new collection b. Add a new accession to an existing collection c. Continue working on a saved process d. Save work at current point e. View 'personal' workspace f. Generate reports g. Lock session/log out 3. User selects 'Generate reports'. 4. User supplies report parameters. 5. Cairo generates report.
Notes	

Use case number	P16
Use case class	Processor
Use case description	Processor wishes to view a list of work processed in a given time period
Priority	Must
Primary actor	Processor
Secondary actor	N/A
Steps	1. User logs in to Cairo as a Processor and is authenticated 2. Cairo asks User if they want to a. Process a new collection b. Add a new accession to an existing collection c. Continue working on a saved process d. Save work at current point e. View 'personal' workspace f. Generate reports g. Lock session/log out 3. User selects 'Generate reports'. 4. User supplies report parameters. 5. Cairo generates report.
Notes	This might be useful for compiling statistics

Usecases - Administrator

Typically the administrator will configure the tool's default behaviours, add new Users to the tool and manage existing Users.

Summary

Use Case	**Administrator Use case description.**
A1	Administrator wants to add new User
A2	Administrator wants to delete a User
A3	Administrator wants to modify a User's details
A4	Administrator wants to configure which metadata to include for objects
A5	Administrator wants to add a file format to a content model so that it inherits the metadata configuration assigned to that content model in the tool's current configuration
A6	User forgets password/username and Administrator must reset these details
A7	Administrator wishes to reset Cairo tool basic defaults
A8	Administrator wants to create new content model
A9	Administrator wishes to generate a report of all the archivists who have worked on processing a specified collection or archive
A10	Administrator wishes to modify 'privileges' assigned to a given user level, e.g. Processor

Detail

Use case number	A1
Use case class	Administrator
Use case description	Administrator wants to add new User
Priority	Must
Primary actor	Administrator
Secondary actor	N/A
Steps	1. User logs in to Cairo as a Processor and is authenticated 2. User changes their system privilege level from 'Processor' to Administrator' (see usecase P8 'User wishes to change their level of system permission') 3. Administrator selects 'Manage Users' 4. Administrator selects 'Add new User' 5. Administrator enters details of new User 6. Administrator sets appropriate 'privileges' for new User, e.g. whether Processor or Administrator 7. Administrator sets password for new User 8. Cairo checks for duplication of existing User 9. (Cairo will notify of duplicates and provide options/confirmation to proceed or cancel) 10. Cairo asks for confirmation on completion 11. Administrator logs out
Notes	Notes: each User needs to authenticate so that Cairo can use this information to record the human agent responsible for events. This will be added to the audit trail in <premis:events> in the <digiprovMD> in a dedicated event object METS file; the user will also have an agent record in the repository which can be linked to. When an Administrator logs into the system the different functions of an Administrator will appear as additional options on their 'personal' User page, perhaps as an 'administrator' tab. The assumption is that all Administrators will log-in as Processors and will undertake Administrator functions in an 'Administrator tab' once they have changed role from 'Processor' to 'Administrator' - some graphical indication (e.g. change of theme) may be useful to emphasise change of role.

Use case number	A2
Use case class	Administrator
Use case description	Administrator wants to delete a User
Priority	Must
Primary actor	Administrator
Secondary actor	N/A
Steps	1. User logs in to Cairo as a Processor and is authenticated 2. User changes their system privilege level from 'Processor' to Administrator' (see usecase P8 'User wishes to change their level of system permission') 3. Administrator selects 'Manage Users' 4. Administrator selects 'Delete User' option 5. Administrator selects 'User' to be deleted from list 6. Administrator invokes 'Delete' option 7. System asks for confirmation 8. Administrator logs out
Notes	A User may be deleted since their details will have been passed to the audit trail and will be held in the METS record for any object for which they were an agent or actor. Cairo is not a repository of metadata associated with the object.

Use case number	A3
Use case class	Administrator
Use case description	Administrator wants to modify a User
Priority	Must
Primary actor	Administrator
Secondary actor	N/A
Steps	1. User logs in to Cairo as a Processor and is authenticated 2. User changes their system privilege level from 'Processor' to Administrator' (see usecase P8 'User wishes to change their level of system permission') 3. Administrator selects 'Manage Users' 4. Administrator selects User 5. Administrator modifies details for User 6. Cairo asks for confirmation 7. Administrator logs out
Notes	Notes: User details can change and new details be added to audit trail from this point forward. There will be no retrospective modification of past User details to reflect changes.

Use case number	A4
Use case class	Administrator
Use case description	Administrator wants to configure which metadata to include for objects (e.g. which metadata should be added to objects subscribing to the ‘text’ content model)
Priority	Must
Primary actor	Administrator
Secondary actor	N/A
Steps	1. User logs in to Cairo as a Processor and is authenticated 2. User changes their system privilege level from ‘Processor’ to Administrator’ (see usecase P8 ‘User wishes to change their level of system permission’) 3. Administrator selects ‘Manage content model configuration’ 4. Administrator can choose from ‘Work with existing content models’ and ‘Create new content model’ 5. Administrator selects ‘Work with existing content model’. 6. Cairo presents list of existing content models and Administrator selects ‘text’ from this list 7. Administrator is presented with a screen on the ‘text’ content model, which includes the metadata profile for the content model, and a list of the formats which are to be treated as subscribing to the ‘text’ content model. 8. Administrator selects appropriate metadata 9. Cairo offers option to save configuration as ‘Default’ or as a different content model 10. Administrator inputs details 11. Cairo seeks confirmation of changes
Notes	

Use case number	A5
Use case class	Administrator
Use case description	Administrator wants to add a file format to a content model so that it inherits the metadata configuration assigned to that content model in the tool's current configuration
Priority	Must
Primary actor	Administrator
Secondary actor	N/A
Steps	1. User logs in to Cairo as a Processor and is authenticated 2. User changes their system privilege level from 'Processor' to Administrator' (see usecase P8 'User wishes to change their level of system permission') 3. Administrator selects 'Manage content model configuration' 4. Administrator can choose from 'Work with existing content models' and 'Create new content model' 5. Administrator selects 'Work with existing content model'. 6. Cairo presents list of existing content models and Administrator selects 'text' from this list 7. Administrator selects 'Add new file format' option 8. Administrator completes details of new format to be added to 'text' content model 9. Cairo seeks confirmation of changes
Notes	'Old' file formats cannot be deleted from a content model since a given file format (even if obsolete) may appear in a future accession, and objects of that format may already exist in the archive. Use case A5 would also cover adding a new version of a file format. Different versions of the same format should appear as stand alone objects; the dependency between versions of the same format (e.g. MS Word, 2,3,4,6,9, etc.) means that if one is included then all should be by default.

Use case number	A6
Use case class	Administrator
Use case description	User forgets password/username and administrator must reset this
Priority	Must
Primary actor	Administrator
Secondary actor	N/A
Steps	1. User forgets username or password 2. User logs in to Cairo as a Processor and is authenticated 3. User changes their system privilege level from ‘Processor’ to Administrator’ (see usecase P8 ‘User wishes to change their level of system permission’) 4. Administrator selects ‘Manage Users’ 5. Administrator selects User 6. Administrator modifies details for User – resetting password (when user logs in, they should be prompted to change password) 7. On completion Administrator logs out
Notes	

Use case number	A7
Use case class	Administrator
Use case description	Administrator wishes to reset Cairo tool basic defaults
Priority	Must
Primary actor	Administrator
Secondary actor	N/A
Steps	1. User encounters configuration problems with Cairo tool 2. User logs in to Cairo as a Processor and is authenticated 3. User changes their system privilege level from 'Processor' to Administrator' (see usecase P8 'User wishes to change their level of system permission') 4. Administrator selects 'Tool configuration' 5. Administrator selects 'Reset basic defaults' 6. (If others are currently logged in and using Cairo system informs Administrator and disallows action until all Users logged out) 7. Administrator is asked for confirmation of action 8. Administrator confirms or rejects action 9. Administrator is informed that system is returned to a default condition 10. On completion Administrator logs out
Notes	This manual reset may be useful if a User is encountering problems, or cannot determine what the current configuration settings may apply

Use case number	A8
Use case class	Administrator
Use case description	Administrator wants to create new content model
Priority	Must
Primary actor	Administrator
Secondary actor	N/A
Steps	1. User logs in to Cairo as a Processor and is authenticated 2. User changes their system privilege level from 'Processor' to Administrator' (see usecase P8 'User wishes to change their level of system permission') 3. Administrator selects 'Manage content model configuration' 4. Administrator can choose from 'Work with existing content models' and 'Create new content model' 5. Administrator selects 'Create new content model'. 6. Administrator enters details of new content model, formats covered by that content model, and metadata configuration assigned to that content model. 7. Cairo seeks confirmation of changes
Notes	

Use case number	A9
Use case class	Administrator
Use case description	Administrator wishes to generate a report of all the archivists who have worked on processing a specified collection or archive
Priority	Should
Primary actor	Administrator
Secondary actor	N/A
Steps	1. User logs in to Cairo as a Processor and is authenticated 2. User changes their system privilege level from 'Processor' to Administrator' (see usecase P8 'User wishes to change their level of system permission') 3. Administrator selects 'Generate reports' 4. Administrator supplies report parameters 5. Cairo generates report
Notes	This function might be useful for auditing/tracking purposes. Searching on these parameters might be most appropriately restricted to the Administrator level .

Use case number	A10
Use case class	Administrator
Use case description	Administrator wishes to modify ‘privileges’ assigned to a given user level, e.g. Processor
Priority	Must
Primary actor	Administrator
Secondary actor	
Steps	1. User logs in to Cairo as a Processor and is authenticated 2. User changes their system privilege level from ‘Processor’ to Administrator’ (see usecase P8 ‘User wishes to change their level of system permission’) 3. Administrator selects ‘Manage Users' 4. Administrator selects ‘Modify user level' 5. Administrator modifies privileges by changing User from Processor to Administrator 6. Cairo asks for confirmation of changes
Notes	Processors may become Administrators.

Usecases - Sysadmin

The Sysadmin will typically be the person who manages the infrastructure on which the Cairo tool and the digital object repository resides. They will not be 'Administrator' or 'Processor' users of the system.

The Sysadmin should not require a separate User account on the system in order to perform this role.

Summary

Use Case	**Sysadmin Use case description.**
SA1	Sysadmin wants to install Cairo
SA2	Sysadmin creates first 'Administrator' user
SA3	Sysadmin installs updates

Detail

Use case number	SA1
Use case class	Sysadmin
Use case description	Sysadmin wants to install Cairo
Priority	Must
Primary actor	Sysadmin
Secondary actor	Administrator
Steps	1. Sysadmin reads Cairo tool documentation 2. Sysadmin prepares environment according to documentation 3. Sysadmin obtains install files (from supplied CD/Sourceforge website) 4. Sysadmin follows documented install procedures accepting default values 5. Sysadmin checks install procedure has happened correctly
Notes	The installation should be as simple as possible to allow for organisations with limited technical support. We would prefer that there is an installer for Unix/Linux and Windows environments.

Use case number	SA2
Use case class	Sysadmin
Use case description	Sysadmin creates first ‘Administrator’ user
Priority	Must
Primary actor	Sysadmin
Secondary actor	Administrator
Steps	1. Sysadmin installs Cairo 2. During install process Sysadmin is asked to create the first ‘Administrator’ 3. Sysadmin is presented with ‘Manage Users – add new User’ screen 4. Sysadmin enters details of ‘Administrator’ 5. Sysadmin enters password for ‘Administrator’ 6. Sysadmin is asked to confirm ‘Administrator’ password 7. Cairo tool creates ‘Administrator’ user
Notes	This stage happens at some convenient point during the installation process. Installation cannot be complete until a user level ‘Administrator’ has been created.

Use case number	SA3
Use case class	Sysadmin
Use case description	Sysadmin installs updates
Priority	Must
Primary actor	Sysadmin
Secondary actor	Administrator
Steps	1. Sysadmin logs in to system on which Cairo tool is hosted 2. Sysadmin reads Cairo tool update documentation 3. Sysadmin prepares environment according to documentation 4. Sysadmin copies update files from supplied CD 5. Sysadmin follows documented install procedures accepting default values 6. Sysadmin checks install procedure has happened correctly
Notes	The tool may need to be updated in light of changes to tool modules, support for new metadata standards or further development of the Cairo interface itself. It will be important that any update, or new installation allows for the import of existing configuration and User information.

Usecases - Developer

The developer may be external to the archival organisation using the tool. On an ad hoc basis they may require all levels of user access to the system in creating and testing new or additional functionality. They will interact with both the sysadmin and the administrator to perform any required task.

Summary

Use Case	**Developer Use case description.**
D1	Developer wants to add new metadata mappings to Cairo Tool
D2	Developer needs a guide on getting Started with Cairo Tool development
D3	Developer needs to setup a Development Environment
D4	Developer wants to configure their Code repository setup (e.g. SVN)
D5	Developer wants to configure their Integrated Development Environment (IDE), run and debug settings (e.g. Eclipse)
D6	Developer wants to commit changes to master Cairo code repository (e.g. Sourceforge SVN)
D7	Developer wants to work with code versions/revisions
D8	Developer wants to setup bug tracking account (e.g. via Sourceforge)
D9	Developer needs to know development conventions and guidelines
D10	Developer needs to have regression testing performed

Details

Use case number	D1
Use case class	Developer
Use case description	Developer wants to add new metadata mappings to Cairo tool
Priority	Must
Primary actor	Developer
Secondary actor	Sysadmin User
Steps	1. Developer opens Cairo source files 2. Developer copies new components to default locations 3. Developer edits Cairo XML tool mapping config file 4. Developer saves work 5. Developer compiles code 6. (Developer installs new metadata mappings) 7. Developer tests new metadata mappings 8. Developer edits code if necessary 9. Developer gives new compiled code to Sysadmin for installation locally
Notes	A developer may wish to add the functionality of a new metadata extraction tool to the Cairo system. The system should allow for mapping the output of new tools to the basic content models defined by the tool.

Use case number	D2
Use case class	Developer
Use case description	Developer needs a guide on getting Started with Cairo Tool development
Priority	Must
Primary actor	Developer
Secondary actor	N/A
Steps	Developer accesses project Sourceforge site to find links to Development Resources: ● developers guide ● reporting bugs ● getting code ● browsing code ● committing code ● coding guidelines ● project plan
Notes	The developer is interested in becoming involved with the development or enhancement of the tool and needs to know what is required. The guide needs to cover the steps necessary to begin development on the Cairo Tool using, for example Eclipse, SVN, and Bugzilla.

Use case number	D3
Use case class	Developer
Use case description	Developer needs to how to setup a Development Environment
Priority	Must
Primary actor	Developer
Secondary actor	N/A
Steps	Developers Guide Environment section contains: ● pre-requisites ● list of components to download and links ● setting up a development sandbox ● setting up a project integration sandbox ● step by step instructions ● hints
Notes	Sandbox – is a term for a controlled technical environment whose scope is well defined. Developers work within their own individual sandbox.

Use case number	D4
Use case class	Developer
Use case description	Developer wants to configure their code repository (SVN) setup
Priority	Must
Primary actor	Developer
Secondary actor	N/A
Steps	Developers Guide Code repository section contains: ● how to connect to repository ● checking out source code ● step by step instructions ● hints
Notes	

Use case number	D5
Use case class	Developer
Use case description	Developer wants to configure their IDE run and debug settings (e.g. Eclipse)
Priority	Should
Primary actor	Developer
Secondary actor	N/A
Steps	Developers Guide Integrated Development Environment (IDE) tool section contains: ● how to launch the IDE (e.g. Eclipse) ● how to configure run settings ● how to configure debug settings ● checking out source code ● step by step instructions ● hints
Notes	Most IDE's allow coding testing and debugging we would expect to document ours (probably Eclipse) but Netbeans et al developers could augment?

Use case number	D6
Use case class	Developer
Use case description	Developer wants to commit changes to master Cairo code repository (e.g. Sourceforge SVN)
Priority	Must
Primary actor	Developer
Secondary actor	N/A
Steps	Developers Guide Committer section contains: ● how to commit (e.g. Eclipse) ● how to undo (revert) ● how to synchronise ● checking out source code ● step by step instructions ● hints
Notes	Assumption is other IDE based developers would contribute to the documentation here

Use case number	D7
Use case class	Developer
Use case description	Developer wants to work with specific code versions/revisions
Priority	Should
Primary actor	Developer
Secondary actor	N/A
Steps	Developers Guide Committer section contains: ● how to examine revisions (e.g. Eclipse) ● retrieve specific revisions source code ● step by step instructions ● hints
Notes	Assumption is other IDE based developers would again contribute to the documentation here

Use case number	D8
Use case class	Developer
Use case description	Developer wants to setup Bug tracking account
Priority	Should
Primary actor	Developer
Secondary actor	N/A
Steps	Developers Guide Bugs section contains: ● how to create an account ● how to submit bugs ● how to submit change requests ● step by step instructions ● hints
Notes	Assumption is other IDE based developers would again contribute to the documentation here

Use case number	D9
Use case class	Developer
Use case description	Developer needs to know development conventions and guidelines
Priority	Must
Primary actor	Developer
Secondary actor	N/A
Steps	Developers Guide Resources section - Development Conventions and Guidelines: ● Naming Conventions - How to name things like packages, classes, and methods ● Coding Conventions - How to layout Java code/ XML code ● Javadoc - How to write documentation comments, especially for API ● User Interface Guidelines - How to achieve user interface consistency ● Version Numbering - How to evolve plug-in version numbers
Notes	In any project there are areas where standards, conventions and other guidelines play a role in ensuring that the results present a unified whole rather than simply a collection of parts. In many respects these are already documented and actively in use in existing projects such as Sun Java and Eclipse.

Use case number	D10
Use case class	Developer
Use case description	Developer needs to have regression testing performed
Priority	Should
Primary actor	Sysadmin/User
Secondary actor	Developers
Steps	Developers Guide Testing section – Regression Testing: ● Developer setting up System and Acceptance test environments (Test/QA sandbox) ● setting up a demo sandbox ● Sysadmin install (use case S1) ● Sysadmin/User perform the tests ● Sysadmin/User report the bugs
Notes	Production systems cannot act as the final test. It's too late to find bugs there. A fully fledged production test environment with well understood data not all perfect (deliberate well known issues) is required to act as the final stage prior to sign off of a production release.

Usecases - System

The system will have some automated functionality by which it passes information back to the different users, most typically the Administrator and the Processor. This includes such things as notification of error conditions, completion of requested tasks etc.

Summary

Use Case	**System Use case description.**
S1	Personal user screens
S2	System monitors saved work
S3	System timeout
S4	Autosave
S5	System record locking on unexpected event
S6	System has not successfully processed an archive
S7	System validates XML METS files
S8	System detects/encounters error in carrying out usecase U1
S9	System detects/encounters error in using an external service
S10	System detects validation failure in completed METS file
S11	Unauthenticated user attempts to login
S12	System logs action
S13	System initiated authentication checking of User entering system
S14	System initiates timeout on a Users login that has been inactive for *n* period of time
S15	System automatically assigns ‘ownership’ of a process to the User who initiated that process
S16	Cairo should be able to connect to different repository instances for ingest, potentially using different repository software

Detail

Use case number	S1
Use case class	System
Use case description	Personal user screens
Priority	Must
Primary actor	System
Secondary actor	All Users
Steps	1. Each User may log in to their 'personal' work page 2. Cairo welcomes User on return 3. Screen carries User name and contact details 4. Screen lists basic action options (listed in usecase P8) 5. Screen lists current User's saved work (or option to view this) with time and date of last action on that saved work, and allows User to select and continue a piece of work listed here 6. Screen includes options to generate reports for that User 7. Screen lists any other system messages for that User
Notes	Personal User screens will allow Users to access and modify some details relating to their User account, and to view and access their own work in progress.

Use case number	S2
Use case class	System
Use case description	System monitors saved work
Priority	Must
Primary actor	System
Secondary actor	All Users
Steps	1. Authenticated User saved incomplete process 2. Cairo logs data and time of save 3. After *n* period of time elapses system sends message to User notifying of saved work 4. After *n*+ period of time if User has not completed saved work system sends message to Administrator notifying of uncompleted saved work not being completed 5. Administrator acts on message
Notes	It is important that ingest is completed in a timely manner. This ensures that the work is completed while high-level metadata relating to the collection is fresh in the archivist's mind, and allows the archivist to re-contact the donor to clarify any issues more easily. It also secures the archival material in a managed preservation repository as soon as possible. The system will therefore monitor the status of work-in-progress and alert relevant staff where work appears to have been abandoned mid-process. For non-networked repositories messaging will be through Cairo; an option to receive email notifications too or instead would be useful.

Use case number	S3
Use case class	System
Use case description	System timeout
Priority	Must
Primary actor	System
Secondary actor	All Users
Steps	1. User has logged into Cairo and been authenticated 2. User has not entered keystroke for *n* minutes 3. Cairo goes into suspension 4. User must re-authenticate to enter 5. Cairo informs User of any active/running processes
Notes	For security reasons, the system will timeout if inactive for a given period of time. System time out will not stop any Cairo tool process initiated by a User.

Use case number	S4
Use case class	System
Use case description	Autosave
Priority	Should
Primary actor	System
Secondary actor	All Users
Steps	1. User has logged into Cairo and been authenticated 2. User has been working for X amount of time without saving work 3. Cairo autosaves to guard against loss of work in event of power outage, or other failure.
Notes	Some Users may wish to have access to an autosave feature, which saves their work on a regular basis. Some Users may prefer not to use autosave, so perhaps this ought to be a User preference set in the personal User screen?

Use case number	S5
Use case class	System
Use case description	System record locking on unexpected event
Priority	Must
Primary actor	System
Secondary actor	All Users
Steps	1. User has logged into Cairo and been authenticated 2. User has been working in system 3. User quits by unexpected/inappropriate means or system unexpectedly shuts down 4. Cairo locks current record 5. (Lock must be released by User or administrator before work on that record can continue) 6. When ‘owner’ of locked record next logs in Cairo tool informs them of locked record offers them unlock option.
Notes	

Use case number	S6
Use case class	System
Use case description	System has not successfully processed an archive (See usecase P1)
Priority	Must
Primary actor	System
Secondary actor	All Users
Steps	1. Processor has completed usecase P1 2. Cairo has failed to process archive successfully 3. Cairo sends message to User 4. Cairo presents Processor with options, a. Delete unprocessed work and begin again b. Review work and correct errors 5. Processor selects option 6. Cairo seeks confirmation of Processor action
Notes	There could be a number of different errors, e.g. one particular tool not responding to a call. Ideally error reports should provide the Processor with some assistance with troubleshooting.

Use case number	S7
Use case class	System
Use case description	System validates XML METS files created in usecase P1
Priority	Must
Primary actor	System
Secondary actor	All Users
Steps	1. Processor has completed usecase P1 2. Cairo takes METS files and attempts to validate them 3. If validation, a. Successful = continue processing b. Unsuccessful = invoke unsuccessful processing error (S6) 4. Send message to Processor in either event 5. The message should list any METS file(s) that failed to validate, and allow the Processor to view the file(s) in an XML editor, where any validation error(s) can be identified and rectified. Testing the tool may identify some ‘regular’ errors, for which Cairo might suggest causes or solutions?
Notes	

Use case number	S8
Use case class	System
Use case description	System detects/encounters error in carrying out usecase P1
Priority	Must
Primary actor	System
Secondary actor	All Users
Steps	1. Processor is undertaking usecase P1 2. Cairo encounters problem/error 3. If problem/error, a. Can be internally resolved = continue processing b. Is fatal = invoke unsuccessful processing error (S10) 4. Send message to Processor in either event
Notes	

Use case number	S9
Use case class	System
Use case description	System detects/encounters error in using an external service, e.g. PRONOM
Priority	Must
Primary actor	System
Secondary actor	All Users
Steps	1. Processor has initiated usecase P1, selected/default settings require access to external service 2. Cairo encounters problem/error with that service, e.g. non-response 3. Notify Processor in either event 4. Tool presents three options a. Continue processing without external service input b. Suspend processing, saving work completed to date for completion when external service is made available c. Halt processing completely 5. If Processor selects option 4a they are invited to insert a free text note of explanation which is inserted into the METS 6. If Processor selects option 4b system invokes the save present work option 7. If Processor selects option 4c system halts all processing and deletes any work undertaken to that point 8. Cairo creates event log of incident
Notes	There could be a number of different errors, e.g. one particular tool not responding to a call; a service could be unavailable; a server may have crashed, etc. Ideally error reports should provide the Processor with some assistance with troubleshooting.

Use case number	S10
Use case class	System
Use case description	System detects validation failure in completed METS file
Priority	Must
Primary actor	System
Secondary actor	All Users
Steps	1. Processor has initiated usecase P1, and has followed process to completion 2. Cairo encounters problem/error whilst validating completed METS file 3. Cairo notifies Processor of failed validation 4. Tool presents four options a. Re-submit METS file to validation process b. Suspend processing, saving work completed to date for completion when external service is made available c. Halt processing completely d. View/edit in XML editor 5. If Processor selects option 4a they are invited to insert a free text note of explanation which is inserted into the METS 6. If Processor selects option 4b system invokes the save present work option 7. If Processor selects option 4c system halts all processing and deletes any work undertaken to that point 8. If Processor selects option 4d then Cairo sends XML to their preferred XML editor 9. System creates event log of incident
Notes	Validation of the METS file is an essential step, invalid XML negates the reasons for holding techMD in this format. Validation failure should only occur if there is either a failure of an external validation service or some form of network/system/application failure of a local service. Validation failure may be encountered after the modification or creation of a Cairo tool component.

Use case number	S11
Use case class	System
Use case description	Unauthenticated user attempts to login
Priority	Must
Primary actor	System
Secondary actor	All Users
Steps	1. User attempts login 2. User isn't recognised by system 3. Cairo generates 'Not recognised – try again' screen 4. User isn't recognised by system 5. Cairo generates 'Not recognised – try again' screen 6. Cairo loops message 5 times and times out for a period before the user can retry
Notes	What is standard operating procedure in this case? Is there a best practice/recommended model we can follow?

Use case number	S12
Use case class	System
Use case description	System logs action
Priority	Must
Primary actor	System
Secondary actor	All Users
Steps	1. User logs into system 2. Cairo records actions performed by User 3. Cairo records User settings of Cairo tool 4. User is authenticated by system 5. User uses Cairo tool 6. On conclusion of actions User is presented with log file of their actions 7. Cairo writes log file to METS as audit trail of actions carried out and settings applied by User with Cairo tool
Notes	

Use case number	S13
Use case class	System
Use case description	System initiated authentication checking of User entering system
Priority	Must
Primary actor	Sysadmin Administrator User Developer
Secondary actor	All Users
Steps	1. A User attempts to log into system 2. A User enters Username 3. A User enters password 4. A User hits ‘Log in’ button or presses Enter key 5. Cairo checks that username exists 6. If username does not exist authentication fails, User is returned to basic log in screen 7. Cairo checks that entered password matches that for username 8. If password does not match authentication fails, User is returned to basic log in screen
Notes	Authentication affects all Users of the Cairo tool.

Use case number	S14
Use case class	System
Use case description	System initiates timeout on a Users login that has been inactive for *n* period of time
Priority	Must
Primary actor	Sysadmin Administrator User Developer
Secondary actor	All Users
Steps	1. User logs into the system and is successfully authenticated 2. Cairo monitors User activity a. User initiates no process, command system action for *n* period of time b. User makes no keyboard input for *n* period of time 3. If Cairo detects no User activity after *n* period of time has elapsed system locks User account 4. Cairo continues to process tasks as/if requested prior to locking User account 5. Upon entering any keyboard input system presents User with authentication screen 6. User must re-authenticate to gain system access
Notes	Timeout affects all Users of the Cairo tool.

Use case number	S15
Use case class	System
Use case description	System automatically assigns ‘ownership’ of a process to the User who initiated that process
Priority	Must
Primary actor	Sysadmin Administrator User Developer
Secondary actor	All users
Steps	1. User logs into the system and is successfully authenticated 2. User initiates a process 3. Cairo gives process a unique number that is used to associate a User with a process 4. Cairo ‘binds’ ownership of that process with that User 5. That process can only be modified or cancelled by the owning User or the Cairo Administrator
Notes	Processors can only ‘work’ on, modify or cancel processes they ‘own’ (although they may view processes carried out by other Processors, when generating reports for example). An administrator has rights to ‘work’ on, modify or cancel processes ‘owned’ by a Processor

Use case number	S16
Use case class	System
Use case description	Cairo should be able to connect to multiple repositories
Priority	Must
Primary actor	Sysadmin Administrator User Developer
Secondary actor	All users
Steps	
Notes	Processors may need to submit materials to different repositories (potentially based on different software) and the Cairo Processor should therefore be able to specify which repository they wish to ingest material into and the type of that repository (e.g. Fedora or DSpace)