**Specification for Recording Events and Agents in DAMS object records**

**Brad Westbrook, Esme Cowles, and Declan Fleming**

**16 December 2012**

**Outline:**

**I: Event Requirements**

**II: Data Model**

**III: Process**

**IV: Object example**

**V: Reports**

**VI: Implementation**

**VII: Use Cases**

**VIII: RDF encoded example**

**I: Event Requirements**

•            Record stipulated events occurring to all digital objects managed in the DAMS. (DLP will stipulate the events to be tracked.)

* Each discrete event requires an event record. An event record will be an addition to the overall object record. The object record, in the DAMS, are all the triples for one intellectual entity, be it simple or complex.

•            For each event, record the type of event, the date the event occurred, any details of the event necessary for understanding what occurred, and the outcome of the event (Only successful events are recorded. Assessment of an event in respect to success or failure is conducted outside of the DAMS metadata system. Events that fail are reported appropriately and then repeated until success is obtained.)

* For each event, identify the agent (person, agency, or software) responsible for executing the event and, if applicable, the agent (person, agency, or software) at which the event is directed
* Link each event record to the digital object record (representation or file) to which the event occurred
* Retain in the object record a history of all recorded events, with links from digital objects to events and from events to agents
* Use PREMIS semantic units for recording events and agents and relating them to objects
* Use data values for certain data elements, as stipulated below

**II: Data Model**

Relationship Rules:

1. Every DAMS object must consist of at least one content master file, one metadata record, and structural relationship(s) between metadata and file(s). The digital object in its entirety is the representation of an intellectual entity, or work. It exists as a set of RDF triples in the UCSD DAMS.
2. Zero or more events may occur for any digital object or any of its component parts. Events can include (but need not be limited to):
	1. Replicating the digital object and transferring it to a preservation management service
	2. Modifying or replacing any of the master content files first acquired as part of the digital object
	3. Modifying any of the original (e.g., descriptive, file (master), or rights) metadata for the digital object
	4. Augmenting the original digital object by adding additional master content files and / or additional (e.g., descriptive, file (master), or rights)metadata
3. One or more agents may be responsible for any given event. Agents may include (but need not be limited to):
	1. Persons
	2. Organizations
	3. Repository
	4. Software

Recording Rules for Events and Agents:

1.0: Each distinct event to be tracked in the DAMS will be recorded as a separate segment (i.e., event record) of the object’s complete metadata record. (In the context of METS documents, each distinct occurrence of an event should be recorded as event metadata in the <digiprovMD> container within the METS record for the object. Bear in mind that in a different implementation the complete metadata record may be distributed across different databases, with, for example, the descriptive, file, and rights metadata being recorded in the object triple store; the event metadata in a different database, triples or SQL; and the agent metadata in yet another database, triples or SQL. )

1.1 Each event will have a unique identifier. It will consist of the A) ARK for the object, B) the word “event”, and C) the date stamp for when the event occurred, e.g., [http://libraries.ucsd.edu/ark:/20775/bb25609271-event-2007-08-21T15:16:50](http://libraries.ucsd.edu/ark%3A/20775/bb25609271-event-2007-08-21T15%3A16%3A50)

2.0 Each distinct agent associated with an event for an object will be linked to from the event record within the object.

2.1 The link from the event record to the agent will include a declaration of the agent’s role in respect to the event.

2.2 Agent records per se will exist outside of the object record.

2.3 There will be one agent record for each unique agent.

2.4 Each agent record should be explicitly linked from every event to which it is associated. One agent record may, and will undoubtedly be, linked from many event records.

2.5 Agent records themselves do not link to event records; they are always linked from event records.

3.0 Although the event record is embedded in the object record, each event record should be explicitly linked to from the appropriate part of the object record.

3.1 If an event occurs at the representation level, then a new object sub-record needs to be added to the digital object record. That new record will consist of:

* object/objectIdentifierType (ARK)
* object/objectIdentifierValue (ARK for the digital object record, less any extension; same as ARK in “OBJID” in declaration.
* objectCategory (Representation)
* linkingEventIdentifier/linkingEventIdentifierType
* linkingEventIdentifier/linkingEventIdentifierValue

3.2 If the event occurs at the file level, then the following will be added to the already extant object record for the file to which the event applies:

* linkingEventIdentifier/linkingEventIdentifierType
* linkingEventIdentifier/linkingEventIdentifierValue

Data Diagram:

**Intellectual Entities**

Content that be described as a unit (e.g., painting, book, song, movie, data set, etc.)

**Rights**

Assertion of copyright, statutory rights, license rights, and permissions/restrictions

**Agents**

People, organizations, or software, having rights to the content / objects or responsibility for events occurring to the content / objects.

**Objects**

Discrete units of information in digital form. Can be in form of representation, files, or bitstreams

**Events**

Actions involving object(s) and agent(s) known to the system.

Data Relationships:

* One intellectual entity can have one or more representations. Each representation corresponds to a digital object record in the UCSD DAMS.
* Each representation can be factored into component parts, consisting of sub-representations, files, or bitstreams—all resolving to a single representation of the intellectual entity.
* An object or any of its component parts may have one or more rights statements asserted for it. The rights statement may pertain to copyright, statutory rights, or license rights.
* An object or any of its component parts may have one or more events occurring to it
* A rights assertion may be for the complete object or for a component part of the object.
* A rights assertion may be associated with one or more agents. The agent is assumed to be the “rights holder” in some fashion.
* An event may occur for the entire object (representation) or for a component part (file)
* An event may have one or more agents associated with it. The agent is assumed to be the executor of the event (e.g. transmitted files) or the recipient of the event (e.g., received files). All events have at least one executor, but not all events need have recipients.
* An agent associated to a rights assertion or to an event report always has an explicit role for that association.
* Agents are linked to from rights or events, but they do not themselves link to rights or events.

**Semantic Units for Describing Events:**

Legend:

* Entity: for indicating the semantic unit is part of the Object, Event, or Agent (The relationships of Agents to Rights and of Rights to Objects is not accounted for in this table)
* Semantic unit: The PREMIS name for the data element.
* Definition: The PREMIS definition for the entity, with some modifications for the DAMS environment
* Req: Whether or not the semantic unit is required for each occurrence of an event report. Required refers to the PREMIS schema but also to UCSD usage. Thus metadata may not be required by the PREMIS schema, but may be required for all UCSD digital object records
* Rep: Whether or not the semantic unit can be repeated within a report for an event
* Controlled values: The values that are permissible for a semantic unit. In this document all the controlled values listed are proposed controlled values, although some are derived from the PREMIS standard.

For the UCSD DAMS:

* Object metadata is created for every master file comprising a digital object. Object metadata is not required for derivatives or for individual files bundled in a tar/gz file
* Event metadata is required for each occurrence of an event type that DAMS management stipulates to be tracked in the digital object record. These events are (though not limited to):
* Agent metadata is required for each unique agent implicated in the events described in the event metadata.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Entity** | **Semantic Unit** | **Definition** | **Req.** | **Rep.** | **Controlled values** |
| **Object** | Object metadata is created for every master file comprising a digital object. Object metadata is not required for derivatives, or for individual files bundled in a tar/zip file.In the DAMS, an event will typically occur at either the representation or file level. If it occurs at the representation level, a new object sub-record needs to be added to the digital object record. That new record will consist of:* object/objectIdentifierType (ARK)
* object/objectIdentifierValue (ARK for the digital object record, less any extension; same as ARK in “OBJID” in declaration.
* objectCategory (Representation)
* linkingEventIdentifier/linkingEventIdentifierType
* linkingEventIdentifier/linkingEventIdentifierValue

If the event occurs at the file level, then the following will be added to the already extant object record for the file to which the event applies:* linkingEventIdentifier/linkingEventIdentifierType
* linkingEventIdentifier/linkingEventIdentifierValue
 |  |  |  |
|  | obect/objectIdentifierType | The type of identifier used for either the representation of an intellectual entity or a component file of that representation | YES | No | ARK |
|  | objectIdentifier/objectIdentifierValue | The actual ARK for the digital object or for one of its files | Yes | No | “http://libraries.ucsd.edu/ark:/20775/[Value]” |
|  | objectCategory | The category to which the event pertains, typically for UCSD assets the representation as a whole or to a particular file or set of files comprising the representation.  | Yes | No | FileRepresentation |
|  | objectCharacteristics | The technical properties for file or bit stream. (Object characteristics does not apply to the category of Representation.)  |  |  |  |
|  | objectCharacteristcis/compositionLevel | An indication if the object is subject to one or more processes of decoding or unbundling | YES | No |  |
|  | objectCharacteristics/fixity | Information used to monitor whether an object has been altered or not. Applicable to file and bitstream categories only.  | YES | No |  |
|  | objectCharacteristics/fixity/messageDigestAlorithm | Identification of the algorithm used to generate the value expressed in the messageDigest | Yes | No | MD5SHASHA-256 |
|  | objectCharacteristics/fixity/messageDigest | The value generated using the algorithm identified in messageDigestAlgorithm | YES | No |  |
|  | objectCharacteristics/size | The size in bytes of the file or bitstream. Also used to verify fixity.  | YES | No |  |
|  | objectCharacteristics/format/formatDesignation/formatName | The name of the file format, e.g., TIFF | YES | No |  |
|  | objectCharacteristics/format/formatDesignation/formatVersion | The version of the file format, e.g., “6.0” for TIFF 6.0 | YES | No |  |
|  | objectCharacteristics/creatingApplication/dateCreatedByApplication | The date the file was created | Yes | No |  |
|  | linkingEventIdentifier/linkingEventIdentifierType |  | YES | No | Required for each event. |
|  | linkingEventIdentifier/linkingEventIdentifierValue |  | YES | No | Required for each event linked to  |
|  |  |  |  |  |  |
| **Event** | Each event for a digital object is linked to from the appropriate place in the object. If the event is for the entire digital object, a new object record sub-record is added to the digital object, for which the mandatory elements are listed above. The new object record will include a link to the event record. If the event is for a file comprising the digital object, a link is established from the object record for that file to the event record.  |  |  |  |
|  | eventIdentifier/eventIdentifierType | A designation of the domain within which the event identifier is unique. The domain is the UCSD DAMS.  | YES | NO | cu-s |
|  | eventIdentifier/eventIdentifierValue | An alphanumeric string used to uniquely identify the event within the domain context. Note: The eventIdentifierType and eventIdentifierValue is mandatory for each event record and uniquely identifies the event within the DAMS. An identifier of the type “ark:/20775/bb9693126t**-event-** YYYY-MM-DDTHH:MM:SS” uses a pattern for the identifier similar to other patterns used in the object record and relates the event strongly to the object via the same parent ARK. Using the date stamp as a suffix will help to disambiguate events and chronologically sequence them within the context of a certain object.  | YES | NO | Object ARK with hyphenated suffix for the event, e.g., “ark:/20775/bb9693126t**-event-** YYYY-MM-DDTHH:MM:SS” |
|  |  |  |  |  |  |
|  | eventType | A categorization of the event. A list of event types is provided by the Library of Congress (<http://id.loc.gov/vocabulary/preservationEvents.html>)“ingestion” (<http://id.loc.gov/vocabulary/preservationEvents/ingestion>) is the term to be used for transmitting objects to a preservation service. “Ingestion” to another repository implicitly entails sub-events such as fixity check, replication, validation, and virus check—in short all actions provided by the preservation repository’s ingest process. | Yes | NO | LoC Glossary:capturecompressioncreationdeaccessiondecompressiondecryptiondeletiondigital signature validationfixity checkingestionmessage digest calculationmigrationnormalizationreplicationvalidationvirus checkLocal supplements:? |
|  | eventDateTime | The date and time the event occurred. Since the event record reflects both the initiation of the event and its outcome, the date recorded here should be for when the outcome is determined.  | YES | NO | YYYY-MM-DDTHH:MM:SS |
|  |  |  |  |  |  |
|  | eventDetail | Each event type should have a basic descriptive formula associated with it, which restates the event type, associates the event type to an agent, and explicitly names all event sub-types occurring as part of the primary event. Thus, the detail for ingestion might be:“Ingestion, to Chronopolis Preservation System, including replication, fixity check, and validation of the transmitted object. Tag/gz transfer package built on YYYY-MM-DDTHH:MM:SS. Checksum type =”SHA-256”; Checksum value=”????????”; Size=”NNNNNNN””This would be the place to make implications explicit.  |  | NO |  |
|  |  |  |  |  |  |
|  | eventOutcomeInformation/eventOutcome | A list of terms for describing the outcome of an event. The list should be a controlled list, although it will grow over time. It can also be a coded list, if that is more desirable. For ingestion, and probably for most events, one term will suffice: 1) Action succeeded | NO | YES | Action succeeded |
|  | eventOutcomeInformation/eventOutcomeDetail/eventOutcomeDetailNote | A place for recording a detailed report of an event outcome, if needed. The date that the successful report is received could be record here. E.g.:“Chronopolis reported successful validation of transfer package on YYYY-MM-DD”  | NO | YES |  |
|  | eventOutcomeInformation/eventOutcomeDetail/eventOutcomeDetailExtenssion | A container field for adding event semantic units, if warranted.  |  |  |  |
|  |  |  |  |  |  |
|  | linkingAgentIdentifier | An ARK assigned to the agent.  | YES | YES | ark:/20775/nnnn-DAMSAGENT |
|  | linkingAgentIdentifier/linkingAgentIdentifierType | A term that indicates the domain in which the linkingAgentIdentifierValue is unique | YES | NO | cu-s |
|  | linkingAgentIdentifier/linkingAgentIdentifierValue | A string uniquely identifying the agent within in the specificied domain. | YES | No |  |
|  | linkingAgentIdentifier/linkingAgentRole | The role of the agent in relation to the event. Events can involve more than one agent. Each agent may have a different role. We will use the MARC relator controlled vocabulary where possible, and supplement if necessary |  | No |  |
|  |  |  |  |  |  |
|  | linkingObjectIdentifier |  | YES | No |  |
|  | linkingObjectIdentifier/linkingObjectIdentifierType | A designation of the domain in which the object identifier value is unique.  | YES | No | An event that occurs at the representation level will link to the digital object as a whole. The link will reference the object identifier (OBJID).An event that occurs at the file level will link to the file(s) to which the event occurred. The link will reference the file identifier, which is the object identifier with a file extension.  |
|  | linkingObjectIdentifier/linkingObjectIdentifierValue | An identifier for the digital object. The identifier must be unique within the stated domain.  | YES | No |
|  | linkingObjectRole | The role the object has in respect to the event. For transmission of a DAMS object to preservation management, the role of the object would be “source”.  |  | No |
|  |  |  |  |  |  |
| **Agent** |  |  |  |  |  |
|  | agentIdentifier/agentIdentifierType | A designation of the domain in which the agent identifier is unique | YES | NO | cu-s |
|  |  |  |  |  |  |
|  | agentIdentifier/agentIdentifierValue | A string uniquely identifying the agent within the specified domain. | YES | NO | ark:/20775/nnnn-DAMSAGENT |
|  |  |  |  |  |  |
|  | agentName | A text string used to resolve the value provided in agentIdentifierExamples: Gabriela MontoyaChronopolis Preservation RepositoryoXygen 12.0 | YES | YES |  |
|  |  |  |  |  |  |
|  | agentType | A term categorizing the agentHow granular would we want this set of terms to be?  | NO | NO | * ucsd dams staff
* organization
* repository
* software
* vendor
 |
|  |  |  |  |  |  |
|  | agentNote | Any additional information about the agent worthy of recording. One might include the name and version of a software agent here, or information for configuring the software agent | NO | YES |  |
|  |  |  |  |  |  |
|  | agentExtension | A container for adding additional semantic units | NO |  |  |
|  |  |  |  |  |  |

**III: Process**

The particulars of the process will be determined by the implementation and the use case for the event. At this point in time, we have agreed that events will be recorded within the object record and will become part of the object record. Each event will utilize the object’s ARK as the basis of the event’s unique identifier. Agents, on the other hand, will be recorded outside of the object record and will be linked to from either the event or rights record within the object. The agent records will utilize a new series of ARKs.

The following process description should be generally true for all use cases and implementations. The process uses PREMIS XML encoding to illustrate the metadata that is recorded and related for documenting an event to a digital object. Identifiers used in the examples are provisional.

1. An event record is created whenever an event that is tracked by DAMS management occurs. (DAMS management should decide in advance and establish a taxonomy for the events that need to be tracked. The taxonomy should distinguish between events at the representation level and events at the file level). Categorically, some of those events might be:
* **Replicating** the digital object and transferring it to a preservation management service
* **Modifying** or **replacing** any of the master content files first acquired as part of the digital object
* **Modifying** any of the original metadata for the digital object
* **Augmenting** the original digital object by adding additional master content files and / or additional descriptive, file (master), or rights metadata

The event record also reflects the outcome of the event. A log report (or snippet thereof) could form part of the outcome report.



1. The event record is linked to the digital object record (if the event is for the representation) or to the digital object component file record (if the event is for one or more component files comprising the digital object)

Representation:



File:



1. The event record is linked to the agent record for each agent associated with the event. The agent’s role in respect to the event is asserted in the link.



1. The link resolves to a record of a unique agent. If an agent record does not exist for an agent responsible for or associated with the event, then a new agent record is created.




2. A reciprocal link is made from the digital object record to the event record. A link is added for each new event.

If the event is for the representation, a new object sub-record is inserted into the digital object record.



If the event is for a file comprising part of the digital object, then the link is inserted into object record for that file. Note: object records are retained only for master files in the digital object record. Thus, events will not typically be recorded for derivative files.



1. A completed events record might appear as follows:



 **IV: Object example**

See attached file: 20775-bb4676790q. Understand that the agent records in the attached object record would actually be outside the object record.

**V: Reports**

Sometimes event metadata will be recorded in batches as the result of periodic processes applied to the whole collection, such as transfer of the collection to a preservation repository. More often events will occur singly for a single object or small groups of objects. Regardless, it should be possible to generate diverse reports utilizing the event / agent metadata recorded in the DAMS digital object record.

* One such report might be the total number of digital objects transferred to a certain preservation repository in a given year, including in the report the ratio of successful/failed transmissions and the causes for failures. Such a report might be the result of any batch transmission, but I think being able to provide a cumulative representation for such an event for a period of time would also be useful.
* Another report might be a list of all digital objects, indicating if the digital object is replicated, where it was replicated, and when it was last replicated / validated by the preservation management system.
* Yet another report might be a list of all failed events and the recorded reasons for those failures.

The list could be extended further. Whatever reports are implemented should be decided, designed, and implemented by the DAMS management team.

**VI: Implementation:**

The data model and process may be implemented in numerous ways, of which here are three obvious scenarios for the UCSD DAMS environment:

* as additional triples integrated in the triple set for a given object record (i.e., one object record contains all relevant event records and associated agent records. This could dramatically increase the size of the digital object record. This might present challenges for reports on events and agents themselves.)
* as triples outside but linked to the triple set for the a given object record (this has less impact on the digital object record per se, and could better facilitate other management reports for events and agents. Also, this approach takes advantage of the RDF foundation the DAMS has.)
* as SQL tables related to the triples for a given object (Similar to the scenario immediately above but not utilizing RDF for recording event and agent information.)

The second two options will require use of more linking elements to associate agents to events and events to objects.

Implementation Needs:

Regardless of the type of implementation, certain local conventions, and policies for their use, need to be established. They are:

* Identifiers for agents
* Vocabulary for certain fields in the object-event-agent matrix
	+ eventType (? Extend LoC vocabulary)
	+ eventDetail (Format and content of boilerplate)
	+ eventOutcomeInformation
	+ linkingAgentRole
	+ agentType
* Routines for updating object records with links to event records.

**VII: Use Cases**

A: For a DAMS digital object that is transferred to a preservation management system, such as Chronopolis or Merritt, we can imagine the following sequence of steps and metadata updates:

1. A DAMS digital object is packaged and ready for transfer to the Chronopolis service. The package consists of:
	* All content files, masters and derivatives comprising the digital object and referenced in the digital object file inventory
	* XML serializations of the RDF and METS representations of the digital object record

	In this process, the transfer package, typically a tar/gz file, is considered to be ephemeral. Authenticity attributes necessary for monitoring the accuracy of the transfer are recorded, but this information is not maintained, nor are the transfer packages and their contents, beyond the “closure” of the event.
2. The object is transferred to Chronopolis, resulting in
* an event record recording the transfer, known as “ingestion”, indicating the digital object was transmitted to the Chronopolis Preservation Repository and the date it was transferred and indicating the transfer was successful per the confirming report from the Chronopolis service
* related agent records and links thereto, at least one for the staff person authorized to transmit the digital object and one other for the Chronopolis Preservation Repository as recipient of the digital object
1. Two years later, all digital objects, including this one, stored in the Chronopolis Preservation Repository are refreshed, resulting for this digital object in
* the addition of a second “ingestion” event record, dated for the second transfer to Chronopolis
* related agent records for the sender and recipient of the event

At the conclusion of this story, the digital record will have at least two event records indicating the digital object had been transmitted two times to the Chronopolis Preservation Repository for ingestion.

**VIII: RDF encoded example**

The following illustrates how an event record would be encoded in the UCSD DAMS:

<bb4676790q> <pre:event> <bb4676790q-event-2011-07-01T11:06:00>

<bb4676790q-event-2011-07-01T11:06:00> <pre:eventIdentifier> \_:bn2

\_:bn2 <pre:eventIdentifierType> "UCSDDAMS"

\_:bn2 <pre:eventIdentifierValue> "ark-20775-bb4676790q-event-2011-07-01T11:06:00"

<bb4676790q-event-2011-07-01T11:06:00> <pre:eventType> "ingestion"

<bb4676790q-event-2011-07-01T11:06:00> <pre:eventDateTime> "2011-07-01T11:06:00"

<bb4676790q-event-2011-07-01T11:06:00> <pre:eventDetail> "Ingestion, to Chronopolis

 Preservation Repository. Package attributes: Size='NNNNNNNNN';

 CheckSum='??????????'; CheckSumType='SHA-256'; PackageID='?????'."

<bb4676790q-event-2011-07-01T11:06:00> <pre:eventOutcomeInformation> \_:bn3

\_:bn3 <pre:eventOutcome> "Action succeeded"

\_:bn3 <pre:eventOutcomeDetail> \_:bn4

\_:bn4 <pre:eventOutcomeDetailNote> "Chronopolis report: Successful ingest, manifest check

 and replication, received 2011-12-14"

<bb4676790q-event-2011-07-01T11:06:00> <pre:linkingObjectIdentifier> \_:bn5

\_:bn5 <pre:linkingObjectIdentifierType> "UCSDDAMS"

\_:bn5 <pre:linkingObjectIdentifierValue> <[http://libraries.ucsd.edu/ark:/20775/bb4676790q](http://libraries.ucsd.edu/ark%3A/20775/bb4676790q)>

<bb4676790q-event-2011-07-01T11:06:00> <pre:linkingAgentIdentifier> \_:bn6

\_:bn6 <pre:linkingAgentIdentifierType> "UCSDDAMS"

\_:bn6 <pre:linkingAgentIdentifierValue> <[http://libraries.ucsd.edu/ark:/20775/1234-DAMSAGENT](http://libraries.ucsd.edu/ark%3A/20775/1234-DAMSAGENT)>

\_:bn6 <pre:linkingAgentRole> "Transmitter"

<bb4676790q-event-2011-07-01T11:06:00> <pre:linkingAgentIdentifier> \_:bn7

\_:bn7 <pre:linkingAgentIdentifierType> "UCSDDAMS"

\_:bn7 <pre:linkingAgentIdentifierValue> <[http://libraries.ucsd.edu/ark:/20775/1235-DAMSAGENT](http://libraries.ucsd.edu/ark%3A/20775/1235-DAMSAGENT)>

\_:bn7 <pre:linkingAgentRole> "Recipient"