DAITSS and PREMIS Conformance

version 1.0, March 2006

This document shows how the values of PREMIS semantic units could be obtained from the information known to DAITSS.

© Copyright 2006 by the Florida Center for Library Automation.

In the PREMIS data model there are Intellectual Entities and three types of Object entity: representation, file and bitstream. In the DAITSS model there are Intellectual Entities and two types of Object: files and bitstreams. Representations are not currently tracked. This will be changed in 2006, but it has not yet been decided what metadata will be recorded pertaining to representations.

This discrepancy is not non-conforming in principle. PREMIS does not require a repository to track all three types of Object, it only requires that if a particular type is controlled, applicable semantic units should be supported.

At this point in time, there is an assumption that DAITSS Intellectual Entities are one-to-one with representations; that is, that a single SIP contains a single representation of an Intellectual Entity. Therefore the information pertaining to Intellectual Entities in DAITSS can be considered to pertain to the PREMIS representation, and some elements are recorded as applicable to representations in the chart below. However this is not entirely accurate because the assumption breaks down in two places: 1) we can't control what people send, and have no guarantee a SIP contains a single representation of a single Intellectual Entity, and 2) archive-created migrations and normalizations create representations that are not tracked.

The table below shows how the values of PREMIS elements could be supplied from information currently known to DAITSS.

Under Object Type, R=Representation, F=File, B=Bitstream.

Where element values can be taken from a value recorded in the DAITSS management database, the value is identied as [table_name].[column_name]. E.g. "admin.oid" means the column "oid" of the "admin" table.

PREMIS semantic unit	**Object Type**	**Mapping from DAITSS**	**Notes**
objectIdentifier			
. objectIdentifierType	R, F, B	constant "FDA"	all identifiers are local
. objectIdentifierValue	R	admin.oid	assigned by Ingest
. objectIdentifierValue	F	data_file.dfid	assigned by Ingest
. objectIdentifierValue	B	bsid within the dfid	assigned by Ingest
preservationLevel	R	none	not implemented at representation level
preservationLevel	F	data_file.pres_level	assigned by Ingest based on parameters supplied by

			depositor when account is set up
objectCategory	R, F, B	can be supplied	not recorded, but can be inferred from tables and/or object instances
objectCharacteristics			
. compositionLevel	F, B	none	not supported in DAITSS; might be able to supply in some cases
. fixity			
. . messageDigestAlgorithm	F	message_digest.code	assigned by Ingest
. . messageDigestAlgorithm	B		not supported
. . messageDigest	F	message_digest.value	created by Ingest
. . messageDigest	B		not supported
. . messageDigestOriginator	F	message_digest.origin	assigned by Ingest
. . messageDigestOriginator	B		not supported
. size	F	data_file.size	determined by Ingest
. size	B		not supported
. format			
. . formatDesignation			
. . . formatName	F	format.media_type	determined by Ingest by parsing file in JHOVE-like way; data_file.format gives a code that maps to format table
. . . formatName	B	can be provided	determined in Ingest as above; the bitstream object knows what it is; some metadata also in bitstream-specific tables
. . . formatVersion	F	format.media_type_version	see note for formatName
. . . formatVersion	B	can be provided	see note for formatName
. . formatRegistry			
. . . formatRegistryName	F, B		registries not supported yet
. . . formatRegistryKey	F, B		

. . . formatRegistryRole	F, B		
. significantProperties	R,F,B		not supported
. inhibitors			
. . inhibitorType	F	data_file_severe_element.severe_element	contains a severe_element code assigned by Ingest; some codes are for inhibitors
. . inhibitorType	B		not supported
. . inhibitorTarget	F	constant "all content"	not supported in DAITSS so would have to assume applies to all
. . inhibitorTarget	B		not supported
. . inhibitorKey	F , B		not supported
. creatingApplication			
. . creatingApplicationName	F	data_file.creator_prog	determined by Ingest if contained in file header; name would have to be parsed out, as our value includes both name and version
. . creatingApplicationName	R, B		not supported
. . creatingApplicationVersion	F	data_file.creator_prog	see creatingApplicationName
. . creatingApplicationVersion	R, B		not supported
. . dateCreatedByApplication	F	data_file.create_date	mapping will usually but not always be correct; taken by Ingest from the file header when possible (but when not present, the Ingest date will be assigned); assigned by Ingest when file is created in the archive
. . dateCreatedByApplication	R, B		not supported
. originalName	F	data_file.file_title	determined by Ingest
. storage			
. . contentLocation			

. . . contentLocationType	F	would be provided as a constant	can assume location type for files
. . . contentLocationType	B	bitstream.location_type	assigned by Ingest
. . . contentLocationValue	F	storage_desc.identifier	assigned by Ingest
. . . contentLocationValue	B	bitstream.location_value	assigned by Ingest
. . storageMedium	F, B	can be inferred from storage_instance.method	currently this would be "TSM" from which we'd infer the tape unit
. environment			
. . environmentCharacteristics	R, F, B		not supported
. . environmentPurpose	R, F, B		not supported
. . environmentNote	R, F, B		not supported
. . dependency			
. . . dependencyName	R, F, B		not supported
. . . dependencyIdentifier			
. . . . dependencyIdentifierType	R, F, B		not supported
. . . . dependencyIdentifierValue	R, F, B		not supported
. . software			
. . . swName	R, F, B		not supported
. . . swVersion	R, F, B		not supported
. . . swType	R, F, B		not supported
. . . swOtherInformation	R, F, B		not supported
. . . swDependency	R, F, B		not supported
. . hardware			
. . . hwName	R, F, B		not supported
. . . hwType	R, F, B		not supported
. . . hwOtherInformation	R, F, B		not supported
. signatureInformation			
. . signatureInformationEncoding	F, B		not supported
. . signer	F, B		not supported
. . signatureMethod	F, B		not supported
. . signatureValue	F, B		not supported
. . signatureValidationRules	F, B		not supported
. . signatureProperties	F, B		not supported
. . keyInformation			
. . . keyType	F, B		not supported
. . . keyValue	F, B		not supported
. . . keyVerificationInformation	F, B		not supported
. relationship			only relationships between files are recorded in relationship table,

			others must be inferred
. . relationshipType	R	constant "structural"	only "hasPart" supported, for relation of Representation to File; assigned by Ingest
. . relationshipType	F	can be inferred from sub type	
. . relationshipType	B	constant "structural"	only "isPartOf" supported, for relation of Bitstream to File; determined by Ingest
. . relationshipSubType	R		not supported
. . relationshipSubType	F	relationship.rel_type	
. . relationshipSubType	B	supply constant meaning "isPartOf"	see relationshipType
. . relationshipObjectIdentification			
. . . relatedObjectIdentifierType	R, F, B	constant "FDA"	
. . . relatedObjectIdentifierValue	R	identifier of the data file related to the Intellectual Entity ID	
. . .relatedObjectIdentifierValue	F	relationship.dfid_1 or relationship.dfid_2	
. . .relatedObjectIdentifierValue	B	the bitstream identifier for the bitstream with bitstream.dfid	
. . . relatedObjectSequence	R, F		not supported
. . . relatedObjectSequence	B	bitstream.sequence	
. . relatedEventIdentification			
. . . relatedEventIdentifierType	R, F, B	constant "FDA"	
. . . relatedEventIdentifierValue	R	relationship.event_id	
. . . relatedEventIdentifierValue	F	relationship.event_id	
. . . relatedEventIdentifierValue	B		not supported
. . . relatedEventSequence	R, F	constant "1"	all relationships are the result of a single event
. linkingEventIdentifier			
. . linkingEventIdentifierType	R, F	constant "FDA"	
. . linkingEventIdentifierType	B		not supported
. . linkingEventIdentifierValue	R, F	event.id	get to proper event table entry from event.oid, the id of

			the related file or Int Entity
. . linkingEventIdentifierValue	B		not supported
. linkingIntellectualEntityIndentifier			
. . linkingIntellectualEntityIdentifierType	R, F, B	constant "FDA" or constant "FDAuser"	
. . linkingIntellectualEntityIdentifierValue	R, F, B	int_entity.ieid or int_entity.entity_id	ieid assigned by Ingest, entity_id submitted in metadata by depositor
. linkingPermissionStatementIdentifier			
. . linkingPermissionStatementIdentifierType			not supported
. . linkingPermissionStatementIdentifierValue			not supported

EVENTS	**mapping from DAITSS**	**notes**
		Only events performed by DAITSS are recorded
eventIdentifier		
. eventIdentifierType	constant "FDA"	
. eventIdentifierValue	event.id	
eventType	event.event_type	
eventDateTime	event.date_time	
eventDetail	event.event_procedure	
eventOutcomeInformation		
. eventOutcome	event.outcome	
. eventOutcomeDetail	event.note	
linkingAgentIdentifier		DAITSS does not link agents to events, but archive software creates all events, so the archive could be identified as the

		agent by default
. linkingAgentIdentifierType		see above
. linkingAgentIdentifierValue		see above
. linkingAgentRole		see above
linkingObjectIdentifier		
. linkingObjectIdentifierType	constant "FDA"	
. linkingObjectIdentifierValue	event.oid, event.rel_oid	Events can have one or two related objects; if two the second is in rel_oid
AGENTS		The only individuals defined in DAITSS are contact persons for problems, billing, etc. Data for contacts is entered manually into the system. Contacts could be mapped to PREMIS agents.
agentIdentifier		
. agentIdentifierType	constant "FDA"	
. agentIdentifierValue	contact.id	
agentName	contact.name	
agentType	can be inferred from ?	
RIGHTS		Rights are not supported in DAITSS
permissionStatement		
. permissionStatementIdentifier		
. . permissionStatementIdentifierType		not supported
. . permissionStatementIdentifierValue		not supported
. linkingObject		not supported
. grantingAgent		not supported
. grantingAgreement		
. . grantingAgreementIdentification		not supported
. . grantingAgreementInformation		not supported
. permissionGranted		
. . act		not supported
. . restriction		not supported
. . termOfGrant		
. . . startDate		not supported

. . . endDate		not supported
. . permissionNote		not supported