OSID V3 Specifications
id package

Version Draft 3

This specifications represent a draft for OSID V3 interface definitions. These definitions may change at any time.

Last Modified: 4 July 2008

prepared by:
Tom Coppeto
OnTapSolutions

	OSID License
Copyright	Copyright © 2002-2008 Massachusetts Institute of Technology. All Rights Reserved.
License	This Work is being provided by the copyright holder(s) subject to the following license. By obtaining, using and/or copying this Work, you agree that you have read, understand, and will comply with the following terms and conditions.

This Work and the information contained herein is provided on an "AS IS" basis. The Massachusetts Institute of Technology, the Open Knowledge Initiative, and THE AUTHORS DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE WORK OR THE USE OR OTHER DEALINGS IN THE WORK.

Permission to use, copy and distribute unmodified versions of this Work, for any purpose, without fee or royalty is hereby granted, provided that you include the above copyright notice and the terms of this license on ALL copies of the Work or portions thereof.

You may nodify or create Derivatives of this Work only for your internal purposes. You shall not distribute or transfer any such Derivative of this Work to any location or to any third party. For the purposes of this license, Derivative shall mean any derivative of the Work as defined in the United States Copyright Act of 1976, such as a translation or modification.

The export of software employing encryption technology may require a specific license from the United States Government. It is the responsibility of any person or organization comtemplating export to obtain such a license before exporting this Work. |

Package Description	osid.id package

Interfaces	osid.id.IdProfile
	osid.id.IdManager
	osid.id.IdProxyManager
	osid.id.IdLookupSession
	osid.id.IdAdminSession
	osid.id.Id
	osid.id.IdList

Package	osid.id
Title	**Id Open Service Interface Definitions**
Version	3.0.0

Description	The Id OSID provides the means for creating and mapping identifiers. All OSID objects are identified by a unique and immutable Id. The Id OSID can be used to generate new Ids when creating new objects.
	
	Consumers wishing to persist an OSID object should instead persist the reference to the object by serializing the Id.
	
	Most OSID interfaces are used to encapsulate implementation-specific objects from provider to consumer. The Id interface is bi-directional and as such cannot be used to encapsulate implementation-specific data other than what is defined explicitly in the Id. A provider must respect any Id based on its interface alone.
	
	The Id service can also be used as a means to map one identifier to another when an object is known by multiple identifiers. Mapping identifier spaces is often a critical part of interoperability and the Id service can be used as a shim to bridge different systems.
	
	Id mapping example:
	
	`public Asset getAsset(assetId) {`
	`    Id id = idSession.getId(assetId);`
	`    return (other_impl.getAsset(assetId));`
	`}`

Interface	osid.id.IdProfile	
Implements	osid.OsidProfile	
Description	The IdProfile describes the interoperability among id services.	
Method	**supportsIdLookup**	
Description	Tests if Id lookup is supported.	
Return	boolean	true if Id lookup is supported, false otherwise
Compliance	mandatory	This method must be implemented.
Method	**supportsIdAdmin**	
Description	Tests if an Id administrative service is supported.	
Return	boolean	true if Id administration is supported, false otherwise
Compliance	mandatory	This method must be implemented.

Interface	osid.id.IdManager	
Implements	osid.OsidManager	
	osid.id.IdProfile	
Description	This manager provides access to the available sessions of the Id service. Ids are created through the IdAdminSession which provides the means for creating a unique identifier. The IdLookupSession can be used for mapping one Id to another in addition to getting a list of the assigned identifiers.	
Method	**getIdLookupSession**	
Description	Gets the session associated with the id lookup service.	
Return	osid.id.IdLookupSession	an IdLookupSession
Errors	OPERATION_FAILED	unable to complete request
	UNIMPLEMENTED	supportsIdLookup() is false
Compliance	optional	This method must be implemented if supportsIdLookup() is true.
Method	**getIdAdminSession**	
Description	Gets the session associated with the id admin service.	
Return	osid.id.IdAdminSession	the new IdAdminSession
Errors	OPERATION_FAILED	unable to complete request
	UNIMPLEMENTED	supportsIdAdmin() is false
Compliance	optional	This method must be implemented if supportsIdAdmin() is true.

Interface	osid.id.IdProxyManager	
Implements	osid.OsidProxyManager	
	osid.id.IdProfile	
Description	This manager provides access to the available sessions of the Id service. Methods in this manager support the passing of an Authentication object for the purpose of proxy authentication. Ids are created through the IdAdminSession which provides the means for creating a unique identifier. The IdBrowserSession can be used for mapping one Id to another in addition to getting a list of the assigned identifiers.	

Method	getIdLookupSession	
Description	Gets the session associated with the id lookup service using the supplied Authentication.	
Parameters	osid.authentication.Authentication authentication	proxy authentication
Return	osid.id.IdLookupSession	an IdLookupSession
Errors	NULL_ARGUMENT	authentication is null
	OPERATION_FAILED	unable to complete request
	PERMISSION_DENIED	authentication is invalid
	UNIMPLEMENTED	supportsIdLookup() is false
	UNSUPPORTED	the authentication service is not supported
Compliance	optional	This method must be implemented if supportsIdLookup() is true.

Method	getIdAdminSession	
Description	Gets the session associated with the id administrative service.	
Parameters	osid.authentication.Authentication authentication	proxy authentication
Return	osid.id.IdAdminSession	a IdAdminSession
Errors	NULL_ARGUMENT	authentication is null
	OPERATION_FAILED	unable to complete request
	PERMISSION_DENIED	authentication is invalid
	UNIMPLEMENTED	supportsIdAdmin() is false
	UNSUPPORTED	the authentication service is not supported
Compliance	optional	This method must be implemented if supportsIdAdmin() is true.

Interface	osid.id.IdLookupSession		
Implements	osid.OsidSession		
Description	This session is for retrieving Id objects. getIds() retrieves all known Ids. The existence of a single identifier can be confirmed through the getId() method, or it can be used as a means of Id translation.		
Method	**canLookupIds**		
Description	Tests if this user can perform Id lookups. A return of true does not guarantee successful authorization. A return of false indicates that it is known all methods in this session will result in a PERMISSION_DENIED. This is intended as a hint to an application that may opt not to offer lookup operations.		
Return	boolean		false if lookup methods are not authorized, true otherwise
Compliance	mandatory		This method must be implemented.
Method	**getId**		
Description	Gets an Id.		
Parameters	osid.id.Id	id	an Id
Return	osid.id.Id		the Id
Errors	NOT_FOUND		id is not found
	NULL_ARGUMENT		id is null
	OPERATION_FAILED		unable to complete request
	PERMISSION_DENIED		authorization failure
Compliance	mandatory		This method must be implemented.
Method	**getIds**		
Description	Gets all Ids.		
Return	osid.id.IdList		the list of all Ids
Errors	OPERATION_FAILED		unable to complete request
	PERMISSION_DENIED		authorization failure
Compliance	mandatory		This method must be implemented.

Interface	osid.id.IdAdminSession	
Implements	osid.OsidSession	
Description	This session is used to create a new Id.	
Method	**canAdministerIds**	
Description	Tests if this user can perform administer ids. A return of true does not guarantee successful authorization. A return of false indicates that it is known all methods in this session will result in a PERMISSION_DENIED. This is intended as a hint to an application that may opt not to offer lookup operations.	
Return	boolean	false if administrative methods are not authorized, true otherwise
Compliance	mandatory	This method must be implemented.
Method	**createId**	
Description	Creates a new Id.	
Return	osid.id.Id	the created Id
Errors	OPERATION_FAILED	unable to complete request
	PERMISSION_DENIED	authorization failure
Compliance	mandatory	This method must be implemented.
Method	**addId**	
Description	Adds a new Id.	
Parameters	osid.id.Id	id	a new Id
Errors	NULL_ARGUMENT	id is null
	OPERATION_FAILED	unable to complete request
	PERMISSION_DENIED	authorization failure
Compliance	mandatory	This method must be implemented.
Method	**makeEquivalent**	
Description	Makes two Ids equivalent. Calls to IdLookupSession.getId(id) return the realId.	
Parameters	osid.id.Id	primaryId	the primary Id
	osid.id.Id	equivalentId	an Id to be made equivalent
Errors	NOT_FOUND	realId or id is not found
	NULL_ARGUMENT	realId or id is null
	OPERATION_FAILED	unable to complete request
	PERMISSION_DENIED	authorization failure
Compliance	mandatory	This method must be implemented.

Interface	osid.id.Id
Implements	
Description	Id represents an identifier object. Ids are designated by the following elements: • identifier: a unique key or guid • namespace: the *namespace* of the identifier • authority: the issuer of the identifier Two Ids are equal if their namespace, identifier and authority strings are equal. Only the identifier is case-sensitive. Persisting an Id means persisting the above components
Method	**getAuthority**
Description	Gets the authority of this Id. The authority is a string used to ensure the uniqueness of this Id when using a non-federated identifier space. Generally, it is a domain name identifying the party responsible for this Id. This method is used to compare one Id to anoher.
Return	string	the authority of this Id
Compliance	mandatory	This method must be implemented.
Method	**getIdentifierNamespace**
Description	Gets the namespace of the identifier. This method is used to compare one Id to another.
Return	string	the authority of this Id
Compliance	mandatory	This method must be implemented.
Method	**getIdentifier**
Description	Gets the identifier of this Id. This method is used to compare one Id to another.
Return	string	the identifier of this Id
Compliance	mandatory	This method must be implemented.
Method	**isEqual**
Description	Determines if the given Id is equal to this one. Two Ids are equal if the namespace, authority and identifier components are equal. The identifier is case sensitive while the namespace and authority strings are not case sensitive.
Parameters	osid.id.Id	id	the Id to compare
Return	boolean	true if the given Id is equal to this one, false otherwise
Compliance	mandatory	This method must be implemented.

Interface	osid.id.IdList
Implements	osid.OsidList

Description	Like all OsidLists, IdList provides a means for accessing Id elements sequentially either one at a time or many at a time. Examples:
	
	`while (il.hasNext()) {`
	`    Id id = il.getNextId();`
	`}`
	
	or
	
	`while (il.hasNext()) {`
	`    Id[] ids = il.getNextIds(il.available());`
	`}`

Method	getNextId	
Description	Gets the next Id in this list.	
Return	osid.id.Id	the next Id in this list. The hasNext() method should be used to test that a next Id is available before calling this method.
Errors	ILLEGAL_STATE	no more elements available in this list
	OPERATION_FAILED	unable to complete request
Compliance	mandatory	This method must be implemented.

Method	getNextIds		
Description	Gets the next set of Ids in this list. The specified amount must be less than or equal to the return from available().		
Parameters	cardinal	n	the number of Id elements requested which must be less than or equal to available()
Return	osid.id.Id[]		an array of Id elements. The length of the array is less than or equal to the number specified.
Errors	ILLEGAL_STATE		no more elements available in this list
	OPERATION_FAILED		unable to complete request
Compliance	mandatory		This method must be implemented.