OSID Specification Request: 7
Track: Interface
Status: Accepted

Tom Coppeto
Ingenescus

April 2014

Reliable Notifications

Status

This document is a request for a specification change for review.

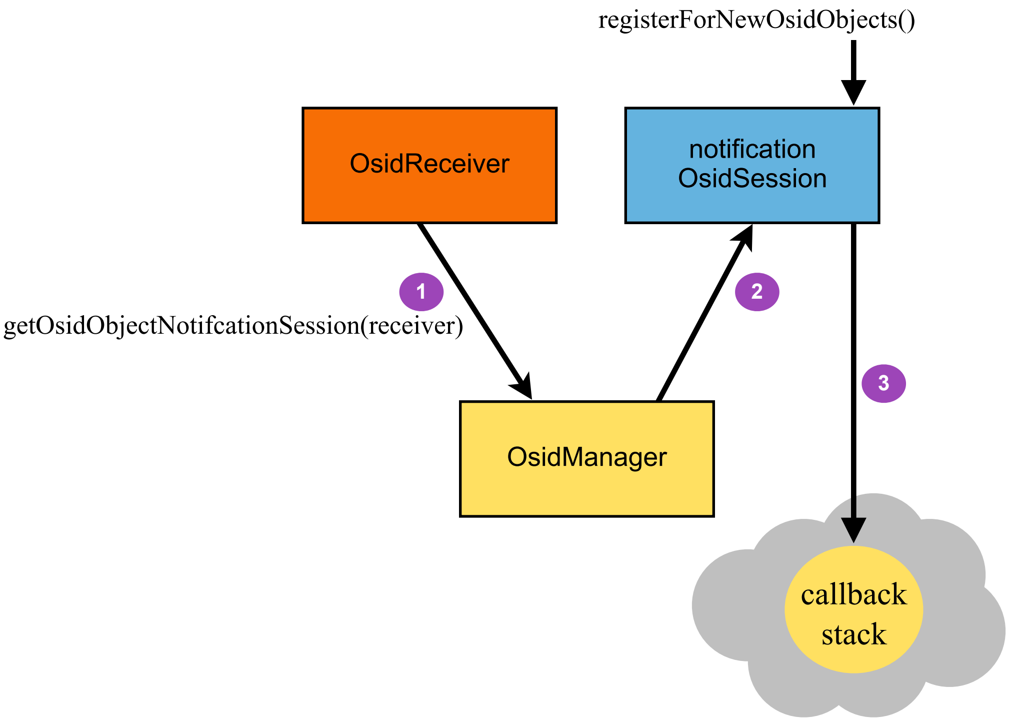
Summary

OSID Providers can notify OSID Consumers of events pertaining to the status of OsidObjects through OsidReceiver callback interfaces. This document proposes OsidReceiver and notification OsidSession changes to implement reliable notifications.

Table of Contents

1. Current Specification

OSID Providers can notify OSID Consumers of events pertaining to the status of OsidObjects through OsidReceivers. An OsidReceiver is a consumer-owned object that handles the events. OsidReceivers are supplied when launching a notification OsidSession.


registerForNewOsidObjects()
OsidReceiver
notification OsidSession
1
2
3
getOsidObjectNotifcationSession(receiver)
OsidManager
callback stack

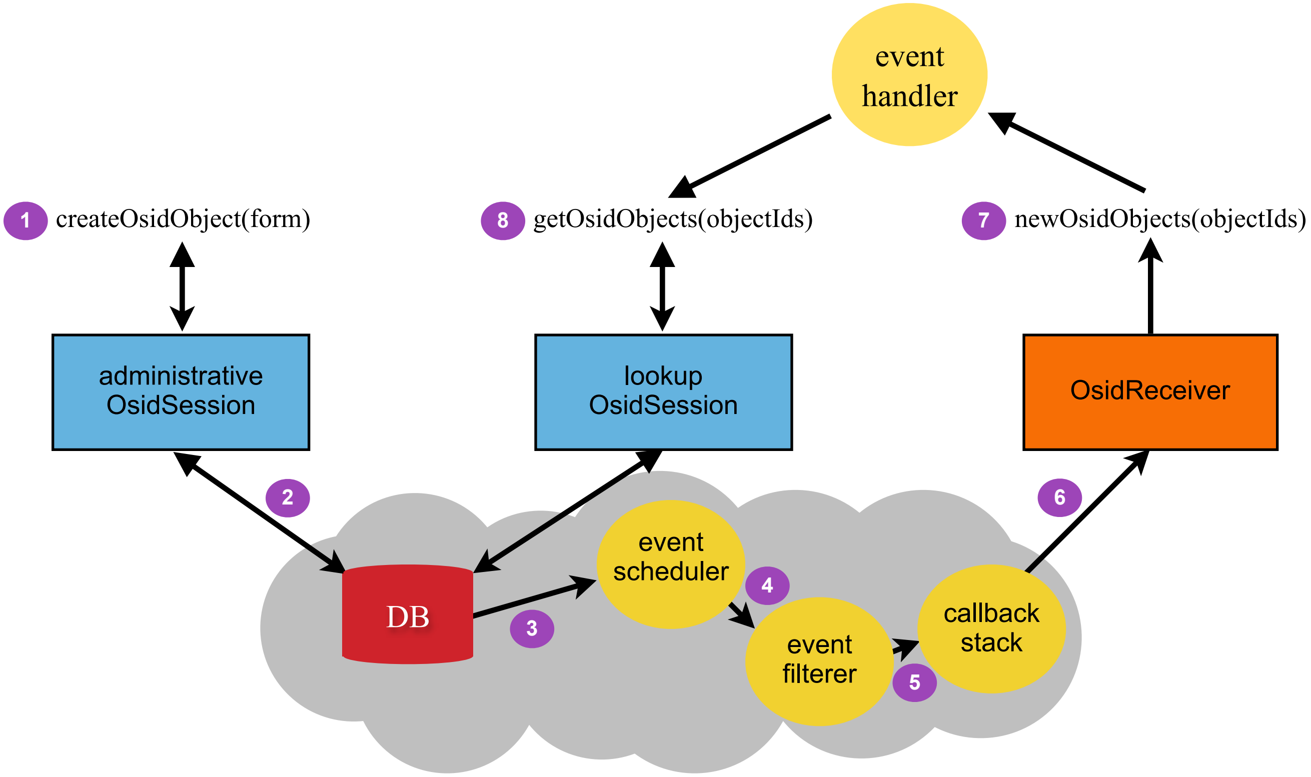
OSID Providers track notification registrations. When an event is triggered, such as the creation of an OsidObject, OSID Consumers who have registered for the event receive a callback via their OsidReceivers.

No information is supplied in the callback other than the Ids of the OsidObjects. OSID Consumers have to retrieve the OsidObjects for the Ids supplied in the callback.

Notification sessions and the accompanying OsidReceivers break down OsidObject events along typical create/update/delete boundaries. There are also events for changes in hierarchies.

High level filtering can be performed along the major relations of an OsidObject. Detailed filtering can be performed through the selection of specific OsidCatalogs. The OSIDs do not suggest where this filtering takes place nor what the nature of the message bus is (i.e. multicast or unicast). The callback stack can be maintained anywhere along the path.

The OsidReceiver contains no data other than the identity of the entity to which the event pertains. These notifications are sent outside the context and security perimeter of the event trigger. The recipient may have a different authorization, data presentation, or an entirely separate service endpoint. OsidReceivers are expected to retrieve the OsidObject in response to create and update notifications.



2. Problem

There is no way for an OSID Provider to know that a notification event has been successfully processed.

3. Proposed Changes

Receiver callback methods include an identifier along with the list of OsidObject Ids that can be used to acknowledge persistence or processing of the event. This identifier is an identifier to the entity representing the notification transaction.

The notification OsidSessions provide a means for acknowledging received notifications.

4. Impacts

4.1. Specification

This is a widespread change across all OsidReceivers. A consideration is the limited existence of notification implementations that may make an interface change at this stage of release candidate acceptable. Interface adapters can be inserted to mitigate the extra parameter.

4.2. OSID Consumers

Interface adapters can be inserted to mitigate the extra parameter.

4.3. OSID Providers

OSID Providers which do not offer reliable communication can supply a unique Id for the notification.

5. Interoperability Considerations

OSID Providers may or may offer reliable notifications. OSID Consumers may or may not care about reliable notifications.

OSID Consumer	OSID Provider	
unreliable	unreliable	OSID Provider supplies a random unique Id to OsidReceiver Callbacks. OSID Consumer ignores it an sets NotificationSession.unreliable() to state that it won't acknowledge anything.
unreliable	reliable	OSID Provider supplies an Id to a notification it is storing in its notification queue. OSID Consumer ignores it an sets NotificationSession.unreliable() to state that it won't acknowledge anything. OSID Provider assumes acknowledgement upon invoking the callback.
reliable	reliable	OSID Provider supplies an Id to a notification it is storing in its notification queue. OSID Consumer sets NotificationSession.reliable() to state that it will acknowledge each notification. OSID Provider assumes acknowledgement upon invoking the callback and does so.

OSID Consumer	OSID Provider	
reliable	unreliable	OSID Provider supplies a random unique Id to OsidReceiver Callbacks. OSID Consumer sets NotificationSession.reliable() to state that it will acknowledge each notification. OSID Provider ignores acknowledgements. OSID Consumer observes unreliable behavior from this OSID Provider and shops for a new OSID Provider.

6. Proposed Interfaces

6.1. Example OsidReceiver

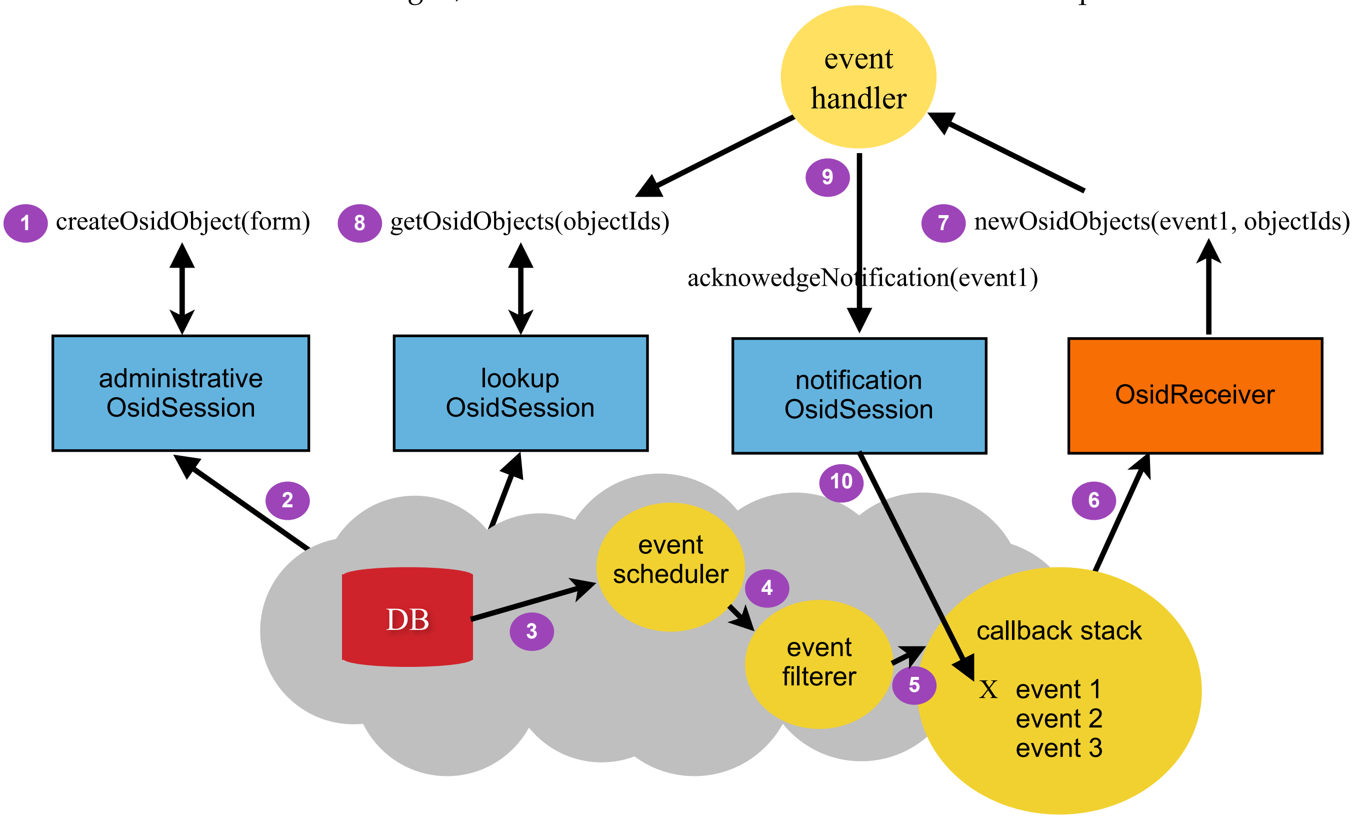
Interface	**osid.assessment.BankReceiver**		
Implements	osid.OsidReceiver		
Description	The bank receiver is the consumer supplied interface for receiving notifications pertaining to new, updated, or deleted Banks.		
Method	**newBanks**		
Description	The callback for notifications of new banks.		
Parameters	osid.id.Id	notificatiionId	the notification Id
	osid.id.IdList	bankIds	a list of bank Ids
Compliance	mandatory		This method must be implemented.
Method	**changedBanks**		
Description	The callback for notifications of changed banks.		
Parameters	osid.id.Id	notificatiionId	the notification Id
	osid.id.IdList	bankIds	a list of bank Ids
Compliance	mandatory		This method must be implemented.
Method	**deletedBanks**		
Description	The callback for notifications of deleted banks.		
Parameters	osid.id.Id	notificatiionId	the notification Id
	osid.id.IdList	bankIds	a list of bank Ids
Compliance	mandatory		This method must be implemented.
Method	**changedChildOfBanks**		
Description	The callback for notifications of changes to parents and children of bank hierarchy nodes.		
Parameters	osid.id.Id	notificatiionId	the notification Id
	osid.id.IdList	bankIds	a list of bank Ids
Compliance	mandatory		This method must be implemented.

6.2. Example Notification OsidSession

Interface	**osid.course.CourseNotificationSession**	
Implements		
Description	A notification.	
Method	**reliable**	
Description	Reliable notifications are desired. In reliable mode, notifications will be acknowledged using acknowledgeNotification().	
Compliance	mandatory	This method must be implemented.
Method	**unreliable**	
Description	Unreliable notifications are desired. In unreliable mode, notifications will not be acknowledged.	
Compliance	mandatory	This method must be implemented.
Method	**acknowledgeNotification**	
Description	Acknowledge a notification.	
Parameters	osid.id.Id	the notification Id
Errors	OPERATION_FAILED	unable to complete request
	PERMISSION_DENIED	an authorization failure occurred
Compliance	mandatory	This method must be implemented.

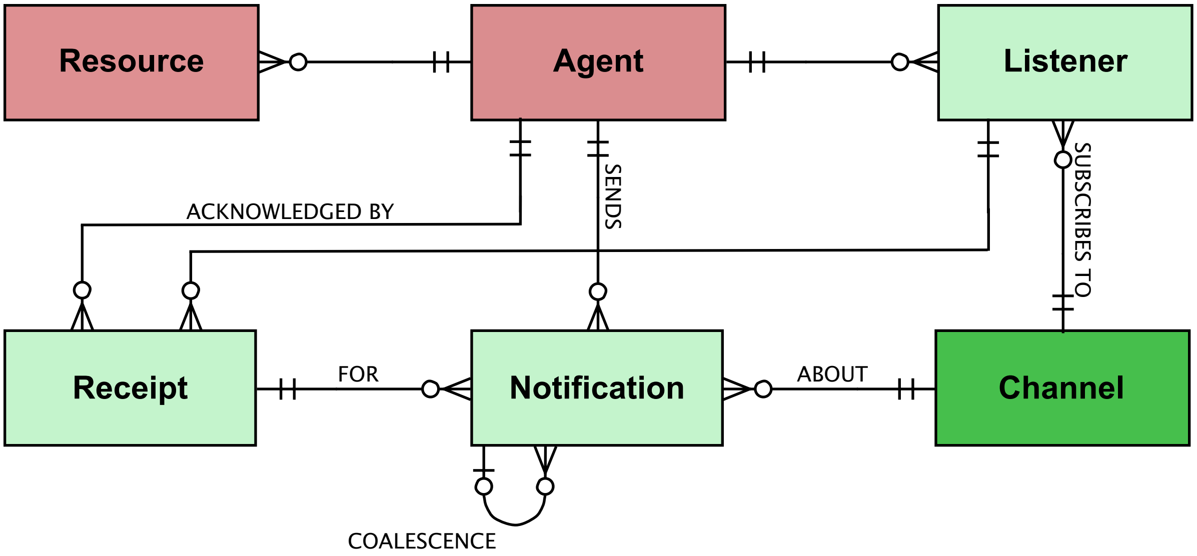
7. Scenario

An OSID Provider may provide reliable notifications by storing the callbacks. When the notification event is acknowledged, the OSID Provider removes the event from its queue.



8. Notification Ids

Ids identify OsidObjects. The notification Id should be grounded in one. This is intended to be a placeholder for a new OSID that manages both the sending and receipt of notifications. The inline notification pattern can be considered a cover for a more functional notification service.



9. Statement