OSID Specification Request: 9
Track: Interface
Status: Proposed

Tom Coppeto
Ingenescus

July 2014

Hierarchies

Status

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

Summary

Hierarchies are typically used for federating OsidCatalogs. There are several other OsidObjects which are also hierarchical. These OsidObjects also appear in OsidCatalogs and this creates an intersection among two hierarchies. While this document does not propose any change or clarification to implementation considerations, the references in any OSID defining such federateable OsidObjects can be clarified.

Table of Contents

1. Current Specification

1.1. OsidCatalogs

OSIDs include hierarchy OsidSessions for each OsidCatalog. These hierarchical OsidSessions indicate the osid.hierarchy.Hierarchy to which the operations pertain. This Hierarchy appears to serve as a means of orchestrating with other OSIDs. OSID Consumers may not choose a particular the hierarchy or operate on it in any other way. There is a single hierarchical structure of OsidCatalogs per OSID.

1.2. Non-OsidCatalog Federateables

OSIDs include hierarchy OsidSessions for each Federateable. These include:

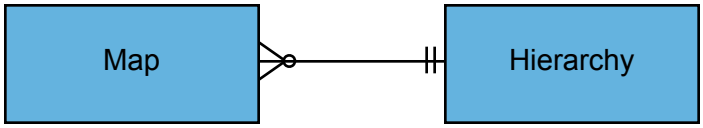
- osid.authorization.Qualifier
- osid.checklist.Todo
- osid.course.Term
- osid.financials.Account
- osid.financials.Activity
- osid.inventory.Stock
- osid.learning.Objective
- osid.mapping.Location
- osid.ontology.Subject
- osid.personnel.Organization

Management of these hierarchies occur within the context of two OsidCatalogs. The nodes are in the context of the OsidCatalog defined each of these OSIDs. The hierarchical structure is defined in the context of an osid.hierarchy.Hierarchy.

```
MappingManager {
    …
    osid.mapping.LocationHierarchySession
            getLocationHierarchySessionForMap(osid.id.Id mapId);
    …
}

LocationHierarchySession {
    …
    osid.hierarchy.Hierarchy getLocationHierarchy();
    …
}
```

This implies the following model:



The orchestration between the Map and the Hierarchy occur behind the scenes. These hierarchical OsidSessions presumably operate on Maps known to the Hierarchy which, via the relation, are known to the Map.

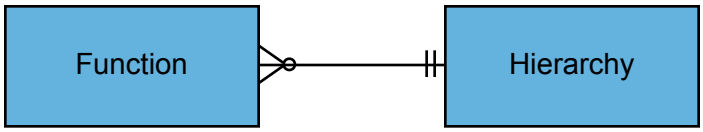
In basic cases, the Map and the Hierarchy can be one in the same (both are OsidCatalogs). In complex cases, the Hierarchy may not be aligned with the Map. For example, Locations outside of the given Map are part of the Hierarchy. The OSIDs provide no guidance here other than to suggest that it could happen.

There are several interpretations:

- There is a single global Location Hierarchy known to the entire OSID Provider. The Hierarchy includes Locations across all Maps. However, this can create an asymmetry as the Map that was used to instantiate the hierarchy OsidSession would show Locations that are not visible or manageable in other OsidSessions instantiated with the same Map.
- There is a single Location Hierarchy known to the entire OSID Provider. The hierarchical OsidSessions show only those Locations visible through the Map that was used for instantiation. However, the hierarchical OsidSessions show the same Hierarchy although its structure would appear to change as Locations appear and disappear by moving up and down the Map Hierarchy.
- Multiple Location Hierarchies exist (presumably managed in the Hierarchy OSID) and are mapped to different Maps. In this case, federation of the Map has no effect on the Hierarchy other than to constrain what Locations can appear in the Hierarchy.

1.3. Authorization Qualifiers

Qualifiers in the Authorization OSID differ from the other non-OsidCatalog Federateables. Here, the Qualifier Hierarchy is directly associated with a Function.



It is assumed that the selection of the Hierarchy is based on an orchestrated Hierarchy OSID bypassing any filtering or grouping attained by the Vault OsidCatalog. In other words, all Qualifier Hierarchies are global to all OsidCatalogs in the Authorization OSID.

2. Problem

The jump from the OsidCatalog to the Hierarchy is confusing.

3. Proposed Changes

3.1. Non-OsidCatalog Federateables

Indicate the primary OsidCatalog in the hierarchical OsidSessions, not the Hierarchy. This would follow the same pattern as its sibling OsidSessions and remove the confusion resulting from the magic leap from the primary OsidCatalog to a Hierarchy.

Moving between the primary OSID and the auxiliary Hierarchy OSID can still be performed through an orchestration agreement between the primary OsidCatalog and the Hierarchy, as would be done with other OSIDs. Reaching over to the Hierarchy OSID should be rare as most of the functionality is present in the primary OSID.

From the primary OSID point of view, there is a single hierarchy available per OsidCatalog. Any orchestration with other OSIDs can be performed using the OsidCatalog.

3.2. Authorization Qualifiers

The hierarchical Qualifier OsidSessions are a special case since these can operate on more than one Hierarchy. The QualifierHierarchySession and QualifierHierarchyDesignSession methods can support multiple hierarchies by including a Qualifier Hierarchy Id as a parameter.

Surface the Qualifier Hierarchy as a managed entity in the Authorization OSID. Although much of this functionality is redundant with the Hierarchy OSID, this allows for a cleaner alignment between the Qualifier hierarchies and the Vault OsidCatalog.

4. Impacts

4.1. Specification

4.1.1. Non-OsidCatalog Federateables

These changes have minor impact on the 9 sets of hierarchy traversal and management OsidSessions.

4.1.2. Authorization Qualifier Hierarchies

Including Qualifier Hierarchy retrieval and management services involves the addition of another 20 interfaces and 13 new operations in each of the OsidManagers in the Authorization OSID.

4.2. OSID Consumers

4.2.1. Non-OsidCatalog Federateables

OSID Consumers will see a change of return interface from the OsidCatalog identification methods hierarchical OsidSessions. Any Auxiliary OSID used in conjunction with the hierarchy can be done using the OsidCatalog instead. The Hierarchy will not be visible in the primary OSID.

4.2.2. Authorization Qualifier Hierarchies

OSID Consumers of qualifier hierarchical OsidSessions will see the impact of having to specify the qualifier hierarchy Id for these operations. The Qualifier Hierarchies will have both visibility and management in the Authorization OSID.

4.3. OSID Providers

4.3.1. Non-OsidCatalog Federateables

OSID Providers will no longer need to conjure a Hierarchy from an OsidCatalog to support these interfaces. Any existing orchestration supported between the primary OSID and the Hierarchy OSID will need to be reworked.

4.3.2. Authorization Qualifier Hierarchies

OSID Providers will feel the impact of on the QualifierHierarchySession and QualifierHierarchyDesignSession by having to process the Qualifier Hierarchy Id on every operation in these OsidSessions.

If the OSID Provider is currently leveraging the Hierarchy OSID to support these implementations, this will involve a complete rework of this layering as multiple OsidSessions in the Hierarchy OSID will need to be federated to support these new operations.

5. Interoperability Considerations

The orchestration agreement between the OsidCatalog and the Hierarchy moves but remains for those rare OSID Consumers who need to drill into the Hierarchy OSID.

Clarifying a Qualifier Hierarchy removes an orchestration agreement between the Authorization and Hierarchy OSIDs that was necessary to manage Functions.

6. Statement