

### Summary of operations for PersonManager

Operation	Description	REST	Comments on REST
createPerson	To request the creation of a populated 'person' record on the target system where the source is responsible for the allocation of the unique identifier.	PUT http://server/person/{sourcedId} payload = Person XML  response = -	
createByProxyPerson	To request the creation of a populated 'person' record on the target system where the target is responsible for the allocation of the unique identifier.	PUT http://server/person payload = Person XML  response = sourcedId	Not really in keeping?
deletePerson	To request the deletion of a 'person' record. The 'person' record is deleted and all of its associated relationships.	DELETE  http://server/person/{sourcedId} payload = - response = -	
readPerson	To read the full contents of the identified 'person' record. The target must return all of the data it has for the identified 'person' record.	GET  http://server/person/{sourcedId} payload = - response = Person XML	
updatePerson	To write new content into the identified 'person' record. The target must write	POST	

	the new data into the 'person' record. This is an additive operation.	http://server/person/{sourcedId} payload = Person XML response = -	
replacePerson	To replace the content of the identified 'person' record. The target must write the new data into the 'person' record. This is a destructive write-over of all of the original information.	PUT  http://server/person/{sourcedId} payload = Person XML response = -	
changePersonIdentifier	To change the sourcedId of the 'person' record. The completion of this operation will result in later actions using the original sourcedId reporting an unknown identifier status.	POST  http://server/person/{sourcedId} payload = sourcedId response = -	This relies on “server” inspecting the payload.

### Summary of operations for PersonsManager

Operation	Description		
createPersons	To request the creation of a set of populated 'person' records on the target system and the source is responsible for the allocation of each of the unique identifiers.		Not in keeping.
createByProxyPersons	To request the creation of a set of populated 'person' records on the target		Not in keeping.

	system and the target is responsible for the allocation of each the unique identifiers.		
deletePersons	To request the deletion of a set of 'person' record. The 'person' records and all their associated relationships are deleted.		Not in keeping.
readPersons	To read the full contents of the set of identified 'person' records. The target must return all of the data it has for each of the identified 'person' records.	GET http://server/personQueryForm?cql={..} payload= -  response = PersonSet XML	Effectively query
readPersonsForGroup	To retrieve the 'person' records for a particular Group. This returns the person record for every person that is a member of the Group i.e., for whom there is a membership record that associates the Person with this Group.	GET  http://server/group/{sourcedId}/persons payload = -  response = PersonSet XML	Looks like this is really group service
updatePersons	To write new content into the set of identified 'person' records. The target must write the new data into each of the 'person' records. These are additive operations.		Not in keeping
replacePersons	To replace the content of a set of identified 'person' records. The target		Not in keeping

	must write the new data into the 'person' records. These are destructive write-overs of all of the original information.		
changePersonsIdentifier	To change the sourcedId of the set of 'person' records. The completion of this operation will result in later actions using the original sourcedId reporting an unknown identifier status.		Not in keeping

**GroupManager core CRUD operations.**

Operation	Description		
createGroup	To request the creation of a populated 'group' record on the target system and the source is responsible for the allocation of the unique identifier.	PUT http://server/group/{sourcedId} payload = Group XML  response = -	
createByProxyGroup	To request the creation of a populated 'group' record on the target system and the target is responsible for the allocation of the unique identifier.	PUT http://server/group payload = Group XML  response = sourcedId	
deleteGroup	To request the deletion of a 'group' record. The 'group' record is deleted	DELETE http://server/group/{sourcedId}	

	and all of its associated relationships.	payload = - response = -	
deleteGroupRelationship	To request the deletion of a relationship within a 'group' record. This does not delete the associated 'group' records.	DELETE http://server/group/{sourcedId}/relationship/{relSourcedId} payload = - response = -	
readGroup	To read the full contents of the identified 'group' record. The target must return all of the data it has for the identified 'group' record.	GET http://server/group/{sourcedId} payload = - response = Group XML	
updateGroup	To write new content into the identified 'group' record. The target must write the new data into the 'group' record. This is an additive operation.	POST  http://server/group/{sourcedId} payload = Group XML response = -	
replaceGroup	To replace the content of the identified 'group' record. The target must write the new data into the 'group' record. This is a destructive write-over of all of the original information.	PUT  http://server/group/{sourcedId} payload = Group XML response = -	

changeGroupIdentifier	To change the sourcedId of the 'group' record. The completion of this operation will result in later actions using the original sourcedId reporting an unknown identifier status.	POST  http://server/group/{sourcedId} payload = sourcedId response = -	
-----------------------	---	--	--

### GroupsManager core CRUD operations

Operation	Description		
createGroups	To request the creation of a set of populated 'group' records on the target system and the source is responsible for the allocation of each of the unique identifiers.		Not in keeping
createByProxyGroups	To request the creation of a set of populated 'group' records on the target system. The target is responsible for the allocation of the unique identifiers.		Not in keeping
deleteGroups	To request the deletion of a set of 'group' record. The 'group' records and all their associated relationships are deleted.		Not in keeping

deleteGroupsRelationship	To request the deletion of a set of relationships within the 'group' records. This does not delete the associated 'group' records.		Not in keeping
readGroups	To read the full contents of the set of identified 'group' records. The target must return all of the data it has for each of the identified 'group' records.	GET http://server/groupQueryForm?cql={..} payload= -  response = GroupSet XML	
readGroupsForPerson	To retrieve the 'group' records for a particular Person. This returns the set of group records of which the person i.e., for whom a membership record in the Group exists.	GET http://server/person/{sourcedId}/groups payload = -  response = GroupSet XML	Looks like person service?
updateGroups	To write new content into the set of identified 'group' records. The target must write the new data into each of the 'group' records. These are additive operations.		Not in keeping
replaceGroups	To replace the content of a set of identified 'group' records. The target must write the new data into the 'group' records. These are destructive write-overs of all of the original information.		Not in keeping

changeGroupsIdentifier	To change the sourcedId of the 'group' record. The completion of this operation will result in later actions using the original sourcedId reporting an unknown identifier status.		Not in keeping
------------------------	---	--	----------------

### Summary of MembershipManager core operations

Operation	Description		
createMembership	To request the creation of a populated 'membership' record on the target system and the source is responsible for the allocation of the unique identifier.	PUT http://server/membership/{sourcedId} payload = Membership XML response = -	
createByProxyMembership	To request the creation of a populated 'membership' record on the target system and the target is responsible for the allocation of the unique identifier.	PUT http://server/membership payload = Membership XML response = sourcedId	
deleteMembership	To request the deletion of a 'membership' record. The 'membership' record is deleted and all of its associated relationships.	DELETE http://server/membership/{sourcedId} payload = - response = -	

readMembership	To read the full contents of the identified 'membership' record. The target must return all of the data it has for the identified 'membership' record.	GET  http://server/membership/{sourcedId} payload = - response = Membership XML	
updateMembership	To write new content into the identified 'membership' record. The target must write the new data into the 'membership' record. This is an additive operation.	POST  http://server/membership/{sourcedId} payload = Membership XML response = -	
replaceMembership	To replace the content of the identified 'membership' record. The target must write the new data into the 'membership' record. This is a destructive write-over of all of the original information.	PUT  http://server/membership/{sourcedId} payload = Membership XML response = -	
changeMembershipIdentifier	To change the sourcedId of the 'membership' record. The completion of this operation will result in later actions using the original sourcedId reporting an unknown identifier status.	POST  http://server/membership/{sourcedId} payload = sourcedId response = -	

**Summary of MembershipsManager core operations**

Operation	Description		
createMemberships	To request the creation of a set of populated 'group' records on the target system. The source is responsible for the allocation of the unique identifiers.		Not in keeping
createByProxyMemberships	To request the creation of a set of populated 'membership' records on the target system and the target is responsible for the allocation of each of the unique identifiers.		Not in keeping
deleteMemberships	To request the deletion of a set of 'membership' record. The 'membership' records and all their associated relationships are deleted.		Not in keeping
readMemberships	To read the full contents of the set of identified 'membership' records. The target must return all of the data it has for each of the identified 'membership' records.	GET http://server/membershipQueryForm?cql={..} payload= -  response = MembershipSet XML	
readMembershipsForPerson	To retrieve all the 'membership' records for a particular Person. This returns every 'membership'	GET	

	record for the identified 'person' record.	<p>http://server/person/{sourcedId}/memberships</p> <p>payload = -</p> <p>response = MembershipSet XML</p>	
readMembershipsForGroup	To retrieve all the 'membership' records for a particular Group. This returns every 'membership' record for the identified 'group' record.	<p>GET</p> <p>http://server/group/{sourcedId}/memberships</p> <p>payload = -</p> <p>response = MembershipSet XML</p>	Looks misplaced again
updateMemberships	To write new content into the set of identified 'membership' records. The target must write the new data into each of the 'membership' records. These are additive operations.		Not in keeping
replaceMemberships	To replace the content of a set of identified 'membership' records. The target must write the new data into the 'membership' records. These are destructive write-overs of all of the original information.		Not in keeping
changeMembershipsIdentifier	To change the sourcedId of the set of 'membership' records. The		Not in keeping

	completion of this operation will result in later actions using the original sourcedId reporting an unknown identifier status.		
--	--	--	--