Infrastructure - Bug #8641

Story # 8639 (New): Replication performance is too slow to service demand

Any change to SystemMetadata causes a new replication task to be generated without consideration of the object replicability

2018-07-04 12:28 - Dave Vieglais

Status:	New		**Start date:**	2018-07-04
Priority:	Normal		**Due date:**	
Assignee:	Chris Jones		**% Done:**	0%
Category:	d1_replication		**Estimated time:**	0.00 hour
Target version:				
Milestone:	None		**Story Points:**	
Product Version:	*			

Description

The hazelcast event listener implemented by ReplicationEventListener basically does:

```
ReplicationEventListener.entryUpdated()
  if isAuthoritativeReplicaValid()
    createReplicationTask()
```

isAuthoritativeReplicaValid() checks whether the replication status for the Authoritative MN is complete.

Hence, any update or add event on the systemmetadata map in Hazelcast will trigger addition of a replication task if the authoritative MN has a completed replica, even if replication is not allowed for the object. This causes a significant number of entries to be added to the replication task queue even though those tasks will never do anything as they will be later rejected.

It would be appropriate in entryUpdated() to also check whether replication of the object is allowed. The overhead would be minimal since a copy of the system metadata is already available in entryUpdated(). The same logic should also be added to entryAdded().

ReplicationManager implements boolean isAllowed(SystemMetadata sysmeta) which should do the job.