

Infrastructure - Task #1785

Task # 1782 (Closed): CN Replication components should be separated for scalability

CN SystemMetadata synchronization when an offline CN comes online

2011-09-11 19:57 - Chris Jones

Status:	Rejected	Start date:	2011-09-11
Priority:	Normal	Due date:	
Assignee:	Ben Leinfelder	% Done:	0%
Category:	Metacat	Estimated time:	0.00 hour
Target version:		Story Points:	
Milestone:	CCI-0.6.4		
Product Version:	*		

Description

Although updates to the hzSystemMetadata map will be reflected in memory across the Hazelcast storage cluster, only the backing store on the local Metacat (the one that 'owns' the entry in the cluster) will be updated. I confirmed this by looking at the Hazelcast code and documentation, which states that the MapStore interface is intended to update a *centralized* database backend. To remedy this, use the ExecutorService in Hazelcast to submit a CNReplicationTask to each CN in the cluster (except the local CN). This task should create/update the Metacat system metadata backing store with the new SystemMetadata object. If the PID exists in the map and *doesn't* exist the pgsq table, it should be a create, but if it exists in the pgsq table, it should be an update. Use Hazelcast to lock the object prior to writing to the table.

History

#1 - 2011-09-11 19:59 - Chris Jones

- Subject changed from Use Hazelcast ExecutorService to update remote CN Metacats to Use Hazelcast ExecutorService to update remote CN Metacat System Metadata

#2 - 2011-09-15 21:53 - Ben Leinfelder

- Status changed from New to Closed

- Assignee changed from Chris Jones to Ben Leinfelder

I opted for an alternative - I hope simpler - implementation. We use the custom MapStore and MapLoader implementation on the local node, but the other nodes are all listening for entryAdded/entryUpdated calls. When either even occurs, the other nodes check if they have the System Metadata entry locally. If not, the node writes it locally (or updates it if it is an update).

#3 - 2011-09-15 23:34 - Ben Leinfelder

- Status changed from Closed to In Progress

Matt is concerned about what happens when a node is offline - it cannot listen for the add/update. Either: a) we need a robust re-synch process or b) the CNReplicationTask queue approach needs to be used and it too needs a persistent backing store in case every node goes down

#4 - 2011-11-01 16:50 - Ben Leinfelder

- Subject changed from Use Hazelcast ExecutorService to update remote CN Metacat System Metadata to CN SystemMetadata synchronization when an offline CN comes online

#5 - 2012-01-03 16:55 - Chris Jones

- Status changed from In Progress to Rejected

CNReplicationTask isn't needed since Metacat is updated via change listeners on the system metadata map.