Infrastructure - Task #3787

Story # 3729 (Closed): Member Nodes should be the authoritative source of System Metadata

CN should detect changed sysmeta and resync objects when needed

2013-06-04 19:55 - Matthew Jones

Status:	Closed		**Start date:**	2013-06-04
Priority:	High		**Due date:**	
Assignee:	Rob Nahf		**% Done:**	100%
Category:	d1_synchronization		**Estimated time:**	0.00 hour
Target version:	CCI-2.0.0			
Milestone:	CCI-2.0		**Story Points:**	
Product Version:	2.0.0			

Description

The current CN assumes that certain fields in system metadata are immutable, and so does not notice when those fields are updated. This can arise when, for example, the formatid of an object changes, which change the object from DATA to either METADATA or RESOURCE types, which means the object needs to be re-synced (for an example, see ticket #3719).

The -CNCore.systemMetadataChanged- CN.synchronize call will be pushed to the CoordinatingNode from the MemberNode. Given the timing of the push call to -systemMetadataChanged- CN.synchronize and the regular harvesting of content from the MemberNode, a race condition may appear on the CN in which older data overwrites a newer update. Management of the sequence of regular synchronization harvesting and asychronous systemMetadataChanged will need to be designed for d1 processing and the cn rest interface.

To close this bug we need the following changes:

1) System metadata changes on the CN are detected when they are made and inspected to see if any operations need to be taken
2) if a new sync is required, the sync gets scheduled and executed, and the object is inserted into the CN metacat store as needed
3) metacat replication notices that the object needs to be replicated to the other CNs and does it
4) ensure order of synchronous and asynchronous sync events

Related issues:

Related to Infrastructure - Story #7364: update the sync logic for basic v1-v...	**Closed**	**2015-09-22**
Related to Infrastructure - Bug #7371: A systemMetadata update that changes t...	**New**	**2015-09-21**

History

#1 - 2013-06-04 20:05 - Matthew Jones

- Assignee changed from Skye Roseboom to Robert Waltz

#2 - 2013-10-28 20:04 - Robert Waltz

- Milestone changed from None to CCI-2.0

- Product Version set to 2.0.0

#3 - 2014-07-30 03:35 - Robert Waltz

- Assignee changed from Robert Waltz to Ben Leinfelder

#4 - 2014-07-30 20:40 - Robert Waltz

- Description updated

#5 - 2014-10-01 16:45 - Robert Waltz

- Target version set to CCI-2.0.0

#6 - 2015-06-12 16:39 - Rob Nahf

- Assignee changed from Ben Leinfelder to Rob Nahf

- % Done changed from 0 to 30

- Status changed from New to In Progress

Implementing CN.synchronize(pid) instead of CN.systemMetadataChanged(pid), with the behavior of CN.synchronize(pid) adding a SyncObjectTask to the existing queue should eliminate the race condition, except if duplicate SyncObjectTasts exist in the sync queue, and are processed at the same time.

We need to either make sure there's a lock on the pid for sync processing, or ensure that the sync queue doesn't already have the SyncObjectTask (look for duplicates).

#7 - 2015-06-18 23:10 - Rob Nahf

- Description updated

#8 - 2015-09-21 21:26 - Rob Nahf

synchronization uploads non-DATA objects and persists them using cn.create. For situations where the formatID is changed from a DATA format to a METADATA format, for example, the systemMetadata already exists on the CN, and the cn.create will fail due to existing checks on the existence of the identifier. Disabling those checks might impact the ban on reusing identifiers of deleted objects.

A new CN method may be needed to accomplish persisting the uploaded METADATA....

#9 - 2015-09-21 21:26 - Rob Nahf

- Related to Story #7364: update the sync logic for basic v1-v2 interoperability added

#10 - 2015-09-21 21:55 - Rob Nahf

- Related to Bug #7371: A systemMetadata update that changes the formatID from DATA to METADATA or RESOURCE_MAP will not upload the object to the CN added

#11 - 2015-11-03 17:00 - Rob Nahf

- % Done changed from 30 to 50

- Status changed from In Progress to Testing

#12 - 2015-11-03 17:23 - Rob Nahf

- Status changed from Testing to Closed

- translation missing: en.field_remaining_hours set to 0.0

- % Done changed from 50 to 100