

Infrastructure - Task #3539

Story # 3538 (Closed): Replication testing in cn-stage.test environment

mnStagePISCO and mnStageKNB size/checksum do not match on previously replicated objects

2013-01-31 05:31 - Ben Leinfelder

Status:	Closed	Start date:	2013-01-31
Priority:	Normal	Due date:	
Assignee:	Ben Leinfelder	% Done:	100%
Category:		Estimated time:	0.00 hour
Target version:	2014.2-Block.1.1	Story Points:	
Milestone:	None		
Product Version:			
Description I believe this is an issue with escaping characters in XML. Over the years I think Morpho and Metacat have approached this in different ways and we haven't always been perfect about preserving the original version of the document. in this example, the quote on PISCO is escaped with an entity reference that includes an ampersand that is also escaped with an ampersand, but then the ampersand is not escaped in KNB. Original on PISCO: " Replica on KNB: "			

History

#1 - 2013-01-31 05:33 - Ben Leinfelder

I have no idea how to address this in the real world. This will come up in so many documents and in so many ways. I should also mention that before the version of Metacat that stored the exact XML file to disk, we really cannot guarantee that replicated XML content will match byte-for-byte from source to replica.

#2 - 2013-01-31 05:39 - Ben Leinfelder

Those were:
https://test.piscoweb.org/catalog/d1/mn/v1/object/doi:10.6085/AA/BAYXXX_015MTBD009R00_20050525.50.3
and:
https://mn-stage-ucsb-1.test.dataone.org/knb/d1/mn/v1/object/doi:10.6085/AA/BAYXXX_015MTBD009R00_20050525.50.3

#3 - 2013-03-08 16:51 - Chris Jones

Ben and I discussed the following plan to test CN synchronization WRT objects that are replicas on different peering MNs via an out-of-band channel (like Metacat replication), but that have differing checksums due to minor whitespace or XML entity differences:

Test in CN sandbox

[CONFIRMED] confirm that the sync code ensures there's a replica entry when it comes across a replica on a replica node

change system metadata for a few sample pids for testing

Used doi:10.5072/FK2/LTER/knb-lter-gce.100.15

Update authMemberNode field to mnDemo3 since that's what the generate sysmeta code does now (AKA generate SM on production)

Remove replica entries in sysmeta on mnDemo3

Set serial version = to 1 since generated sysmeta will do the same

Evict the pid(s) from hzSystemMetadata so the changes are reflected in memory

Set mn-demo-3 to synchronize to the CNs

For some objects, update mnDemo3|4 to have incorrect checksum in sysmeta (doi:10.5072/FK2/LTER/knb-lter-gce.100.15)

The expected result is that CN synchronization would fail, and send a syncFailed message to the target MN (because of the differing checksum). Confirm this.

#4 - 2013-03-08 16:59 - Chris Jones

- Status changed from New to In Progress

In testing the example pid (doi:10.5072/FK2/LTER/knb-lter-gce.100.15), I changed the various system metadata fields, and set the checksum to: 0123456789ABCDEF0123456789ABCDEF. After updating the MN to sync to the CN, the CN synchronization logs reported:

```
[ INFO] 2013-03-08 16:39:11,027 (SyncObjectTask:call:216) Task-urn:node:mnDemo3-doi:10.5072/FK2/LTER/knb-lter-gce.100.15 submitted for execution
[ INFO] 2013-03-08 16:39:11,028 (TransferObjectTask:call:117) Task-urn:node:mnDemo3-doi:10.5072/FK2/LTER/knb-lter-gce.100.15 Locking task of attempt 1
[ INFO] 2013-03-08 16:39:11,062 (TransferObjectTask:call:123) Task-urn:node:mnDemo3-doi:10.5072/FK2/LTER/knb-lter-gce.100.15 Processing task
[DEBUG] 2013-03-08 16:39:11,143 (SyncObjectTask:call:132) trying future Task-urn:node:mnDemo3-doi:10.5072/FK2/LTER/knb-lter-gce.100.15
[DEBUG] 2013-03-08 16:39:11,395 (SyncObjectTask:call:163) Task-urn:node:mnDemo3-doi:10.5072/FK2/LTER/knb-lter-gce.100.15Waiting for the future :(1): since 2013-03-08T16:39:11.018+00:00
[DEBUG] 2013-03-08 16:39:11,433 (SyncObjectTask:call:132) trying future Task-urn:node:mnDemo3-doi:10.5072/FK2/LTER/knb-lter-gce.100.15
[DEBUG] 2013-03-08 16:39:11,689 (SyncObjectTask:call:163) Task-urn:node:mnDemo3-doi:10.5072/FK2/LTER/knb-lter-gce.100.15Waiting for the future :(1): since 2013-03-08T16:39:11.018+00:00
[ INFO] 2013-03-08 16:39:11,964 (TransferObjectTask:retrieveSystemMetadata:253) Task-urn:node:mnDemo3-doi:10.5072/FK2/LTER/knb-lter-gce.100.15 Retrieved SystemMetadata Identifier:doi:10.5072/FK2/LTER/knb-lter-gce.100.15 from node urn:node:mnDemo3 for ObjectInfo Identifier doi:10.5072/FK2/LTER/knb-lter-gce.100.15
[ INFO] 2013-03-08 16:39:11,964 (TransferObjectTask:call:126) Task-urn:node:mnDemo3-doi:10.5072/FK2/LTER/knb-lter-gce.100.15 Writing task
[ INFO] 2013-03-08 16:39:11,965 (TransferObjectTask:write:379) Task-urn:node:mnDemo3-doi:10.5072/FK2/LTER/knb-lter-gce.100.15 Getting sysMeta from CN
[ INFO] 2013-03-08 16:39:11,990 (TransferObjectTask:write:401) Task-urn:node:mnDemo3-doi:10.5072/FK2/LTER/knb-lter-gce.100.15 Pid Exists. Must be an Update
[DEBUG] 2013-03-08 16:39:11,990 (SyncObjectTask:call:132) trying future Task-urn:node:mnDemo3-doi:10.5072/FK2/LTER/knb-lter-gce.100.15
[ INFO] 2013-03-08 16:39:11,996 (TransferObjectTask:write:425) Task-urn:node:mnDemo3-doi:10.5072/FK2/LTER/knb-lter-gce.100.15 Update sysMeta Not Unique! Checksum is different
[DEBUG] 2013-03-08 16:39:12,242 (SyncObjectTask:call:163) Task-urn:node:mnDemo3-doi:10.5072/FK2/LTER/knb-lter-gce.100.15Waiting for the future :(1): since 2013-03-08T16:39:11.018+00:00
[DEBUG] 2013-03-08 16:39:12,688 (SyncObjectTask:call:132) trying future Task-urn:node:mnDemo3-doi:10.5072/FK2/LTER/knb-lter-gce.100.15
[DEBUG] 2013-03-08 16:39:12,940 (SyncObjectTask:call:163) Task-urn:node:mnDemo3-doi:10.5072/FK2/LTER/knb-lter-gce.100.15Waiting for the
```

future :(1): since 2013-03-08T16:39:11.018+00:00

[DEBUG] 2013-03-08 16:39:13,095 (TransferObjectTask:call:195) Task-urn:node:mnDemo3-doi:10.5072/FK2/LTER/knb-lter-gce.100.15 Unlocked task

[DEBUG] 2013-03-08 16:39:13,492 (SyncObjectTask:call:132) trying future Task-urn:node:mnDemo3-doi:10.5072/FK2/LTER/knb-lter-gce.100.15

[DEBUG] 2013-03-08 16:39:13,493 (SyncObjectTask:call:139) Task-urn:node:mnDemo3-doi:10.5072/FK2/LTER/knb-lter-gce.100.15 Returned from the Future :1:

The MN received the syncFailed() call for the pid as expected. The solution (unfortunately), is to purge the offending objects off of the replica MNs, including the identifier in the identifier table such that the replica MN has no knowledge of these objects. MN.delete() may be useful, but will not suffice because the pid will remain in the identifier table. A combination of MN.delete(), a SQL DELETE statement, and a hazelcast evict() or remove() call will be needed to fully purge the offending pids. Next up: determine the extent of the problematic objects across PISCO/KNB/LTER MNs.

#5 - 2013-03-08 19:20 - Chris Jones

- Assignee changed from Chris Jones to Ben Leinfelder

Assigning back to Ben - he already has written an UpgradeEmptyReplicatedDataFile class that can be modified to remove replicated documents with incorrect bytes and checksums. Our plan is to allow the CNs to attempt to sync all content from the replica MNs (KNB and LTER), fail on a subset, and then we will get the short list of failed pids from the metacat access_log where event = 'synchronization_failed' and the local server is not the home sever for the pid in xml_documents.

#6 - 2013-03-08 23:54 - Ben Leinfelder

Wrote new utility for removing these failed/invalid replicas.

ant runoneclass -Dclassstorun=edu.ucsb.nceas.metacat.admin.upgrade.RemoveInvalidReplicas

allows you to run this on your metacat deployment and will be included in the 2.0.6 distribution.

I think I will also incorporate this class into the 2.0.6 upgrade process so that it is performed automatically in cases where Metacat admins have generated System Metadata for replicas they house and wish to correct the failed synchronization attempts.

#7 - 2013-03-13 15:49 - Chris Jones

We needed to confirm that given a source MN that has replicas via non-DataONE API channels, and those replicas are valid (correct checksum, byte size, etc.), that the replica copies on the replica MNs get registered via synchronization. I've cleared the sandbox CN environment completely, deleted all replica entries on mn-demo-3 and mn-demo-4 (in postgresql), restarted synchronization, and have confirmed that the existing replica objects on mn-demo-3 and mn-demo-4 get registered in the system metadata replica list. Given the utility that Ben wrote up, I think we can confidently remove all invalid system metadata from the KNB and LTER nodes, and use D1 replication to re-transfer those objects.

#8 - 2013-08-14 18:08 - Dave Vieglais

- Target version set to 2013.33-Block.4.4

#9 - 2014-01-06 18:01 - Chris Jones

- Target version changed from 2013.33-Block.4.4 to 2014.2-Block.1.1

#10 - 2014-03-14 21:06 - Chris Jones

- translation missing: en.field_remaining_hours set to 0.0

- Status changed from In Progress to Closed

Closing this task since we have generated system metadata for KNB/PISCO/LTER objects, and the replicas are being registered correctly on the CNs.