

Infrastructure - Task #7655

Story # 7652 (Closed): Enable simple metrics reporting for core services

report on replication activity

2016-02-23 19:36 - Dave Vieglais

Status:	Closed	Start date:	2016-02-23
Priority:	Normal	Due date:	
Assignee:	Rob Nahf	% Done:	100%
Category:	performance-scalability	Estimated time:	0.00 hour
Target version:	CCI-2.2.0	Story Points:	
Milestone:	None		
Product Version:	*		
Description			
At a minimum, report on the size of the replication backlog.			
Results should be written to a file that is easily parsed with various languages. Two common options include JSON and CSV. A CSV file would be convenient for appending entries if more than a single record is to be kept.			
Contents of the file are to be determined but should include the size of the replication backlog, and may include additional per member node information such as the timestamp of the last replication activity and perhaps the number of current replication tasks for that MN.			

Associated revisions

Revision 18057 - 2016-06-08 23:00 - Rob Nahf

refs: #7655: added ReplicationTaskMonitor (runnable) to report replicationTask queue statistics (by status and authMN). Commented out unused 'public InputStream ReplicationService#getObjectFromCN() method.

Revision 18057 - 2016-06-08 23:00 - Rob Nahf

refs: #7655: added ReplicationTaskMonitor (runnable) to report replicationTask queue statistics (by status and authMN). Commented out unused 'public InputStream ReplicationService#getObjectFromCN() method.

Revision 18058 - 2016-06-09 00:49 - Rob Nahf

refs: #7655: added ReplicaStatusMonitor (runnable) to report summary counts of replicas by target node and status. Added Replication MetricEvents.

Revision 18058 - 2016-06-09 00:49 - Rob Nahf

refs: #7655: added ReplicaStatusMonitor (runnable) to report summary counts of replicas by target node and status. Added Replication MetricEvents.

Revision 18069 - 2016-06-09 20:19 - Rob Nahf

refs: #7655. parameterized replication monitoring frequency.

Revision 18069 - 2016-06-09 20:19 - Rob Nahf

refs: #7655. parameterized replication monitoring frequency.

Revision 18074 - 2016-06-09 21:32 - Robert Waltz

refs: #7655

parameterized replication monitoring frequency. fix config properties

refs: #7655

parameterized replication monitoring frequency. fix config properties

History

#1 - 2016-05-26 22:44 - Rob Nahf

replicationDAO has methods to determine how many outstanding requests there are per MN, and hooks into ReplicationManager for reporting. It should be relatively straightforward to schedule a task that runs at regular intervals to generate monitoring statistics.

We may need to add more DAO queries to not filter by task state, or queries that return counts instead of records. (located in ReplicationDaoMetacatImpl in d1_cn_common.

#2 - 2016-05-27 21:29 - Rob Nahf

one pid can be associated with [0..n] replicas (identified by [pid, targetMN])

so, replication has two things that can be potentially backlogged:

1. the pid that has been picked up by the sysmeta map listener (the sysmeta has changed, so need to re-evaluate if enough replicas were created)
2. replica themselves that have been ordered to be created on a target node. (a request issued to a replica node, but not completed).

The first type could be reported per authoritativeMN.

The second by target node. The StaleReplicationRequestAuditor looks for these and tries to address them.

pid statuses:

NEW - the listener picked up a pid to evaluate

IN_PROCESS - a processor picked the pid up to evaluate

(there is no complete status. I believe the item is removed from the repo.)

replica statuses:

QUEUED, REQUESTED, COMPLETED, FAILED, INVALIDATED

#3 - 2016-06-10 16:18 - Robert Waltz

- % Done changed from 0 to 30

- Status changed from New to In Progress

#4 - 2016-06-10 16:18 - Robert Waltz

- % Done changed from 30 to 50

- Status changed from In Progress to Testing

#5 - 2016-06-14 19:45 - Robert Waltz

- Status changed from Testing to Closed

- translation missing: en.field_remaining_hours set to 0.0

- % Done changed from 50 to 100