

Infrastructure - Story #8520

The SeriesIdResolver class in d1_index_processor shouldn't use the D1Client.getCN().getSystemMetadata method

2018-03-23 20:15 - Jing Tao

Status:	Closed	Start date:	2018-03-23
Priority:	High	Due date:	
Assignee:	Jing Tao	% Done:	100%
Category:	d1_indexer	Estimated time:	0.00 hour
Target version:	CCI-2.3.10		
Story Points:			

Description

The d1_index_processor component is shared by both the cn indexer and the Metacat mn indexer.

The the method getPid(Identifier sid) on the SeriesIdResolver class, it use the line:

```
SystemMetadata fetchedSysmeta = D1Client.getCN().getSystemMetadata(null, identifier);
```

It works perfectly with the CN indexers. For MNs, it may have glitches:

1. The head object hasn't been synchronized to cn when the index happens in the MN.
2. MN maybe is configured NOT to being synchronized at all.

History

#1 - 2018-04-27 14:55 - Rob Nahf

- Assignee changed from Dave Vieglais to Rob Nahf

#2 - 2018-09-21 22:55 - Rob Nahf

the rest call is made instead of hazelcast call to get sid-resolution behavior. This incurs a lot of http and authorization overhead.

Is there a better way?

#3 - 2018-09-21 22:57 - Rob Nahf

- Subject changed from *The SeriesIdResolver class in d1_index_processor shouldn't use the D1Client.getCN().getSystemMetacat method* to *The SeriesIdResolver class in d1_index_processor shouldn't use the D1Client.getCN().getSystemMetadata method*

#4 - 2018-09-22 00:53 - Rob Nahf

Perhaps this class can be turned into an interface, and the resolve can be wired in via spring. This would allow the MN to have a separate implementation that doesn't require the CN (or the object to be synchronized with a CN).

#5 - 2018-10-26 16:45 - Chris Jones

- Subject changed from *The SeriesIdResolver class in d1_index_processor shouldn't use the D1Client.getCN().getSystemMetacat method* to *The SeriesIdResolver class in d1_index_processor shouldn't use the D1Client.getCN().getSystemMetadata method*

#6 - 2018-10-26 17:05 - Chris Jones

- Assignee changed from Rob Nahf to Jing Tao

- Priority changed from Normal to Immediate

Jing, since we have a critical issue with the ESS-DIVE MN that this bug affects, I'm re-assigning this to you and bumping the priority. While Rob's idea of creating an interface may be a longer-term solution, I'm thinking that a shorter term solution would be to add a property that sets which host use for getSystemMetadata() calls. Let's discuss this, and at least get an RC tag out the door that we can try on ESS-DIVE.

#7 - 2018-10-27 04:46 - Chris Jones

- Priority changed from Immediate to High

Jing and I discussed this ticket. We've found a workaround with the ESS-DIVE MN so that their packages with members that have seriesId synchronize correctly. We had to set the rightsHolder back to the original group DN that had been set a few objects back in the chain (it had been set to the ORCID id of the owning researcher). By doing this, the indexer matched the rightsHolder Subject string with the rightsHolder Subject string of the current HEAD of the series, so it determined that the Subject was authorized (i.e. no series hijacking).

This illuminated the issue: the SeriesIdResolver class is not expanding group membership, causing NotAuthorized exceptions for group members. This will be fixed.

Also, having the Member Node rely on CN.getSystemMetadata() for local indexing of content can be problematic for MNs not registered in any CN environment. So, our plan is to add a baseUrl configuration parameter in Metacat (that defaults to being unset) so the indexer can locally call MN.getSystemMetadata() (Jing noticed that the indexer doesn't have access to the Hazelcast map, so there's no other internal call to use.) In this way, MNs can be configured to use their own sysmeta, but still call CN.getSubjectInfo() for group expansion when needed. CN deployments will work as they do now, still calling CN.getSystemMetadata().

I'm lowering the urgency of this ticket since we found an immediate workaround, but it's still high priority. It has arisen because ESS-DIVE is using the seriesId field to the fullest extent (assigning DOIs there), and I think they may be the first MN to really exercise this, and these bugs are being exposed just from daily use.

#8 - 2018-10-27 04:57 - Jing Tao

Chris, thank you for the summary. Just one thing need to be clarified - even though the indexer can access the Hazelcast map internally, the return result is NOT the head version of the series chain if you call systemmetamap.get(seriesId). Only the api calls (cn/mn.getSystemMetadata(seriesId)) can return the head version of the series chains.

#9 - 2018-11-20 22:19 - Jing Tao

- Status changed from New to Closed

- % Done changed from 0 to 100

In this method, we will check if a property named mn.dataone.baseUrl is defined. If it is defined, it will use the value of this property to get the head version of the system metadata. Otherwise, it will still call cn.getSystemMetadata.

It works for both CN and MN.