

Infrastructure - Task #8700

downgrade dependencies in indexing trunk to remain consistent with solr 5.2.1

2018-09-20 20:15 - Rob Nahf

Status:	Closed	Start date:	2018-09-20
Priority:	Normal	Due date:	
Assignee:	Rob Nahf	% Done:	100%
Category:	d1_indexer	Estimated time:	0.00 hour
Target version:	CCI-2.4.0	Story Points:	
Milestone:	None		
Product Version:	*		

Description

while it would be great to keep upgraded dependencies in trunk code, upgrading solr and solrj past 5.2.1 causes issues outside the indexing stack of components.

At question were:

solr (core)
solrj (client)
spring (everything)
httpClient
junit

In the end, I needed to downgrade solrj to 5.2.1 and httpClient to 4.3.3 due to incompatibilities between solr and solrj. the solr test framework uses solrj client, and was causing ClassNotFoundException exceptions in d1_cn_index_processor tests.

Other than that, d1_cn_common, d1_cn_index_common, and d1_cn_index_generator were all happy with solr 7.1.0, and httpClient 4.5.3.=

Associated revisions

Revision 19339 - 2018-09-20 20:21 - Rob Nahf

refs: #8700: downgrading spring, solr, solrj, compiler, and httpClient dependency versions to what's in the 2.3 branch and production. Changes made across the d1_cn_common, d1_cn_index_common, d1_cn_index_generator, and d1_cn_index_processor components. There is a refactored HttpSolrClient construction method that needed refactoring. Downgrading loses the builder in favor of the "new" method.

Revision 19339 - 2018-09-20 20:21 - Rob Nahf

refs: #8700: downgrading spring, solr, solrj, compiler, and httpClient dependency versions to what's in the 2.3 branch and production. Changes made across the d1_cn_common, d1_cn_index_common, d1_cn_index_generator, and d1_cn_index_processor components. There is a refactored HttpSolrClient construction method that needed refactoring. Downgrading loses the builder in favor of the "new" method.

History

#1 - 2018-09-21 20:40 - Rob Nahf

- % Done changed from 0 to 30
- Status changed from New to In Progress
- Description updated

Notes:

- we are running junit 4.12, which is not supposed to be compatible with SolrJettyTestBase, but seems to be working. The issue is something to do with authorization, but the @SuppressSSL annotation patch may have fixed the compatibility issue. (see class SolrTestCaseJ4.SuppressSSL)

#2 - 2018-09-21 20:40 - Rob Nahf

- % Done changed from 30 to 100

- Status changed from In Progress to Closed