Item 87
git_comments:

1. ----- try to add a collection with a path that doesn't exist
2. ---- fix col_bad's config (again) and RELOAD to fix failure
3. start with two collections: one valid, and one broken
4. our "ok" collection
5. ----- init the CoreContainer with the mix of ok/bad cores
6. ----- break col_bad's config and try to RELOAD to add failure
7. our "bad" collection
8. :TODO: should really tighten up the exceptions CoreContainer throws (ie: just SolrException)
9. * Licensed to the Apache Software Foundation (ASF) under one or more * contributor license agreements. See the NOTICE file distributed with * this work for additional information regarding copyright ownership. * The ASF licenses this file to You under the Apache License, Version 2.0 * (the "License"); you may not use this file except in compliance with * the License. You may obtain a copy of the License at * * http://www.apache.org/licenses/LICENSE-2.0 * * Unless required by applicable law or agreed to in writing, software * distributed under the License is distributed on an "AS IS" BASIS, * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. * See the License for the specific language governing permissions and * limitations under the License.
10. reused state
11. would be nice to do this in an @Before method, but junit doesn't let @Before methods have test names
12. check that we have the failures we expect
13. let the test end here, with some recorded failures, and let cleanUp() verify that there is no problem shuting down CoreContainer with known SolrCore failures
14. ----- register bogus as an alias for col_ok and confirm failure goes away
15. check that we have the cores we expect
16. ----- "fix" the bad collection
17. solr.xml
18. ---- init the CoreContainer
19. * * Returns an immutable Map of Exceptions that occured when initializing * SolrCores (either at startup, or do to runtime requests to create cores) * keyed off of the name (String) of the SolrCore that had the Exception * during initialization. * <p> * While the Map returned by this method is immutable and will not change * once returned to the client, the source data used to generate this Map * can be changed as various SolrCore operations are performed: * </p> * * Failed attempts to create new SolrCores will add new Exceptions. * Failed attempts to re-create a SolrCore using a name already contained in this Map will replace the Exception. * Failed attempts to reload a SolrCore will cause an Exception to be added to this list -- even though the existing SolrCore with that name will continue to be available. * Successful attempts to re-created a SolrCore using a name already contained in this Map will remove the Exception. * Registering an existing SolrCore with a name already contained in this Map (ie: ALIAS or SWAP) will remove the Exception. *
20. remove first so insertion order is updated and newest is last
21. :TODO: would be really nice if this method wrapped any underlying errors and only threw SolrException
22. :TODO: Java7... http://docs.oracle.com/javase/7/docs/technotes/guides/language/catch-multiple.html
23. :NOOP: :TODO: CoreAdminHandler's exception messages are terrible, otherwise we could asert something useful here
24. attempt to create a bogus core and confirm failure
25. :TODO: because of SOLR-3665 we can't ask for status from all cores
26. check specificly for status of the failed core name

git_commits:

1. **summary:** SOLR-3634: CoreContainer and CoreAdminHandler will now remember and report back information about failures to initialize SolrCores (merge r1366241 + r1366249)
   message: SOLR-3634: CoreContainer and CoreAdminHandler will now remember and report back information about failures to initialize SolrCores (merge r1366241 + r1366249) git-svn-id: https://svn.apache.org/repos/asf/lucene/dev/branches/branch_4x@1366254 13f79535-47bb-0310-9956-ffa450edef68

github_issues:

github_issues_comments:

github_pulls:

github_pulls_comments:

github_pulls_reviews:

jira_issues:

1. **summary:** make CoreAdminHandler STATUS action report back info about core initialization failures
   description: see parent issue for context. * CoreContainer should keep a record of failures related to core initialization, keyed off of the core name * failures should be recorded for both solr.xml initialzation, as well as explicit CREATE/RELOAD requests that fail * record of failures should be tracked indefinitely until some valid core begins using that name ** ie: actions which register a SolrCore with a name (RELOAD, CREATE, SWAP, ALIAS) should "reset" any existing failures associated with that name * failures should be available to clients via CoreAdminHandler STATUS actions

jira_issues_comments:

1. **body:** quick patch with all the major functionality and some tests. lots of nocommits in here, basicly all falling into three categories: * refactoring - the changes to CoreContainer are all complete and relatively trivial, but involved renaming create/reload to doCreate/doReload which are wrapped in new stub methods that call them in try blocks where the "finally" does all the real work. This was just for convinience to keep the patch easy to read w/o a lot of re-indenting, and these methods should be refactored away before committing. * more tests - the key stuff is already covered in the patch, and i did manual testing with CoreAdminHandler, but i definitely want more tests before committing. * "Directory not closed" - something about this test is causing MockDirWrapper to complain it's never closed - buti don't think this is a bug in the new code or the test. I suspect we have an existing bug where SolrCore initialization failures *after* the Directory has already been opened aren't closing the Directory -- so we should get to hte bottom of that (probably in a new issue if i'm correct)
   label: test
2. **body:** Updated patch... * lots of new tests, this covers all the cases i can think of except for what we can't test because of SOLR-3665 * added detailed javadocs for the new public CoreContainer.getCoreInitFailures method * the directory not closed problem vanished on it's own -- i suspect SOLR-3656 fixed it. There are still some nocommits in this patch because of the previously mentioned refactoring (left as is so the patch is readable -- all the refactoring will really do is indent a ton of code inside a try/catch) I think this is good to go, but i'd appreciate some feedback
   label: code-design
3. Committed revision 1366241. - trunk Committed revision 1366249. - trunk fix nocommit refactoring i forgot about Committed revision 1366254. - 4x
4. @Ignore("SOLR-3634: some errors are localized and tests fail in non en, disabling temporarily") Committed revision 1366460. - trunk Committed revision 1366464. - 4x
5. change test to work even on java7 where SAXParseException.getMessage() returns a localized msg even though that defeats the whole point of Throwable.getLocalizedMessage() Committed revision 1366568. - trunk Committed revision 1366569. - 4x