



This repository Search

Explore Features Enterprise Blog

Sign up

Sign in



neo4j / neo4j

Watch 238

Star 1,885

Fork 730

# Add neo4j-logging component

Merged

chrisvest merged 5 commits into neo4j:2.3 from cleishm:logging Apr 14, 2015

Conversation 12

Commits 5

Files changed 599

+6,212 -6,214



cleishm commented Apr 6, 2015

Owner

And replace use of other logging frameworks with it.



cleishm added kernel NOT READY FOR MERGE 2.3 labels Apr 6, 2015



rickardoberg commented Apr 8, 2015

Collaborator

A general question with the design of rotation is why not let log calls put LogMessages on a ring buffer, and have a separate thread continuously pull messages and log them. Then there is no penalty wrt io, flushing, rotation, and so on, for the thread logging the message. It would simplify things a lot, I think.



cleishm commented Apr 8, 2015

Owner

@rickardoberg That's definitely an implementation option we could add: it would certainly be more performant. If we were doing high-volume logging, I'd definitely want that. For our current low-volume diagnostic logging, it is simpler to use the same thread, and there's much less concern that a log message will be missing in a crash. If we do find a performance constraint, then we should definitely review that.



cleishm referenced this pull request Apr 9, 2015

Add JobScheduler#executor(group) #4393

Merged



lutovich commented Apr 10, 2015

Collaborator

I think this PR looks good. It makes our logging process much more straightforward. Vote for merging it in. More reviewers required.



cleishm removed the NOT READY FOR MERGE label Apr 10, 2015



chrisvest commented Apr 10, 2015

Collaborator

This PR depends (actually includes) #4393, which I think should go in first.



cleishm commented Apr 10, 2015

Owner

#4393 is now in



tinwelint commented Apr 13, 2015

Collaborator

If we keep all class/constant names (not necessarily package names), i.e. StringLogger / Logging / DEV\_NULL this big change will not affect merging and conflict resolution much at all between < 2.3 and >= 2.3, except imports, but that's just a auto-optimize imports away anyway. Now many changes will conflict, while most perhaps simple, it will still always be a bit more annoying than before to merge things forward. I think we should consider this.

Labels

2.3

kernel

Milestone

No milestone

Assignee

No one assigned

7 participants





**chrisvest** commented Apr 13, 2015

Collaborator

@tinwelint I think most 2.2 changes from now on will be in parts that don't have much logging, and log calls is not the hardest code to upgrade. So I'm inclined to think it'll be fine.



**tinwelint** commented Apr 13, 2015

Collaborator

@chrisvest you're probably right. I have no strong opinion about it.



**benbc** commented Apr 13, 2015

Owner

@jakewins Does this mesh with your proposed change to logging externals?



**cleishm** added some commits Apr 10, 2015



Add neo4j-logging module

74cc8f0



Use neo4j-logging for internal and console logging (kernel)

7034419



Use neo4j-logging for internal and console logging (community)

1eec38f



Use neo4j-logging for internal and console logging (enterprise)

bf17bc6



Use neo4j-logging for internal and console logging (packaging)

d0a9ce4



**chrisvest** merged commit **ce676da** into **neo4j:2.3** Apr 14, 2015  
1 check passed

[View details](#)



**cleishm** deleted the **cleishm:logging** branch Apr 14, 2015



**jotomo** commented on the diff Jun 23, 2015

...java/org/neo4j/unsafe/batchinsert/BatchInserters.java

[View full changes](#)

		@@ -46,7 +47,7 @@
46	47	* @param storeDir the store directory
47	48	* @return a new {@link BatchInserter}
48	49	*/
49		- public static BatchInserter inserter( String storeDir )
	50	+ public static BatchInserter inserter( String storeDir ) throws IOException



**jotomo** added a note Jun 23, 2015

I have mixed feelings about this ... breaks compilation when upgrading 2.2 -> 2.3 due to this added checked exception. The IOException comes out of the constructor of *org.neo4j.kernel.impl.logging.StoreLogService#StoreLogService* (well, from whatever he's doing). What meaningful thing is a client to do with an IOException when something around the logging fails other than crash? Isn't this specific exception more of a neo4j-internal detail, the logging service specifically?



**cleishm** added a note Jun 23, 2015

Owner

Good catch - breaking the API here is bad. I'll put a change together to avoid that.

That said - it'll still get an `IOException`, just wrapped into a `RuntimeException`. From the client perspective, an `IOException` means what an `IOException` typically does: it couldn't write somewhere. In this case, it couldn't write to something in the specified `storeDir`. Client could handle it by reporting that a failure occurred accessing the specified directory - but reality is that they'll probably just let it die.



**cleishm** added a note Jun 23, 2015

Owner

PR is here: [#4885](#)

Add a line note

Sign up for free

to join this conversation on GitHub. Already have an account? [Sign in to comment](#)

