EDIT - bug #9236

stopping an instance breaks the centralized logging

10/01/2020 05:38 PM - Andreas Kohlbecker

Status:	Closed		**Start date:**	
Priority:	Highest		**Due date:**	
Assignee:	Andreas Kohlbecker		**% Done:**	50%
Category:	cdmserver		**Estimated time:**	0:00 hour
Target version:	Release 5.29			
Severity:	critical		**Found in Version:**	Release 5.17

Description

since switching to centralized logging through logback MDC ([#7085](#)) the logging gets broken when an cdm-webapp instance in the cdm-server is being stopped. Also (some) web services are broken afterwards.

To reproduce:

- stop an instance in the cdm-server management console
- ==> the last entry in the log file is e.e.cdm.server.instance.CdmInstance - lifeCycleStopping and no more logs are received.

IMPORTANT This has also observed on the production sever in a situation when some cdm instances have showed strange behavior. So this issue is critical and it is highly recommended **not to stop an instance on the production server as long as this issue is not solved**.

Related issues:

Related to EDIT - bug #7085: cdmserver logfiles getting too big, use RollingF...		**Closed**

Associated revisions

Revision 7ba74c60 - 01/13/2022 04:53 PM - Andreas Kohlbecker

fix #9236 solving broken logging by excluding the LogbackServletContainerInitializer from the configuration

Revision dbdfb62b - 01/13/2022 05:56 PM - Andreas Kohlbecker

ref #9236 better solution to solve the broken logging: disable the LogbackServletContainerInitializer per webapp init parameter

---

History

#1 - 10/01/2020 05:38 PM - Andreas Kohlbecker

- Related to bug #7085: cdmserver logfiles getting too big, use RollingFileAppender added

#2 - 10/01/2020 05:46 PM - Andreas Kohlbecker

- Description updated

Further test showed that some service endpoints in fact are broken in the situation described above.

#3 - 10/01/2020 05:47 PM - Andreas Kohlbecker

- Description updated

#4 - 10/01/2020 05:49 PM - Andreas Kohlbecker

- Subject changed from stopping an intance breaks the centralized logging to stopping an instance breaks the centralized logging

#5 - 01/27/2021 03:14 PM - Andreas Kohlbecker

- Target version changed from Release 5.19 to Release 5.21

#6 - 03/10/2021 02:55 PM - Andreas Müller

- Target version changed from Release 5.21 to Release 5.22

#7 - 04/19/2021 04:35 PM - Andreas Kohlbecker

- Target version changed from Release 5.22 to Release 5.25

#8 - 07/09/2021 10:42 AM - Andreas Müller

- Target version changed from Release 5.25 to Release 5.45

#9 - 01/13/2022 11:29 AM - Andreas Kohlbecker

The problem is caused by the LoggerContext being shutdown by the LogbackServletContextListener in
LogbackServletContextListener.contextDestroyed(ServletContextEvent sce).
This effectively stops the shared logging context of all web-app instances.

The proposed solution is to

- unregister the LogbackServletContextListener, or
- to register a alternative ContextListener that checks the number of cdm-remote instances that are still running. The LoggerContext only is shut
  down when no more instances are available.

#10 - 01/13/2022 12:05 PM - Andreas Müller

- Status changed from New to In Progress

#11 - 01/13/2022 04:53 PM - Andreas Kohlbecker

- Status changed from In Progress to Resolved

- % Done changed from 0 to 50

Applied in changeset cdm-server|7ba74c603e769569dd1b0052949cd321c86722da.

#12 - 01/13/2022 04:54 PM - Andreas Kohlbecker

- Assignee changed from Andreas Kohlbecker to Katja Luther

solved by excluding the LogbackServletContainerInitializer from the configuration.

Please review

#13 - 01/14/2022 10:09 AM - Katja Luther

- Status changed from Resolved to Closed

- Assignee changed from Katja Luther to Andreas Kohlbecker

This works as expected.

#14 - 02/15/2022 03:40 PM - Andreas Müller

- Target version changed from Release 5.45 to Release 5.29