AsyncProcessorAwaitManager

AsyncProcessorAwaitManager

Available as of Camel 2.15

The AsyncProcessorAwaitManager is used by the routing engine to mange blocked threads waiting for a signal to trigger before the thread should wake up and continue routing the exchange. This is only in use when messages are being routed and there is a handover from synchronous to asynchronous, and the callee thread needs to block while the asynchronous routing continues, and when that part is done, it signals its done by calling the AsyncCallback. The AsyncProcessorAwaitManager is the manager that keeps track of the blocked threads, and the exchange they are correlated with. The manager also offers insight at runtime which allows to see the state from JMX or Java API.

The manager has the following options

Option	Default	Description
interruptThreadsWhileStopping	true	Sets whether to interrupt any blocking threads during stopping. When true then Camel will attempt to shutdown any pending blocked threads during shutdown of Camel itself.
statisticsEnabled	false	Whether utilization statistics is enabled.

When Camel shutdown the manager also reports if there is any pending blocked threads left, and attempt to shutdown these blocked threads so they are not left hanging in the JVM. It is also possible to force a blocked thread to wake up and terminate, at runtime using the Java API or JMX, using the interrupt method.

The following demonstrates what is being logged, as an example, what happens when Camel is being shutdown, and there is some blocked threads that hasn't been released yet.

```
2014-12-21 08:38:03,321 [pool-2-thread-1] WARN  aultAsyncProcessorAwaitManager - The following threads are
blocked and will be interrupted so the threads are released:
Blocked Thread
--------------------------------------------------------------------------------------------------------------
----------------------
        Id:                 1
        Name:               main
        RouteId:            myRoute
        NodeId:             myAsync
        Duration:           19987 msec.
2014-12-21 08:38:03,328 [pool-2-thread-1] WARN  aultAsyncProcessorAwaitManager - Interrupted while waiting for
asynchronous callback, will release the following blocked thread which was waiting for exchange to finish
processing with exchangeId: ID-davsclaus-air-49173-1419147462911-0-2
Blocked Thread
--------------------------------------------------------------------------------------------------------------
----------------------
        Id:                 1
        Name:               main
        RouteId:            myRoute
        NodeId:             myAsync
        Duration:           19989 msec.
Message History
--------------------------------------------------------------------------------------------------------------
----------------------
RouteId              ProcessorId
Processor                                                                          Elapsed (ms)
[myRoute           ] [myRoute           ]
[direct://start                                                                  ] [     20002]
[myRoute           ] [to1               ] [mock:
before                                                                        ] [         4]
[myRoute           ] [myAsync           ] [async:bye:
camel                                                                      ] [        30]
[myRoute           ] [to2               ] [mock:
after                                                                         ] [         0]
[myRoute           ] [process1          ] [org.apache.camel.processor.async.
AsyncProcessorAwaitManagerTest$2$1@2e21712e  ] [     19967]
Exchange
--------------------------------------------------------------------------------------------------------------
----------------------
Exchange[
        Id                  ID-davsclaus-air-49173-1419147462911-0-2
        ExchangePattern     InOut
        Headers             {breadcrumbId=ID-davsclaus-air-49173-1419147462911-0-1}
        BodyType            String
        Body                Bye Camel
]
```

From the logs we can see there is one blocked thread with the id=1 and name=main. The thread was blocked in the route with name myRoute, and at the node named myAsyc, and that the thread has been blocked for almost 20 seconds. Further below the associated exchange details is logged, first is the message history which outputs the current up to date routing plan of the message. We can see that the message is currently at process1 (which is an inlined processor from the unit test). Pay attention now, from the message history we can see that two notes before we have myAsync node which was where the handover took place, and the callee threads is blocked at. So we can see from the message history that the message was further processed by to2, and that it got stuck at process1, for about 20 seconds. What happens is that the blocked thread will be interrupted so it can continue to route and terminate the thread. An exception is set on the exchange with a RejectedExecutionException exception causing the exchange to fail as well.

Java API

The manager is defined using the interface `org.apache.camel.spi.AsyncProcessorAwaitManager`

Statistics

The manager can capture utilization statistics, which can be enabled using Java API or JMX.

```
context.getAsyncProcessorAwaitManager().getStatistics().setStatisticsEnabled(true);
```

JMX managed

The AsyncProcessorAwaitManager is JMX aware as well so you can manage it from a JMX console.

See Also

- Graceful Shutdown
- Asynchronous routing engine
- User Guide