Restful asynchronous Service using JAX-RS

Tools:

Eclipse – Luna

Tomcat version 8.0

Maven

Created a maven project with artifact id of jersey-quickstart-webapp

In web.xml we need to set <async-supported>true</async-suported> in the servlet section.

Rest Service:

In the rest service, AsyncResponse is injected by JAX-RS using @Suspended annotation to provide asynchronous server side processing. URI information injected into the service using @Context annotation. newCachedThreadPool method creates thread pool that reuses a fixed number of threads operating off a shared unbounded queue. AsyncResponse registers callback to receive asynchronous response. Client request to the asynchronous service submits runnable task to the ExecutorService and returns Future object representing that task.

Client:

Client calls the async service endpoint and gets Future object. The Future represents the result of an asynchronous computation. The client request call implements InvocationCallback interface, which contains completed and failed methods. On successful completion of the asynch service call, the completed method of the InvocationCallback gets called.

In this app, to simulate asynch request to the service, client send four async requests to the server. The getAsyncData method gets image name and time that thread will sleep. The veryExpensiveOperation method uses this time to sleep.

Thread one sleep for 400 ms

Thread two sleeps for 300 ms

Thread three sleeps for 200 ms

Thread four sleeps for 100 ms

From the output it is evident that thread four returns the response first.

