package com.twitter.ann.service.query\_server.common

import com.google.inject.Module

import com.twitter.ann.common.thriftscala.AnnQueryService

import com.twitter.app.Flag

import com.twitter.finatra.decider.modules.DeciderModule

import com.twitter.finatra.thrift.ThriftServer

import com.twitter.finatra.mtls.thriftmux.Mtls

import com.twitter.finatra.mtls.thriftmux.modules.MtlsThriftWebFormsModule

import com.twitter.finatra.thrift.filters.{

AccessLoggingFilter,

LoggingMDCFilter,

StatsFilter,

ThriftMDCFilter,

TraceIdMDCFilter

}

import com.twitter.finatra.thrift.routing.ThriftRouter

/\*\*

\* This class provides most of the configuration needed for logging, stats, deciders etc.

\*/

abstract class BaseQueryIndexServer extends ThriftServer with Mtls {

protected val environment: Flag[String] = flag[String]("environment", "service environment")

/\*\*

\* Override with method to provide more module to guice.

\*/

protected def additionalModules: Seq[Module]

/\*\*

\* Override this method to add the controller to the thrift router. BaseQueryIndexServer takes

\* care of most of the other configuration for you.

\* @param router

\*/

protected def addController(router: ThriftRouter): Unit

override protected final lazy val modules: Seq[Module] = Seq(

DeciderModule,

new MtlsThriftWebFormsModule[AnnQueryService.MethodPerEndpoint](this)

) ++ additionalModules

override protected final def configureThrift(router: ThriftRouter): Unit = {

router

.filter[LoggingMDCFilter]

.filter[TraceIdMDCFilter]

.filter[ThriftMDCFilter]

.filter[AccessLoggingFilter]

.filter[StatsFilter]

addController(router)

}

}