package com.twitter.frigate.pushservice.module

import com.twitter.app.Flag

import com.twitter.inject.TwitterModule

import com.twitter.util.Duration

import com.twitter.conversions.DurationOps.\_

object FlagName {

final val shardId = "service.shard"

final val numShards = "service.num\_shards"

final val nackWarmupDuration = "service.nackWarmupDuration"

final val isInMemCacheOff = "service.isInMemCacheOff"

}

object FlagModule extends TwitterModule {

val shardId: Flag[Int] = flag[Int](

name = FlagName.shardId,

help = "Service shard id"

)

val numShards: Flag[Int] = flag[Int](

name = FlagName.numShards,

help = "Number of shards"

)

val mrLoggerIsTraceAll: Flag[Boolean] = flag[Boolean](

name = "service.isTraceAll",

help = "atraceflag",

default = false

)

val mrLoggerNthLog: Flag[Boolean] = flag[Boolean](

name = "service.nthLog",

help = "nthlog",

default = false

)

val inMemCacheOff: Flag[Boolean] = flag[Boolean](

name = FlagName.isInMemCacheOff,

help = "is inMemCache Off (currently only applies for user\_health\_model\_score\_store\_cache)",

default = false

)

val mrLoggerNthVal: Flag[Long] = flag[Long](

name = "service.nthVal",

help = "nthlogval",

default = 0,

)

val nackWarmupDuration: Flag[Duration] = flag[Duration](

name = FlagName.nackWarmupDuration,

help = "duration to nack at startup",

default = 0.seconds

)

}