package com.twitter.timelineranker.parameters

import com.twitter.finagle.stats.StatsReceiver

import com.twitter.servo.decider.DeciderGateBuilder

import com.twitter.timelineranker.parameters.entity\_tweets.EntityTweetsProduction

import com.twitter.timelineranker.parameters.recap.RecapProduction

import com.twitter.timelineranker.parameters.recap\_author.RecapAuthorProduction

import com.twitter.timelineranker.parameters.recap\_hydration.RecapHydrationProduction

import com.twitter.timelineranker.parameters.in\_network\_tweets.InNetworkTweetProduction

import com.twitter.timelineranker.parameters.revchron.ReverseChronProduction

import com.twitter.timelineranker.parameters.uteg\_liked\_by\_tweets.UtegLikedByTweetsProduction

import com.twitter.timelineranker.parameters.monitoring.MonitoringProduction

import com.twitter.timelines.configapi.CompositeConfig

import com.twitter.timelines.configapi.Config

/\*\*

\* Builds global composite config containing prioritized "layers" of parameter overrides

\* based on whitelists, experiments, and deciders. Generated config can be used in tests with

\* mocked decider and whitelist.

\*/

class ConfigBuilder(deciderGateBuilder: DeciderGateBuilder, statsReceiver: StatsReceiver) {

/\*\*

\* Production config which includes all configs which contribute to production behavior. At

\* minimum, it should include all configs containing decider-based param overrides.

\*

\* It is important that the production config include all production param overrides as it is

\* used to build holdback experiment configs; If the production config doesn't include all param

\* overrides supporting production behavior then holdback experiment "production" buckets will

\* not reflect production behavior.

\*/

val prodConfig: Config = new CompositeConfig(

Seq(

new RecapProduction(deciderGateBuilder, statsReceiver).config,

new InNetworkTweetProduction(deciderGateBuilder).config,

new ReverseChronProduction(deciderGateBuilder).config,

new EntityTweetsProduction(deciderGateBuilder).config,

new RecapAuthorProduction(deciderGateBuilder).config,

new RecapHydrationProduction(deciderGateBuilder).config,

new UtegLikedByTweetsProduction(deciderGateBuilder).config,

MonitoringProduction.config

),

"prodConfig"

)

val whitelistConfig: Config = new CompositeConfig(

Seq(

// No whitelists configured at present.

),

"whitelistConfig"

)

val rootConfig: Config = new CompositeConfig(

Seq(

whitelistConfig,

prodConfig

),

"rootConfig"

)

}