package com.twitter.search.earlybird.partition;

import java.io.Closeable;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

import com.twitter.common.util.Clock;

import com.twitter.search.earlybird.EarlybirdServer;

import com.twitter.search.earlybird.EarlybirdStatus;

import com.twitter.search.earlybird.exception.EarlybirdStartupException;

import com.twitter.search.earlybird.thrift.EarlybirdStatusCode;

/\*\*

\* Handles starting and indexing data for a partition, using a PartitionManager.

\*/

public class PartitionManagerStartup implements EarlybirdStartup {

private static final Logger LOG = LoggerFactory.getLogger(EarlybirdServer.class);

private final Clock clock;

private final PartitionManager partitionManager;

public PartitionManagerStartup(

Clock clock,

PartitionManager partitionManager

) {

this.clock = clock;

this.partitionManager = partitionManager;

}

@Override

public Closeable start() throws EarlybirdStartupException {

partitionManager.schedule();

int count = 0;

while (EarlybirdStatus.getStatusCode() != EarlybirdStatusCode.CURRENT) {

if (EarlybirdStatus.getStatusCode() == EarlybirdStatusCode.STOPPING) {

return partitionManager;

}

try {

clock.waitFor(1000);

} catch (InterruptedException e) {

LOG.info("Sleep interrupted, quitting earlybird");

throw new EarlybirdStartupException("Sleep interrupted");

}

// Log every 120 seconds.

if (count++ % 120 == 0) {

LOG.info("Thrift port closed until Earlybird, both indexing and query cache, is current");

}

}

return partitionManager;

}

}