Tokenizer Distributed Executor – System Administrator Guide

Contents

[Overview 2](#_Toc385252729)

[System Requirements 2](#_Toc385252730)

[Persistence 3](#_Toc385252731)

[Overview 3](#_Toc385252732)

[Hardware Requirements 3](#_Toc385252733)

[Indexing 4](#_Toc385252734)

[Overview 4](#_Toc385252735)

[Hardware Requirements 4](#_Toc385252736)

[Orchestrator 4](#_Toc385252737)

[TDE Agents 4](#_Toc385252738)

[Installation and Configuration 4](#_Toc385252739)

[Apache Cassandra 4](#_Toc385252740)

[Apache Solr 4](#_Toc385252741)

[Apache ZooKeeper 5](#_Toc385252742)

[TDE Agents 5](#_Toc385252743)

[Admin Console 5](#_Toc385252744)

# Overview

Tokenizer Distributed Executor (or simply TDE) is an application platform and development framework to run Big Data solutions in a clustered multicomputing environment.

The main feature of TDE which makes it unique and different from other methodology such as MapReduce is real time configuration changes, start/stop, and implementation of fail-over singleton and multiton execution threads in a cluster of N computing nodes.

Practical solutions (also called “Distributed Tasks”) provided:

* + Classic Internet Robot crawling specified domain
  + Sitemaps-Based Internet Robot crawling specified domain
  + HTML Parser retrieving predefined XPath elements and indexing it in SOLR
  + HTML Parser retrieving URLs with XPath defined patterns, such as links to user review pages
  + BLOGs crawler crawling BLOG pages supporting Moreover Weblogs protocol and similar, <http://weblogs.com>
  + RSS feeds subscriber to retrieve journal articles, news, blog posts, and everything published as RSS in a sources such as Toronto Star, Blogspot, Yahoo
  + Twitter Subscriber to retrieve as a stream of data up to 400 preconfigured keywords

Each of these can be singleton, such as Classic Robot which (in order to be polite robot) should be run as a single thread in a cluster, or multiton, such as BLOGs retrieval M-threads evenly distributed in a cluster of N nodes. As a sample, Classic Robot can be configured to run 20 threads in a cluster of 5 nodes, for friendly sites only (other sites can ban for such behavior).

# System Requirements

Currently, TDE is powered by Oracle Java 7 and run on top of

* Apache ZooKeeper: to store tasks configuration and accumulated statistics
* Apache Cassandra: to store raw data
* Apache SOLR: to index parsed data
* Apache Tomcat: to run Sys Admin UI
* TDE Agents: to run TDE multi-node distributed tasks such as web crawlers, parsers, indexers

Although it can be powered by a single node only, single-node solution is not fault tolerant nor scalable.

For production environments, recommended **absolute minimum** with future scalability setup:

* 3 low-range computers to power ZooKeeper: 2-core CPU, 64Gb HDD, 4Gb RAM
* 3 middle-range computers to power Apache Cassandra (cluster of 3 with replication factor 3) + TDE Agents; 8-core CPU; 1Tb HDD; 16Gb RAM – *this cluster will grow in the future*
* 4 high-range computers to power SOLR: Master-Slave for huge amount of historical data with statistical analysis + Master-Slave for smaller amount of today’s data (to be able to do real-time statistics): 16-core CPU, 256Mb SDD, 32Mb RAM
* 1 low-range computer to power Tomcat based Admin Console: 2-core CPU, 64Mb HDD, 4Gb RAM

## Persistence

### Overview

Currently, TDE supports only Apache Cassandra column-oriented clustered “big data” or “NoSQL” database. Future releases will support additionally HBase, MongoDB, and more.

Cassandra is highly scalable, transactionally supporting simplified hash-based secondary indices, supporting replications of data blocks in N (typically 3) nodes in a cluster, fault tolerant, decentralized, highly performant, and elastic, the most suitable for current real-time “vertical search and indexing” needs.

### Hardware Requirements

Hardware: prefer dedicated hardware; do not use virtual environments

Memory: optimal price-performance sweet spot is 16GB to 64GB

CPU: 8-core CPU processors are the current price-performance sweet spot

HDD: 1Tb – 3Tb per node. No need for any RAID. Performance will degrade if capacity is higher than 5Tb per node.

SSD: highly recommended additionally to main HDD, for commit log files and SSTables.

Details:

<http://wiki.apache.org/cassandra/CassandraHardware>

<http://www.datastax.com/documentation/cassandra/2.0/cassandra/architecture/architecturePlanningHardware_c.html>

## Indexing

### Overview

Indexing service is needed for full text search with statistical analysis. It can calculate in real time search results distribution by age, gender, category, feature, location, sentiment, and more. It can retrieve “Toyota” if we search for a “car”, and it can retrieve “glasses” if we search for “spectacles”, and more.

We are currently using Apache SOLR.

<https://lucene.apache.org/solr/>

### Hardware Requirements

For almost real time statistical analysis of one month data from Twitter, two clusters required:

1. First cluster will update and optimize large indexes once a day at 3:00AM
2. Second cluster will update small indexes for the current day only (after 3:00AM) almost in real time

Each cluster will be configured as a master-slave for fail over. All updates will go to “slave”, so that technically “slave” does not need a lot of computing power. All statistics calculations and end-user queries will be done on a “master” which requires more RAM and CPU.

## Orchestrator

This is the role of ZooKeeper which currently manages TDE Agents configuration and monitoring tasks. It is required to have 3 tiny nodes with 2Gb RAM each. Data is replicated on each node. It must be high-availability storage for cluster maintenance tasks, that’s why we need dedicated nodes. Future HBase and Solr Cloud can reuse it.

## TDE Agents

Those will pick configuration from ZooKeeper and run tasks such as Web Crawl, HTML Parser, and etc.

It will run as a standalone Java application; it will share the same nodes as Cassandra.

# Installation and Configuration

## Apache Cassandra

Custom configuration settings are provided as a set of configuration files available for download from GIT repository. In general, follow the technical documentation at Apache Cassandra website.

<http://cassandra.apache.org/>

## Apache Solr

Custom schema, replication, and configuration settings are provided as a set of configuration files available for download from GIT repository. In general, follow the technical documentation at Apache Lucene.

<https://lucene.apache.org/solr/features.html>

## Apache ZooKeeper

Follow the technical documentation at Apache ZooKeeper website. We will document here any special configuration settings. Right now, there are no any special requirements.

<http://zookeeper.apache.org/>

## TDE Agents

The most up-to-date configuration, startup script, and project build files are available at GIT repository. We will document here only most general requirements.

## Admin Console

The most up-to-date configuration and project build files are available at GIT repository. We will document here only most general requirements. Tomcat 7 is currently required, with tiny MySQL database to support user authentication and authorization.