Clustering Guide

Overview

Yellowfin can be clustered on multiple servers to allow for high-availability and load-balancing. The function of load-balancing (multiplexing requests from external requests) can be achieved with a hardware load-balancer, or load-balancing software. This guide will outline the modifications required to Yellowfin's configuration to enable clustering, but not the external environment that directs incoming requests to particular nodes.

It is required that the load-balancing infrastructure delivers network traffic to the Yellowfin application server transparently - as if the packets were connecting directly to the application server. If Session Replication is not enabled, the load-balancing infrastructure will need to provide "sticky-session" functionality. This is where traffic for a user's session is sent to the same node for the duration of their session.

Clustered Environment Components

There are two components of a clustered environment. Application Messaging, which is required for the cluster to operate correctly, and Session Replication, which is optional.

Application Level Messaging

A Yellowfin node will communicate with other clustered nodes when the application needs to make changes that affect the entire cluster. This is usually to maintain the integrity of caches on remote nodes, but also facilitates functions such as user synchronization and license management. Application Level Messaging is configured by the ClusterManagement servlet in the web.xml file.

Click here to learn more about this configuration.

Container Level Session Replication

The Yellowfin application has also been written to allow user session information to be replicated onto multiple cluster nodes concurrently. User session information is the memory footprint that is stored on the server for each user session. Enabling Session Replication allows for the user's session to continue, even if the cluster node they were connected to experiences a failure.

Without Session Replication, failure of a cluster node will destroy a user's session, and they will need to log in again to another node. Session Replication is a function of the Java Application Server, or other external infrastructure. An example of how this can be achieved with Tomcat will be discussed in this guide.

Click here to learn more about this configuration.

Yellowfin Database Clustering

A Yellowfin Cluster must share a single Yellowfin Database instance. The database can be a single database instance, that is shared across all Yellowfin nodes, or a database that is clustered itself. It is important that the Yellowfin database is scaled to handle database requests from all the nodes in the Cluster.

Database Clustering / Replication should be transparent to Yellowfin, as Yellowfin will have a single database URL for the entire cluster. Each node should be connecting to the same logical database, irrespective of how it is configured or clustered.

Licensing

The Yellowfin license file is stored in the Yellowfin database, and because of this, the license must contain a reference to all hostnames in the cluster. A clustered license can be requested from Yellowfin.

Guide Topics

The following topics will help set up a Yellowfin cluster.

- Yellowfin cluster installation
- Application level messaging
- Background tasks
- Container level session replication

Next topic: Cluster installation