

Recommended Environments for Clearspace Deployments

Clearspace runs best on recommended hardware and environmental configurations. Use this document as a starting place for estimating what you'll need to support your deployment.

Note: Also, note that at this time Clearspace is not supported in a virtualized environment. A virtualized environment can slow the application's performance by severely decreasing disk I/O and network I/O under load.

General Requirements

Internal Deployments (Clearspace)

For internal deployments, expect a higher ratio of actions that write to the database. This includes creating new content, adding comments and replies, updating content, and so on. Here, writes are estimated as 15 percent of the number of page views.

Page Views per Month	Database Writes per Month	CPUs	RAM	Location of App Server and Database	Clustered?
500,000	75,000	1 dual core CPU (at least 2 GHz)	2GB, 1GB Java heap	Share a box	No
1,000,000	150,000	1 dual core CPU (at least 2 GHz)	2GB, 1GB Java heap	Separate boxes	No
5,000,000	750,000	1 dual core CPU (at least 2 GHz)	4GB, 2GB Java heap	Separate boxes	Yes
10,000,000	1,500,000	1 dual core CPUs (at least 2 GHz)	4GB, 2GB Java heap	Separate boxes	Yes
25,000,000	3,750,000	2 dual core CPUs (at least 2 GHz)	4GB, 2GB Java heap	Separate boxes. Database box is a quad core.	Yes

External Deployments (Clearspace Community)

In general, and external deployment is going to have a very high ratio of page views relative to database writes. That's because external deployments tend to support communities with a large number of people who rarely contribute but often check back to read what others are posting.

Page Views per Month	Content Created per Month	CPUs	RAM	Location of App Server and Database	Clustered?
500,000	20,000	1 dual core CPU (at least 2 GHz)	2GB, 1GB Java heap	Share a box	No
1,000,000	20,000	1 dual core CPU (at least 2 GHz)	2GB, 1GB Java heap	Separate boxes	No
5,000,000	50,000	1 dual core CPU (at least 2 GHz)	4GB, 2GB Java heap	Separate boxes	Yes
10,000,000	50,000	1 dual core CPUs (at least 2 GHz)	4GB, 2GB Java heap	Separate boxes	Yes
25,000,000	100,000	2 dual core CPUs (at least 2 GHz)	4GB, 2GB Java heap	Separate boxes/ DB box is a quad core	Yes

Example JVM Settings

It's important to use the latest JVM (Java Virtual Machine) for your platform as well as the correct GC (garbage collection) settings. Clearspace requires at least a 1.5 VM; 1.6 is recommended.

Heap	Startup Parameters
1 GB	-Xms512m -Xmx1024m -XX:MaxPermSize=256m -XX:+UseParNewGC -XX:+UseConcMarkSweepGC
2 GB	-Xms512m -Xmx2048m -XX:MaxPermSize=256m -XX:+UseParNewGC -XX:+UseConcMarkSweepGC

In both cases this will allocate a 512 MB heap at minimum as well as allocate 256 MB to the "PermGen" space of the VM heap. Note that you might want to increase the MaxPermGen size to 512 MB if any of the following are true for your instance:

- Web services are enabled on application startup.
- You have more than 100 unique users logging in and creating content/comment every day.
- You have more than 5000 artifacts in the system.

Finally, we configure the "Concurrent Mark Sweep" GC algorithm. We've found this to be the optimal GC setting for deployments on multiple core processor configurations.

If you want to debug the contents of your VM, use the following extra startup parameters:

Debug Parameters
-verbose:gc -XX:+PrintGC -XX:+PrintGCDetails -XX:+PrintGCTimeStamps -Xloggc:/path/to/log/file/here.log

Please refer to your application server's documentation for information on setting these startup parameters.

Time Configuration

For clustered configurations, it is absolutely essential that all cluster nodes have system time with as little drift as possible. This is generally adjusted on a regular interval by syncing with an NTP server inside the firewall. For servers under higher load, it may be necessary to increase the clock sync interval as loaded servers will be more likely to experience clock drift.

Network Topology

It's important to keep your network as simple as possible to minimize environmental interference. For example, a bad network architecture would contain multiple firewalls, proxies and network hops between your users, the application server and the database. Keep the following things in mind:

- **A proxy in front of the application server is not required, but can be useful in some cases.** For example, a web proxy can show a nice "site is down" message if you restart your application server and it can assist with network port translations. If you can run your application without a web proxy then that is recommended.
- **Keeping the boxes on the same network is important.** You can have too much latency in your environment if your application servers and/or database live on different physical networks. It's best to configure those machines on the same network.
- **Evaluate the quality of switches and other network gear in and around your deployment.** Bad switches will drop packets and reduce the overall quality of your network. Further, you must use gigabit Ethernet and switches.

Example Hardware

- Dell PowerEdge 2950 2U Server
 - 2 2.33 GHz Dual Core Xeon CPUs
 - 4 GB RAM
 - Linux OS
 - RAID 5 controller with 3 hard drives