

# Fremantle Beta Release Notes

## Release Notes for Fremantle Beta 2

### Overview

Fremantle is the code name for the next major release of Enterprise Ehcache, Quartz Scheduler and the Terracotta Enterprise Suite:

- Enterprise Ehcache 2.4.0
- Quartz Scheduler 2.0.0
- Terracotta Enterprise Suite 3.5.0

### Summary of Changes

#### Enterprise Ehcache

- Addition of Ehcache Search (see below)
- Improvements to Enterprise Ehcache Disk Store (for use in conjunction with BigMemory)
- New local and local\_jta Ehcache transaction modes
- NonStopCache is now in core
- Explicit locking is now in core
- Ehcache rejoin. Distributed Ehcache L1's are now about to rejoin the cluster on disconnect/reconnect from the Terracotta Server Array
- Ehcache Size method performance improvement for distributed caches

#### Quartz Scheduler

- New fluent configuration API (see below)
- Quartz Where - permitting control of where jobs execute in a clustered environment through node groups and node type constraints  
A getting-started page for using Quartz Scheduler Locality API is available: [Getting Started With Quartz Scheduler Locality API](#)

### About Ehcache Search

The Ehcache Search API allows you to execute arbitrarily complex queries against Ehcache. The standalone Ehcache Search implementation is limited to the memoryStore (ie it has no support for diskStore or BigMemory - aka offHeapStore). Ehcache Search support for distributed caches is an Enterprise-only feature, and is designed to work across all Terracotta server storage tiers (memory, BigMemory, and disk).

The easiest way to get started with the API is to try the sample application posted on github. This includes a ready to run bundle for \*nix (just download, unpack, and execute with 'sh run.sh'). This bundle also includes a maven pom to rebuild (using the latest SNAPSHOT rather than that in the bundle): <https://github.com/sharrissf/Ehcache-Search-Sample/downloads>

Ehcache Search docs are here: <http://ehcache.org/documentation/search.html>

A getting-started page for using Ehcache Search with clustered caches is also available: [Getting Started With Ehcache Search API For Clustered Caches](#)

### New configuration options

NonStopCache is enabled by default, to disable non-stop, update the <terraccotta> element in each <cache>

```
<terraccotta clustered="true">
  <nonstop enabled="false"/>
</terraccotta>
```

Ehcache rejoin on disconnect from Terracotta Server Array is enabled by default and can be disabled as follows in ehcache.xml. Note: that rejoin enable is dependent on NonStopCache being enabled.

```
<terraccottaConfig rejoin="false" url="localhost:9510" />
```

Ehcache now has three transaction modes:

- **local**, which will use a local transaction manager (rather than a JTA one), and permit users to create ACID operations across multiple caches

```
transactionalMode="local"
```

- **xa**, which in Beta1 was temporarily called `jta_local` will use a JTA transaction manager, but will be designed for use with a single resource manager in mind and to work with a lower level of recoverability than the "xa\_strict" mode. (it won't recover transaction in the case of catastrophic failure).

```
transactionalMode="xa"
```

The original "xa" mode is now "xa\_strict"

Local is the fastest mode, followed by xa, followed by xa\_strict

A sample that demonstrates the new transaction modes is available here: <https://github.com/sharrisf/Ehcache-Local-Transaction-Sample/downloads>

## About Quartz Scheduler 2.0

Please see the [What's new in Quartz 2.0](#) document

Note: [Quartz Where](#) is only supported when using the TerracottaJobStore, and is an enterprise-only feature.

## Expected Updates and Scheduled Bug Fixes

[CDV](#) Project Jira fixes

[EHC](#) Project scheduled fixes for Ehcache 2.4.0

## Beta Notes

General

- APIs (Ehcache Search, Quartz Scheduler 2.0 etc) are subject to change
- The implementation is not performance tuned

### New In Beta2

- Monitoring of Search, feature via JMX
- Search Attribute.ne() functional
- Ehcache NonStopCache is now in the Terracotta core kit, and in Beta2 is on by default - but does not interact well at Beta time with some existing Ehcache features. See Known Issues and Limitations
- Ehcache rejoin is now configurable, it is on by default in BETA2 but dependent on NonStopCache
- Ehcache Write Behind updates
- Server-side (L2) BigMemory enhancement (eg better defaults, removal of chunk/segment size restrictions, auto-tuning of chunk and segment sizes, better protection from OOMs, ...)
- Quartz Locality (ehcacheConstraint) -
- Configuration based locality is only supported on Trigger groups only right now (not on Job groups), wildcards and matchers

### Known Issues and Limitations

- We are aware of issues with NonStopCache and Rejoin interaction with Hibernate, BlockingCache and Explicit Locking
  - Workaround: Disable NonStopCache and Rejoin as described above under "New Configuration Options"
- FORGE-594 Quartz avg load monitoring only on JDK6 (JDK5 will require Sigar afaict)
- QTZ-110 Ehcache integration to "Where" only supports String keyed Caches
- (writebehind) maxQueueSize is not implemented
- (writebehind) maxQueueSize isn't configurable using the ehcache.xml and is only functional with Terracotta, not with the standalone ehcache writebehind.
- Server side eviction does not update search index causing incorrect search results
- New transactional caches and search do not work together. xa\_strict should still be OK.

### Specific items which are planned for the final release, but are not included in Beta

- Monitoring of Write Behind, JTA
- Ehcache Search is supported with Terracotta clustered caches, however in the Beta there are a series of known limitations with the implementation:
  - Active/Passive server failover will not be supported
  - Attribute.like() is case sensitive for clustered search (contrary to javadoc in Ehcache core) -
  - There is no timeout for search operations
  - For numeric search attributes with null values the sort order will differ between standalone and clustered
  - Paged/chunked results: a large search results will be transferred in whole to node making the query. This may cause OOME
- Ehcache Search (standalone and clustered): executeAndRemove, cache metadata search (TBD) - *NOT SCHEDULED*
- Developer Console updates to support Quartz 2.0
- Ehcache Terracotta clustering - new default mode - *TBD in Jan not in BETA2*
- Updated Extjs toolkit in Ehcache Monitor
- Usability improvements to Ehcache Monitor

Please email any questions you have regarding the beta to pm <at> terracotta.org