EhCache

- Description
- Cache Topology Types
- Storage Tiers
- Key classes
- Example

Description

Ehcache is a widely used open source Java distributed cache for general purpose caching.

It features

- memory and disk stores,
- replicate by copy and invalidate,
- listeners,
- cache loaders,
- cache extensions,
- cache exception handlers,
- a gzip caching servlet filter,
- RESTful and SOAP APIs.

Ehcache can be used as a library and as stand-alone application cache server.

Cache Topology Types

Standalone

The data set is held in the application node. Any other application nodes are independent with no communication between them.

Distributed

The data is held in a remote server (or array of servers) with a subset of recently used data held in each application node. This topology offers a rich set of consistency options.

Replicated

The cached data set is held in each application node and data is copied or invalidated across the nodes without locking. Replication can be either asynchronous or synchronous, where the writing thread blocks while propagation occurs.

Storage Tiers

You can divide a cache or in-memory data set across the following storage areas, referred to as tiers:

MemoryStore

On-heap memory used to hold cache elements. This tier is subject to Java garbage collection.

OffHeapStore

Off-heap memory used to hold cache elements. With no GC.

DiskStore

Backs up in-memory cache elements and provides overflow capacity to the other tiers.

Key classes

CacheManager

The CacheManager class is used to manage caches. Creation of, access to, and removal of caches is controlled by a named CacheManager.

Cache

A Cache is a thread-safe logical representation of a set of data elements.

Element

An element is an atomic entry in a cache. It has a key, a value, and a record of accesses. Elements are put into and removed from caches. They can also expire and be removed by the cache, depending on the cache settings.

CacheConfiguration

A value object used to represent cache configuration.

Contains information about:

- Time to idle
- Time to live
- Is eternal
- Eviction policy algorithm
- Max entries in Local Heap

Logs

Ehcache uses log4j and slf4j – java logging frameworks.

To initialize them you need:

```
<dependency>
    <groupId>org.slf4j</groupId>
    <artifactId>slf4j-api</artifactId>
    <version>1.7.5</version>
</dependency>
<dependency>
    <groupId>org.slf4j</groupId>
    <artifactId>slf4j-log4j12</artifactId>
    <version>1.7.5</version>
</dependency>
```

And

org.apache.log4j.BasicConfigurator.configure();

for default configuration in the method main().