

# MyBatis Memcached integration - Reference Documentation

Memcached (<http://memcached.org/>) is an in-memory key-value store for small chunks of arbitrary data (strings, objects) from results of database calls, API calls, or page rendering..

The Memcached integration is built on top of the `spymemcached` client, written by Dustin Sallings.

Users that want to use Memcached into their applications, have to download the zip bundle (<http://code.google.com/p/mybatis/downloads/list?q=Product-Cache>), decompress it and add the jars in the classpath; Apache Maven users instead can simply add in the `pom.xml` the following dependency:

```
<dependencies>
  ...
  <dependency>
    <groupId>org.mybatis.caches</groupId>
    <artifactId>mybatis-memcached</artifactId>
    <version>1.0.0</version>
  </dependency>
  ...
</dependencies>
```

then, just configure it in the mapper XML

```
<mapper namespace="org.acme.FooMapper">
  <cache type="org.mybatis.caches.memcached.MemcachedCache" />
  ...
</mapper>
```

The Memcached cache is configurable by putting the `/memcached.properties` classpath resource; if not found, the client will use the default setting.

The following table resumes the supported configurations params; each parameter is optional, if not found in the configuration, the client will use the default value:

Property	Default	Description
org.mybatis.caches.memcached.keyprefix	<code>_mybatis_</code>	any string identifier
org.mybatis.caches.memcached.servers	<code>localhost:11211</code>	space separated list of <code>\${host}:\${port}</code>
org.mybatis.caches.memcached.connectionfactory	<code>net.spy.memcached.DefaultConnectionFactory</code>	Any class that implements <code>net.spy.memcached.ConnectionFactory</code>
org.mybatis.caches.memcached.expiration	the number of seconds in 30 days	the expiration time (in seconds)
org.mybatis.caches.memcached.asyncget	<code>false</code>	flag to enable/disable the async get
org.mybatis.caches.memcached.timeout	<code>5</code>	the timeout when using async get
org.mybatis.caches.memcached.timeoutunit	<code>java.util.concurrent.TimeUnit.SECONDS</code>	the timeout unit when using async get
org.mybatis.caches.memcached.compression	<code>false</code>	if true, objects will be GZIP compressed before putting them to Memcached

If users need to log cache operations, they can plug the Cache logging version:

```
<mapper namespace="org.acme.FooMapper">  
  <cache type="org.mybatis.caches.memcached.LoggingMemcachedCache" />  
  ...  
</mapper>
```