

The new Ehcache 2.0.0 and Hibernate Caching SPI Provider

Presented by:

Chris Dennis, Software Engineer, Terracotta Inc.

May 12, 2010

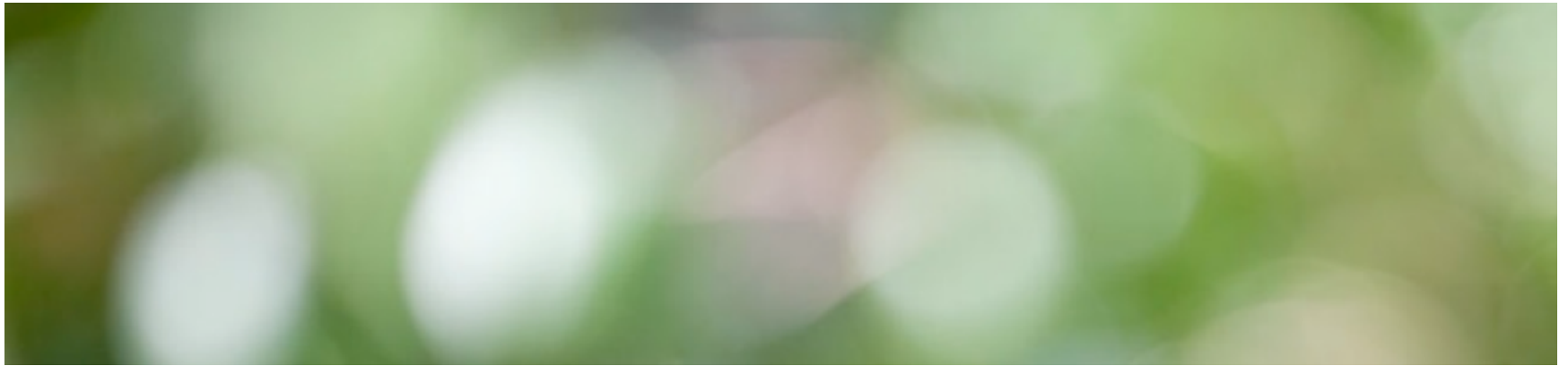
Agenda

- **Intro to Ehcache and Terracotta**
- **Ehcache and Ehcache + Terracotta**
- **Code: Scaling Spring Pet Clinic**
 - With Hibernate
 - With JDBC direct
- **Comparative Performance Testing Results**
 - Database
 - Ehcache EX
 - Memcached
 - Well-known IMDG
- **Just Released: Ehcache 2.0**

About Ehcache

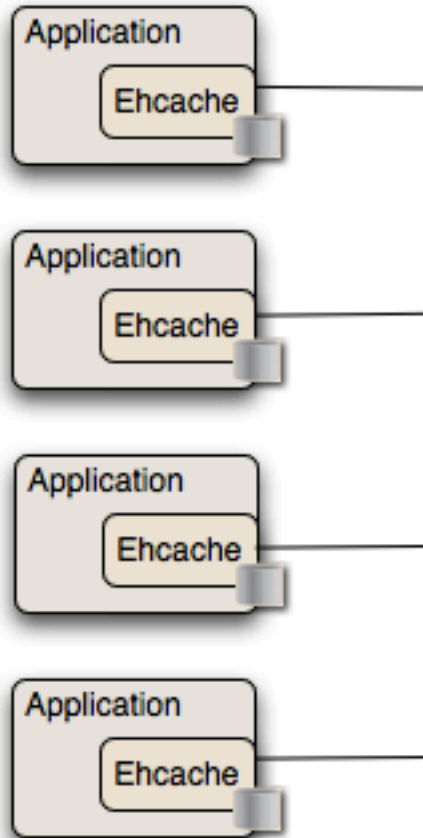
The world's most widely used Java cache

- **Founded in 2003**
- **Apache 2.0 License**
- **Integrated by lots of projects, products**
- **Hibernate Provider implemented 2003**
- **Web Caching 2004**
- **Distributed Caching 2006**
- **Greg Luck becomes co-spec lead of JSR107**
- **JCACHE (JSR107) implementation 2007**
- **REST and SOAP APIs 2008**
- **SourceForge Project of the Month March 2009**
- **Acquired by Terracotta 2009**
- **2 Releases 2009/2010: Integration with Terracotta Server Array & Perf**
- **Ehcache 2.0 March 2010**



Ehcache 2.0

Ehcache before Terracotta



RMI
JGroups
JMS

Up to 8GB in-process

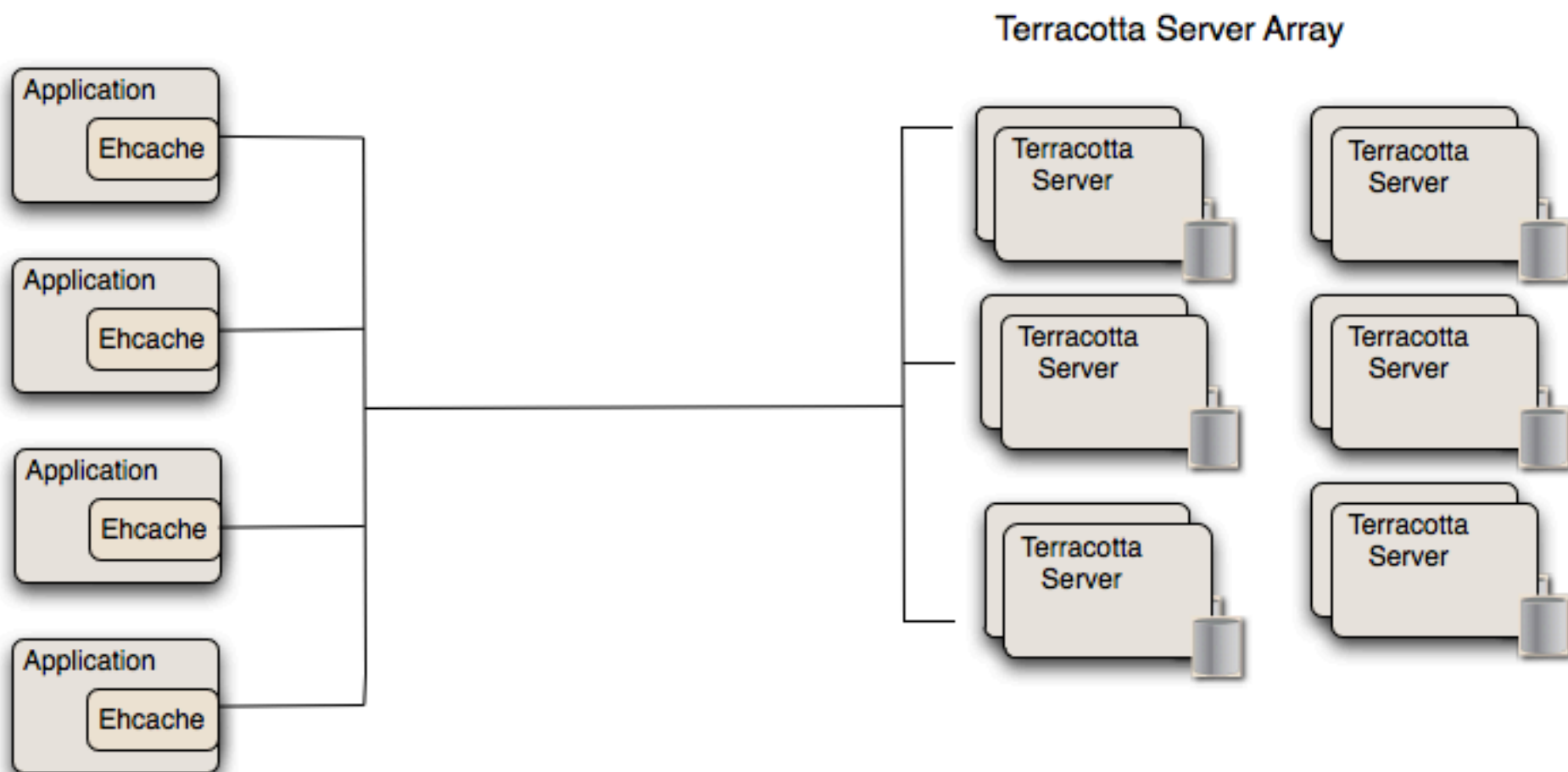
Up to 20GB on disk

Replicated Distribution up to 20 nodes

Not coherent/ transactional/HA

Ehcache after Terracotta

Simple + Performant + Coherent + HA + Scaleable



Adding a specific ehcache.xml

ehcache.xml:

```
<ehcache>
  <defaultCache maxElementsInMemory="10000"
    eternal="false"
    timeToLiveSeconds="120" />

  <cache name="com.company.domain.Pets"
    maxElementsInMemory="10000"
    eternal="true">
</cache>

  <cache name="com.company.domain.Owners"
    maxElementsInMemory="10000"
    timeToLiveSeconds="3000">
</cache>
</ehcache>
```

Adding Terracotta

ehcache.xml

```
<ehcache>
  <terracottaConfig url="someserver:9510"/>
  <defaultCache maxElementsInMemory="10000"
    eternal="false"
    timeToLiveSeconds="120"/>

  <cache name="com.company.domain.Pets"
    maxElementsInMemory="10000"
    eternal="true">
    <terracotta clustered="true" coherent="false"/>
  </cache>

  <cache name="com.company.domain.Owners"
    maxElementsInMemory="10000"
    timeToLiveSeconds="3000">
    <terracotta clustered="true" coherent="true"/>
  </cache>
</ehcache>
```

Ehcache 2.0 - New Features

■ **Hibernate 3.3+ Caching SPI**

- Old SPI was heavily synchronized and not well suited to clusters
- New SPI uses CacheRegionFactory
- Fully cluster safe with Terracotta Server Array
- Unification of the Ehcache and Terracotta 3.2 providers

■ **JTA**

- Cache as an XAResource
- Detects most common Transaction Managers
- Others configurable
- Works with Spring, EJB and manual transactions

Ehcache 2.0 - New Features

■ Write-behind

- Offloads Databases with high write workloads
- CacheWriter Interface to implement
- `cache.putWithWriter(...)` and `cache.removeWithWriter(...)`
- Write-through and Write-behind modes
- Batching, coalescing and very configurable
- Standalone with in-memory write-behind queue.
- TSA with HA, durability and distributed workload balancing

■ Bulk Loading

- incoherent mode for startup or periodic cache loading
- 10 x faster
- No change to the API (put, load etc).
- `SetCoherent()`, `isCoherent()`, `waitForCoherent()`

Ehcache 2.0 - New Features ...cont.

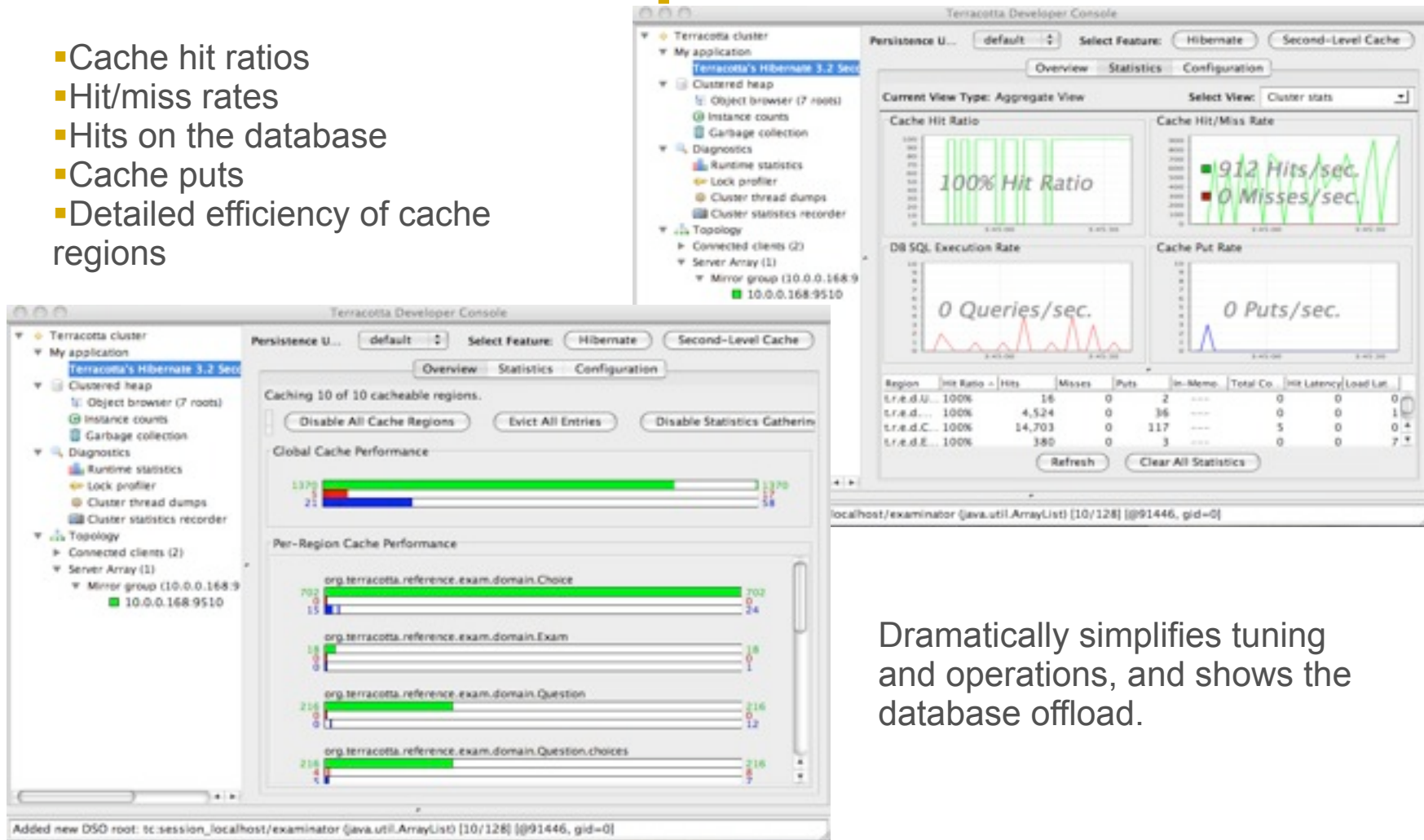
- **New CAP configurability – per cache basis**
 - coherent – run coherent or incoherent (faster)
 - synchronousWrites – true for ha, false is faster
 - copyOnRead – true to stop interactions between threads outside of the cache
 - Cluster events – notification of partition and reconnection
- **Management**
 - Dynamic Configuration of common cache configs from JMX and DevConsole
 - New web-based Monitoring with UI and API

Ehcache 2.0 Monitoring Options

- **JMX**
 - is built in to Ehcache but...
 - JMX needs use portmap
 - Slow
 - Machines may be headless
- **Terracotta Dev Console (if using Terracotta)**
- **Ehcache Console, coming soon for supported customers**

Terracotta Developer Console

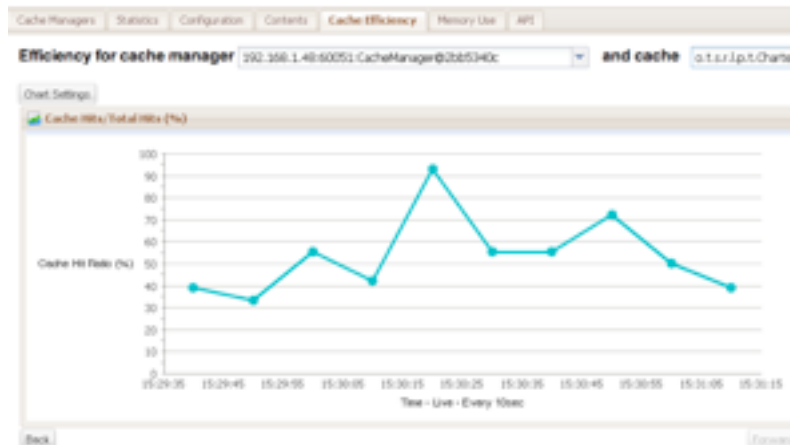
- Cache hit ratios
- Hit/miss rates
- Hits on the database
- Cache puts
- Detailed efficiency of cache regions



Dramatically simplifies tuning and operations, and shows the database offload.

Ehcache Console

- Web based
- Configuration
- Efficiency
- Memory Use



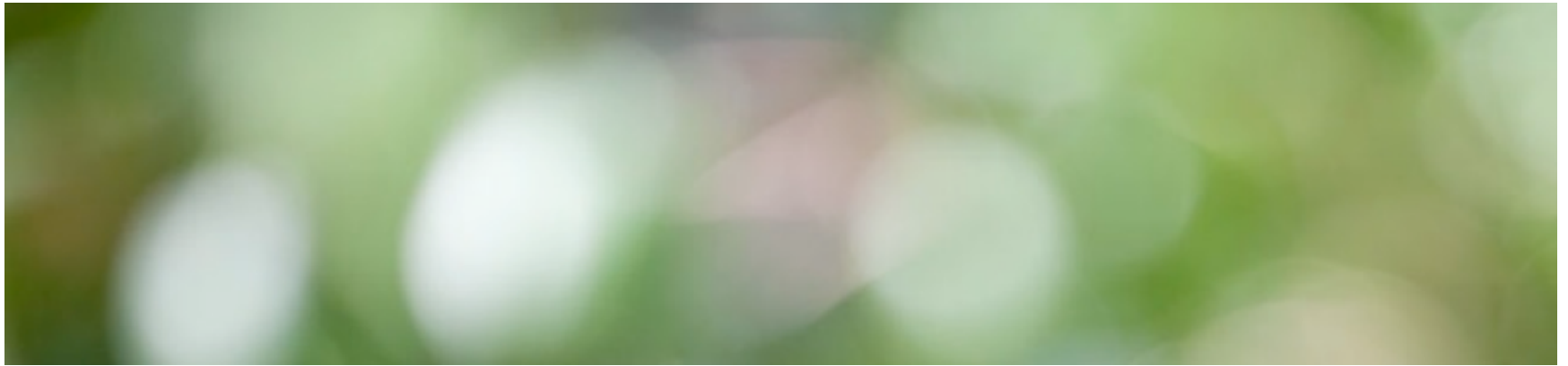
No license key selected

Cache Managers | **Statistics** | Configuration | Contents | Cache Efficiency | Memory Use | API

Statistics for cache manager: All

Cache	Cache Instan...	Total Elements	HE Ratio	HE/s	Misses/s	Put/s
Albury	1	7	0.0%	0.0	0.2	0.2
Freemantle	1	21	0.0%	0.0	0.2	0.2
Oldstone	1	19	0.0%	0.0	0.2	0.2
Logan-City	1	20	0.0%	0.0	0.2	0.2
Nedlands	1	24	0.0%	0.0	0.2	0.2
org.terracotta.some.really.long.package.name.ltd.en.Wollongong	1	19	0.0%	0.0	0.2	0.2
Newcastle	1	29	0.0%	0.0	0.3	0.3
Wollings	1	22	0.0%	0.0	0.3	0.3
org.terracotta.some.really.long.package.name.ltd.en.Perth	1	26	0.0%	0.0	0.3	0.3
Documentation Licensing & Support	1	32	0.0%	0.0	0.4	0.4
No license key selected	1	38	0.0%	0.0	0.4	0.4
client.Horsham	2	44	25.0%	0.1	0.3	0.4
client.Rockingham	1	44	0.0%	0.0	0.5	0.5
client.Rochester	1	25	0.0%	0.0	0.5	0.5
client.Generation	1	69	33.3%	0.3	0.6	0.6
client.Chertsey ...	1	38	11.1%	0.1	0.6	0.6
client.Chertsey ...	1	134	38.8%	0.7	1.1	1.5
client.Oldstone	1	35	0.0%	0.0	1.6	1.6
client.Dubbo	1	192	95.5%	6.4	0.3	4.0
client.Wollings	1	18	14.4%	1.1	0.5	7.2

- Comes with supported versions
- API to connect Operations Monitoring



Demo: Spring Pet Clinic

Code - Spring Pet Clinic

Pet Clinic A Spring Framework Demonstration

Spring

Welcome

- [Find owner](#)
- [Display all veterinarians](#)
- [Tutorial](#)
- [Documentation](#)



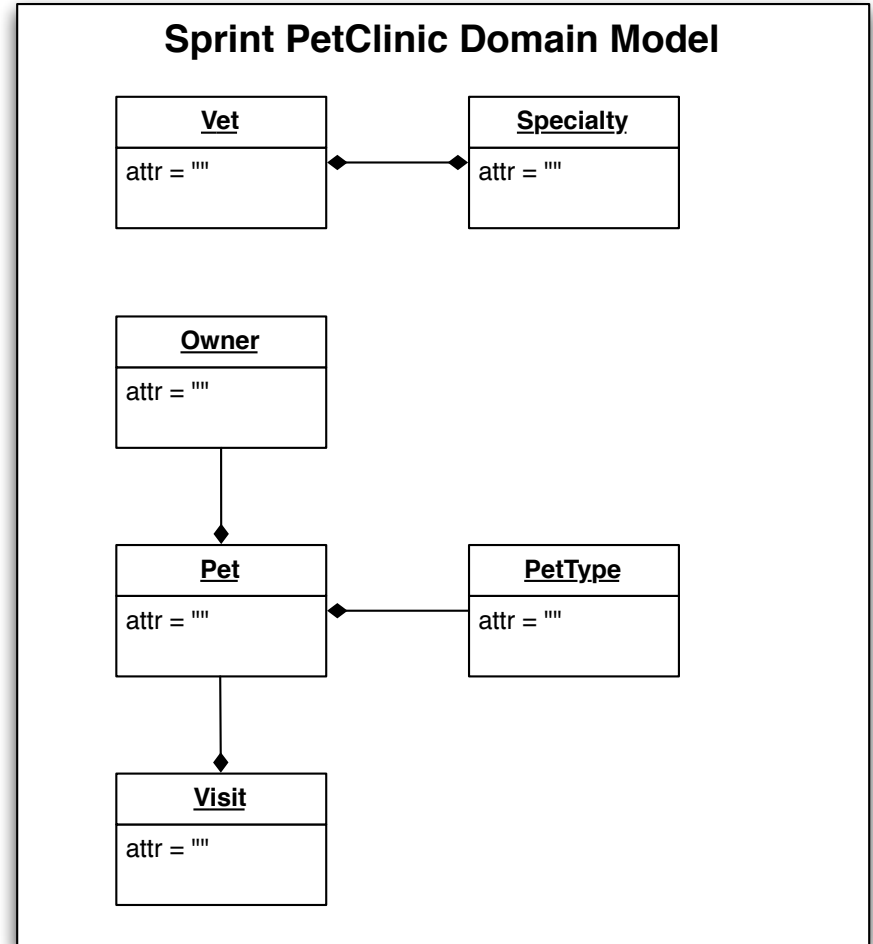
[Home](#)

Sponsored by SpringSource 

Pet Clinic Domain Model

Domain Objects

- Vets
- Specialty
- Owner
- Pet
- PetType
- Visit



Code

Steps:

- Configure PetClinic for Hibernate
- Configure hibernate for second-level cache
- Configure hbm file for caching
- Update query code to add caching

Optional but recommended:

- add ehcache.xml to WEB-INF/classes
- specify cache regions and config

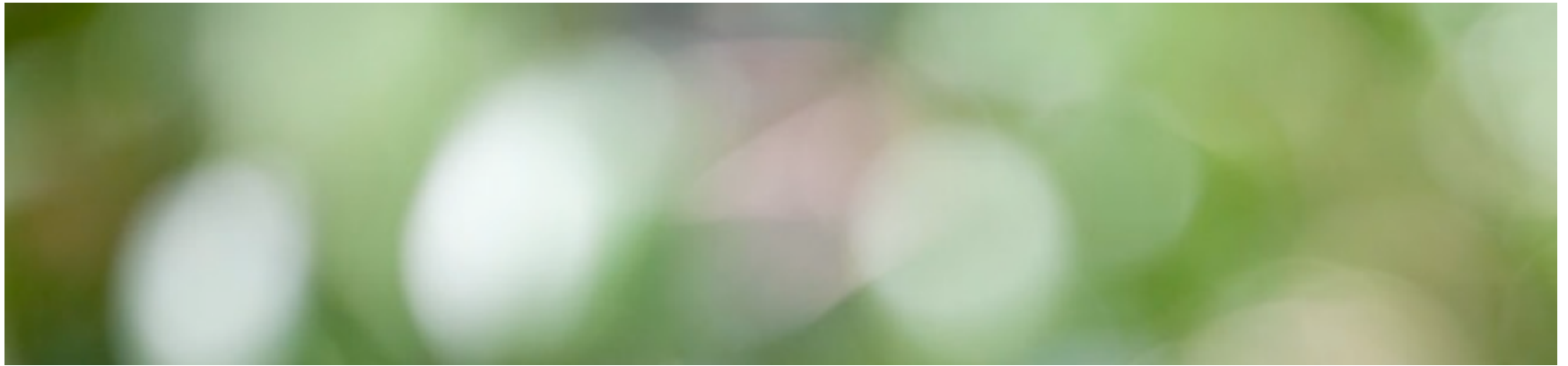
Code

ehcache.xml:

```
<ehcache>
  <defaultCache
    maxElementsInMemory="10000"
    eternal="false"
    timeToLiveSeconds="120"
  />

  <cache name="org.hibernate.cache.UpdateTimestampsCache"
    maxElementsInMemory="10000"
    timeToIdleSeconds="300"
  />

  <cache name="org.hibernate.cache.StandardQueryCache"
    maxElementsInMemory="10000"
    timeToIdleSeconds="300"
  />
</ehcache>
```



Performance

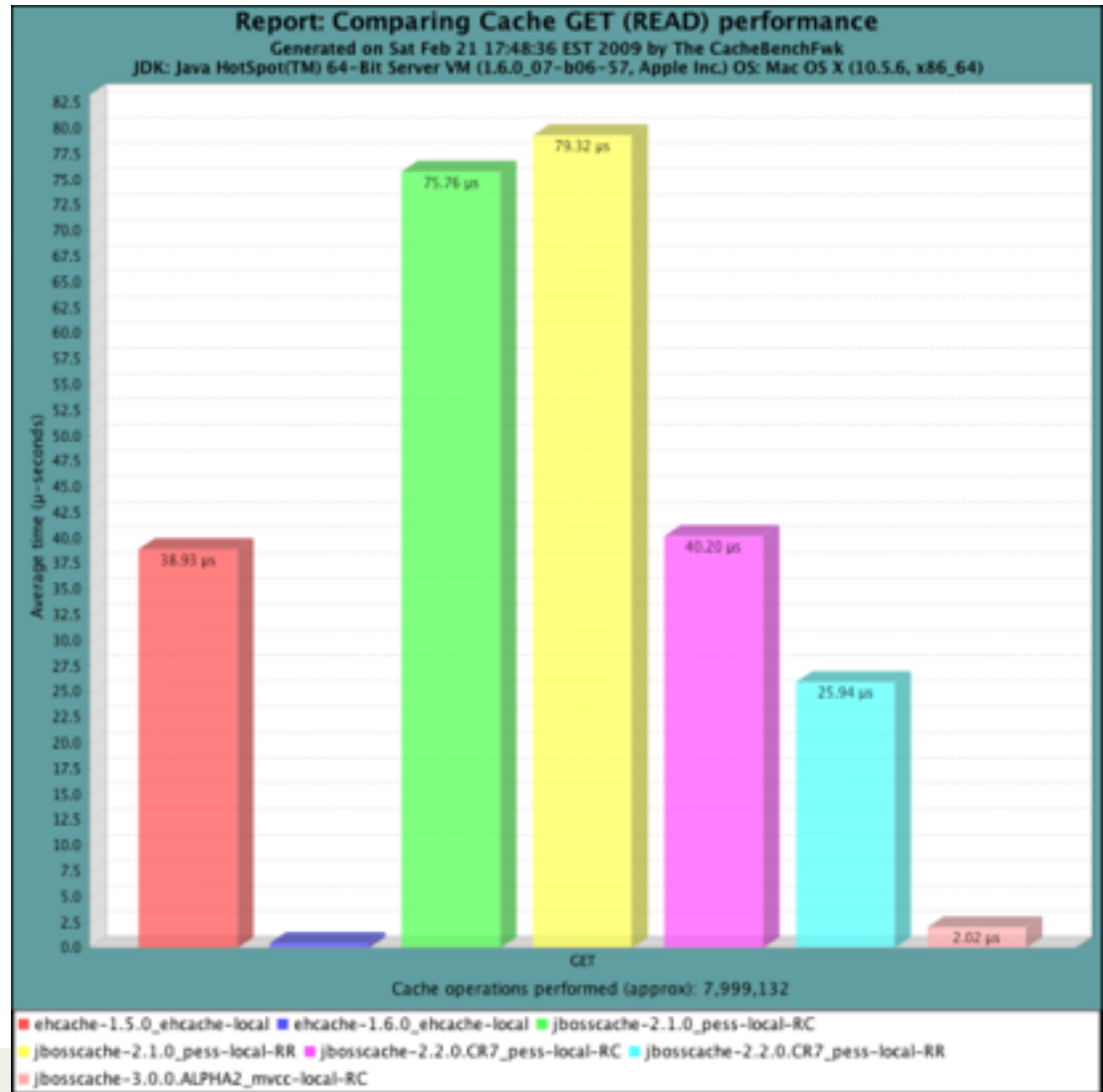
Performance

- Testing Methodology
- Results in Details
- Conclusions



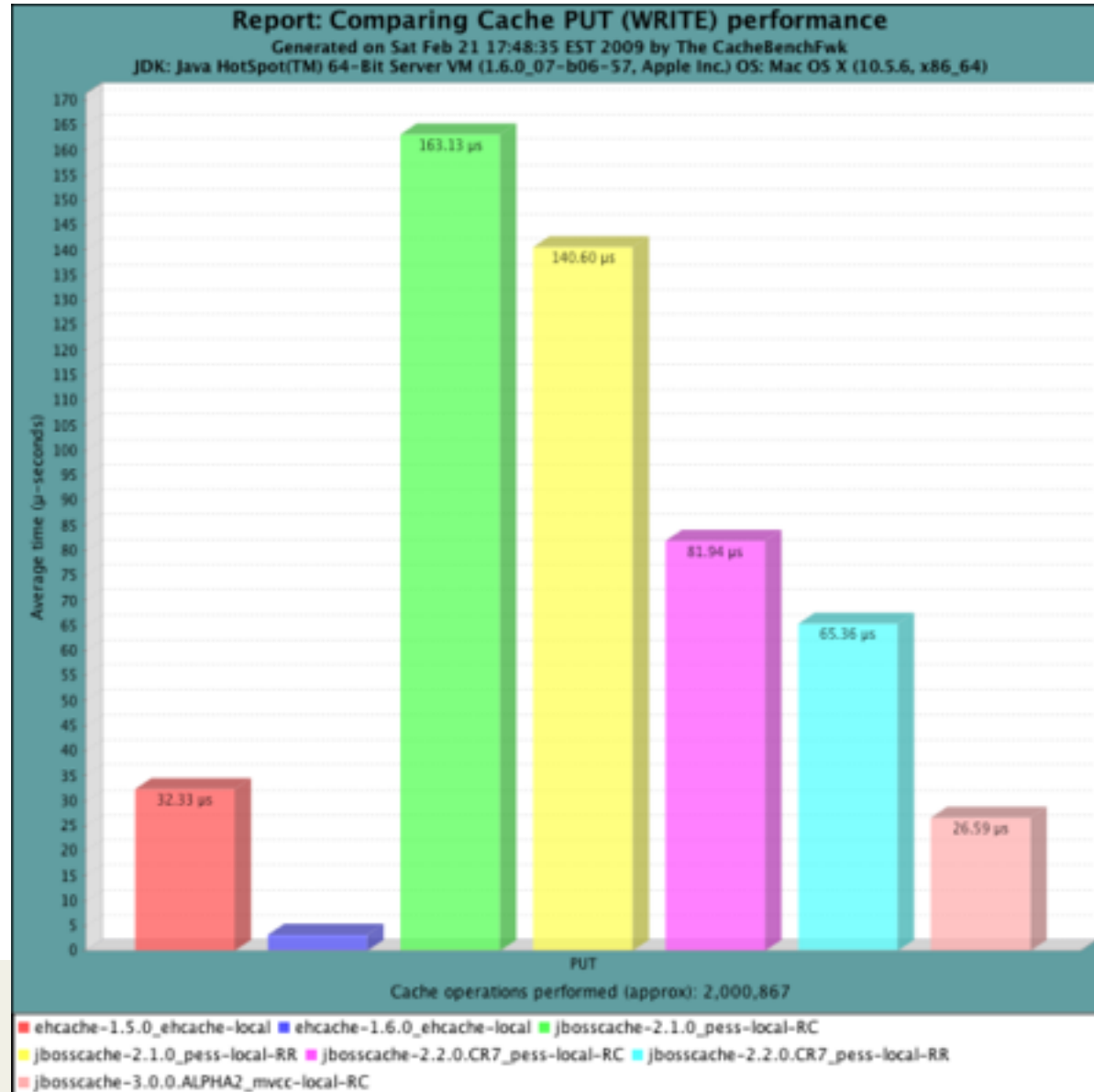
Standalone Performance

- Read Performance

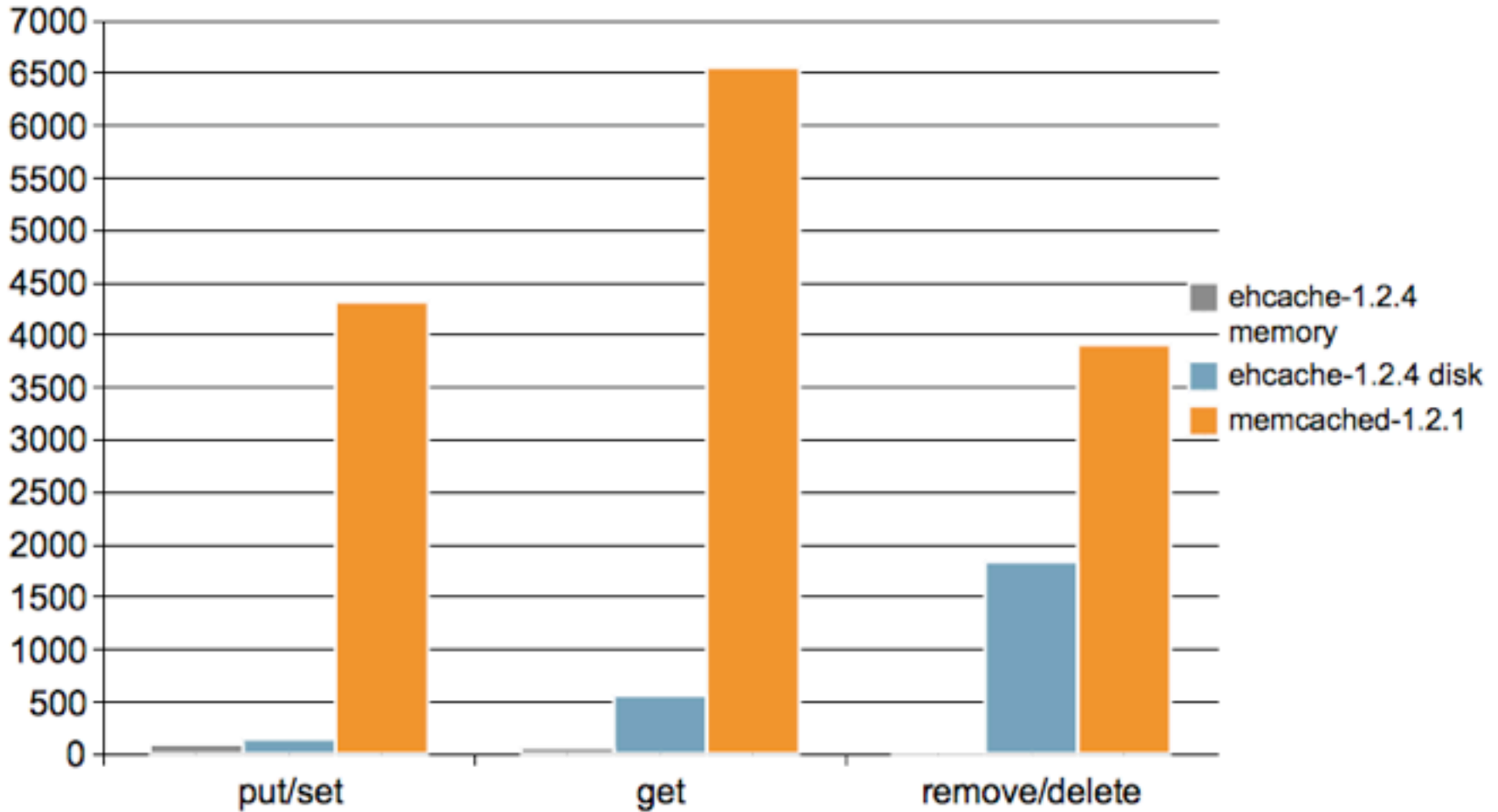


Standalone Performance

- Put Performance



Ehcache in-process vs Memcached



Ehcache with Terracotta vs the Rest

■ Application

- Tests done with Owners = 25K and 125K which translates to total objects of 0.3 M and 1.5 M
- Minimal tuning.

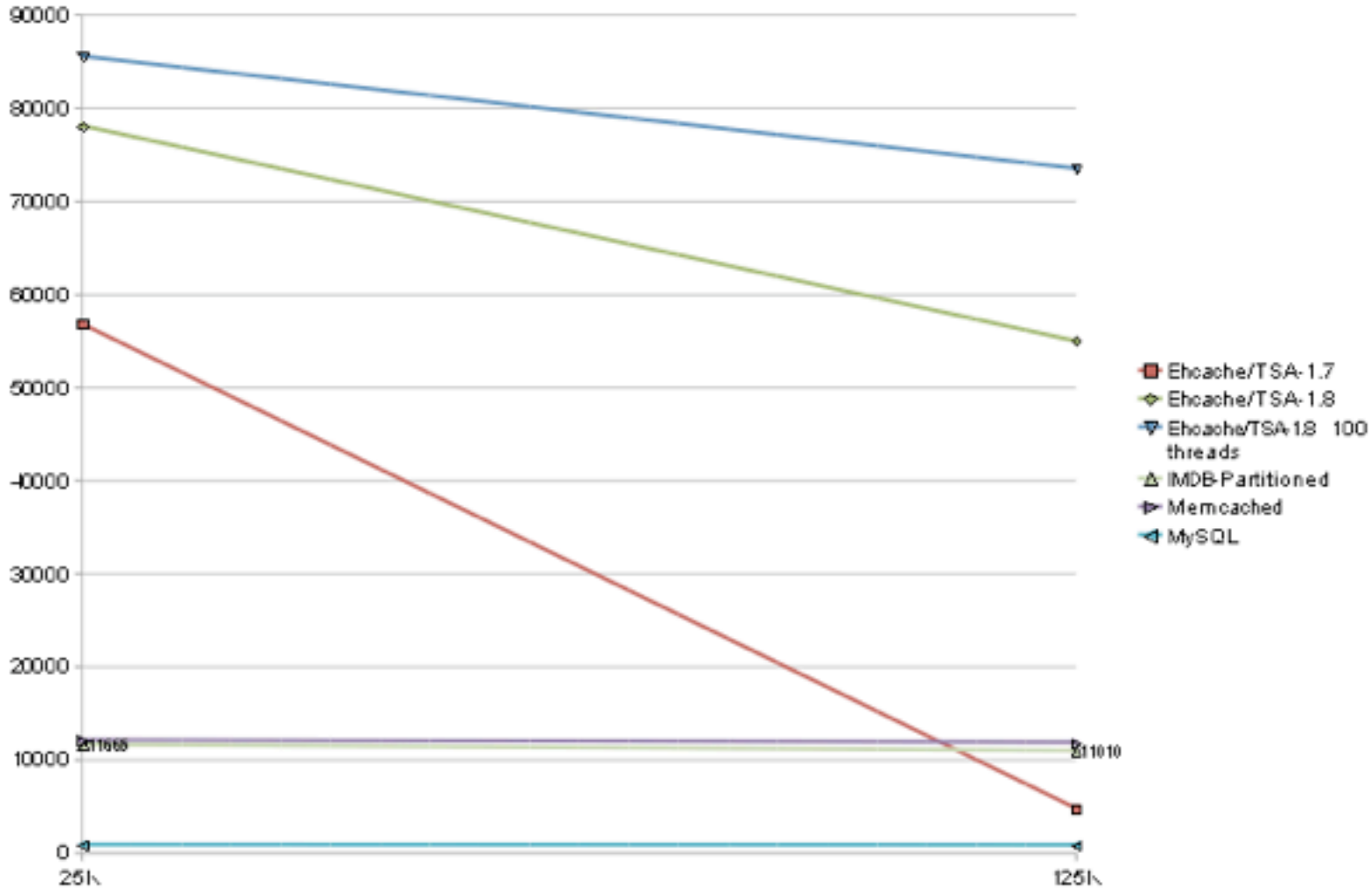
■ Cluster Configuration:

- 8 Client JVMs (1.75G Heap)
- 1 (+0) Terracotta Servers (6G Heap)
- MySql: sales18.

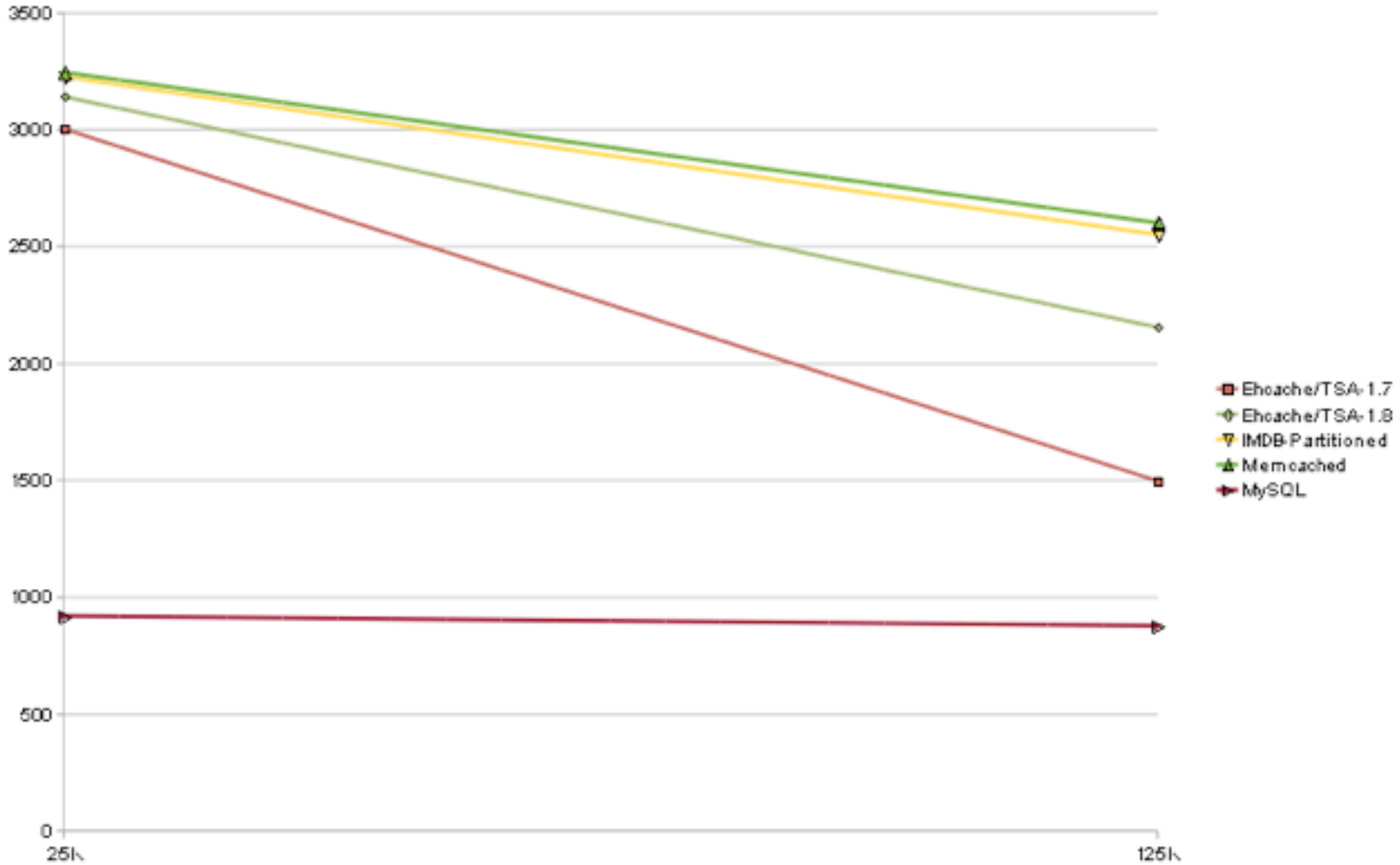
Ehcache with Terracotta vs the Rest

- **Ehcache**
 - Replicated with RMI not included because not coherent
 - Single TSA Server
 - 15 threads and some with 100 threads
- **IMDG**
 - 15 threads
 - Cache deployed in Partitioned Mode
 - Tests were also done with Replicated – which did well for small cache sizes but failed to complete with larger cache sizes. So, it is not included.
- **memcached**
 - 15 threads
 - 1 server

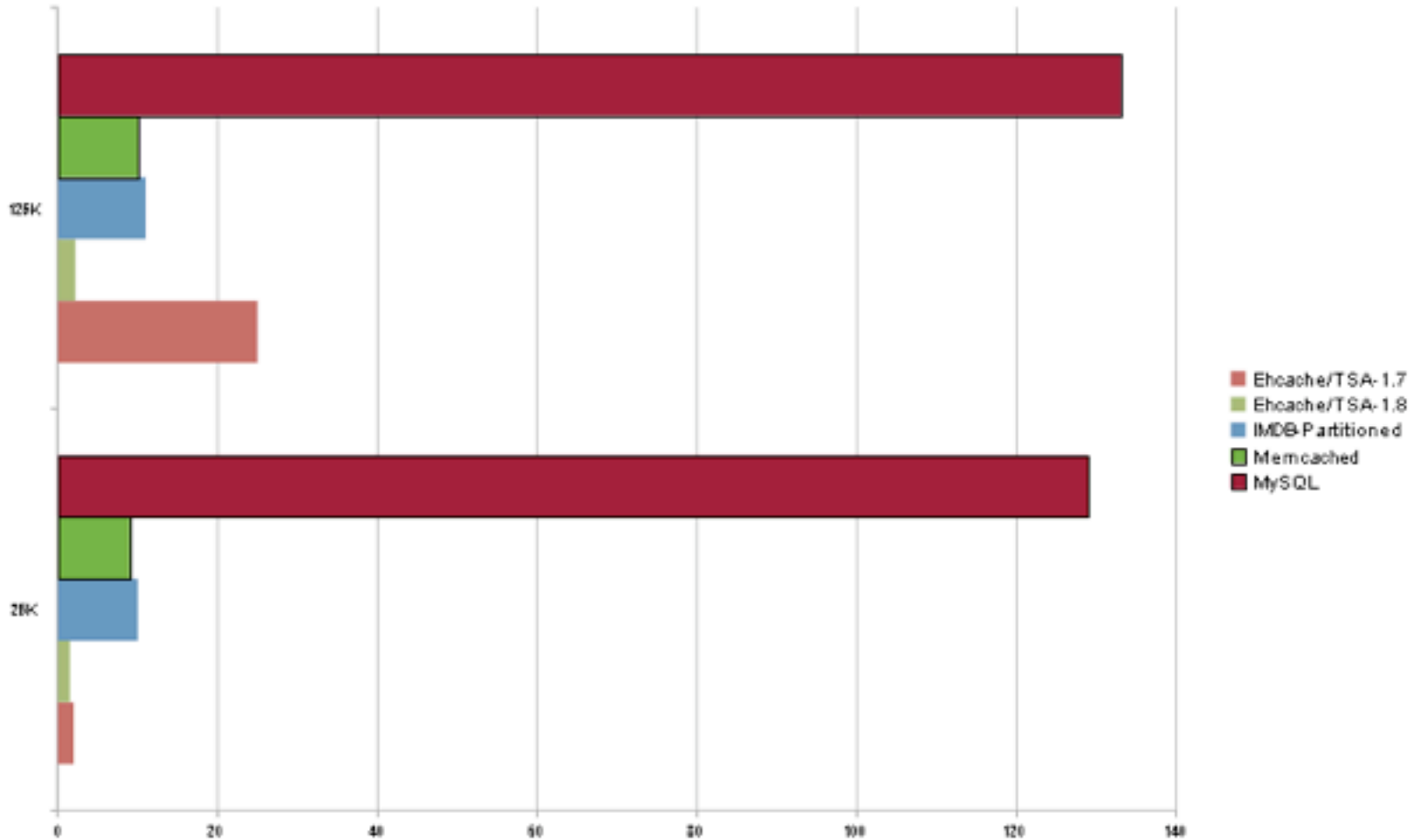
Hibernate - Read Only TPS



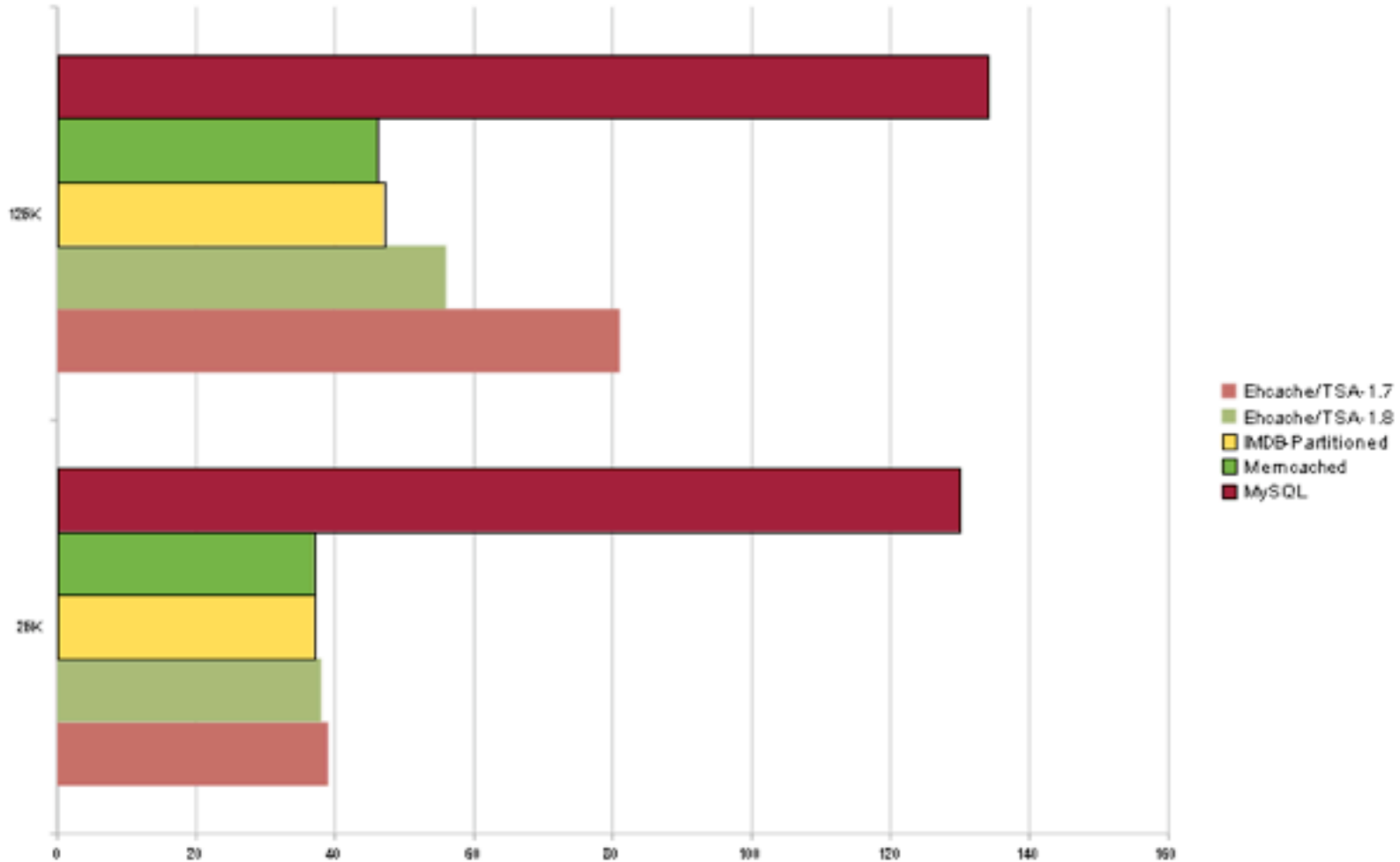
Hibernate - Read Write TPS



Hibernate - Read Only Latency



Hibernate - Read Write Latency



Test Source

- The code behind the benchmarks is in the Terracotta Community SVN repository.
- Download <https://svn.terracotta.org/repo/forge/projects/ehcacheperf/>
(Terracotta Community Login Required)


Performance Conclusions

- **With Hibernate, Using Spring Pet Clinic**
 - After app servers and DBs tuned by independent 3rd parties
 - 30-95% database load reduction
 - 80 times read-only performance of MySQL
 - Notably lower latency
- **1.5 ms versus 120 ms for database (25k)**



Wrap Up

Terracotta Commercial Products

Scale as you grow 

	DX	EX	FX
Ehcache	Enterprise Ehcache	Enterprise Distributed Cache	
	Enterprise Hibernate Cache	Enterprise Hibernate Distributed Cache	
Quartz	Enterprise Job Scheduler	Enterprise Distributed Job Scheduler	
Web Sessions	N/A	HA and Scale for Web Applications	
Terracotta for Spring	N/A	Plug-in Capacity for Spring Applications	
		Terracotta Scalability Platform	

Enterprise Support Included in Commercial Offerings:

- 24x7 support for mission critical business functions
- Guaranteed time-to-respond service level agreement (SLA)
- Thoroughly tested patches

Additional Ehcache Information

- Website: www.ehcache.org
- Documentation: www.ehcache.org/documentation
- Hibernate: www.ehcache.org/documentation/hibernate.html
- Commercial Products: www.terracotta.org/ehcache/
- Twitter: www.twitter.com/Ehcache

Terracotta Contact Information

- Website: www.terraccottatech.com
- Telephone: +1 415-738-4000
- Email: info@terraccottatech.com
- Facebook: www.facebook.com/Terracotta
- Twitter: www.twitter.com/TerracottaTech