JBoss Community

manik@jboss.org | http://twitter.com/maniksurtani | http://blog.infinispan.org

Large scale caching with Infinispan

Manik Surtani
Founder and project lead, Infinispan
Red Hat, Inc.

JBoss Community

manik@jboss.org | http://twitter.com/maniksurtani | http://blog.infinispan.org

Who is Manik?

- R&D Engineer, Red Hat Inc.
  - Founder and project lead, Infinispan
  - Project lead, JBoss Cache
  - Frequent speaker on cloud computing and cloud data storage

http://twitter.com/maniksurtani
http://blog.infinispan.org

Agenda

- Introducing Infinispan
- Using a data grid as a large-scale cache
- Q&A

manik@jboss.org | http://twitter.com/maniksurtani | http://blog.infinispan.org

BUT FIRST...

What is a data grid?

manik@jboss.org | http://twitter.com/maniksurtani | http://blog.infinispan.org

What is a data grid?

- Commodity servers clustered together
  - Connected by (hopefully) fast networks
- Expose collective storage capacity of the cluster as a single resource
  - Distributed by nature
  - Fault tolerant by nature
- May expose either disk capacity (Hadoop, BigTable) or memory capacity (Gigaspaces, Cassandra, Coherence, Infinispan)

Introducing

JBoss Community

manik@jboss.org | http://twitter.com/maniksurtani | http://blog.infinispan.org

Infinispan

What is Infinispan?

- Open source (LGPL) in-memory Data Grid
- Some concepts from Amazon Dynamo

2 usage modes

- Embedded
- Client-server
  - memcached
  - Hot Rod
  - REST

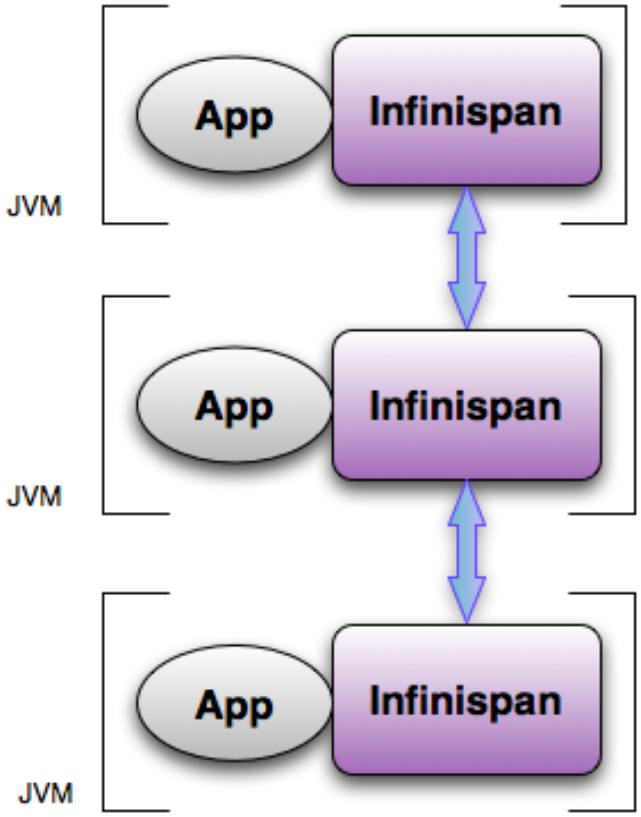
manik@jboss.org | http://twitter.com/maniksurtani | http://blog.infinispan.org

Infinispan

Embedded Mode

manik@jboss.org | http://twitter.com/maniksurtani | http://blog.infinispan.org

P2P Embedded Architecture



manik@jboss.org | http://twitter.com/maniksurtani | http://blog.infinispan.org

Infinispan

API

- Map-like key/value store
- Upcoming JPA-like layer
  - Hibernate OGM
- Other high-level APIs being discussed in the community e.g., ActiveRecord

manik@jboss.org | http://twitter.com/maniksurtani | http://blog.infinispan.org

Infinispan

Consistent hash based distribution

- Self healing
- No single point of failure

Highly concurrent

- MVCC locking

manik@jboss.org | http://twitter.com/maniksurtani | http://blog.infinispan.org

Infinispan

Persistence

- Not just in memory!
- Write through and write behind
- Pluggable “drivers”

Eviction and expiry

- Efficient, adaptive algorithm: LIRS
- Addresses shortcomings of LRU & FIFO

manik@jboss.org | http://twitter.com/maniksurtani | http://blog.infinispan.org

Infinispan

XA Transactions

- 2-phase commit based
- Deadlock detection algorithms
- Coming soon: Atomic Broadcast

manik@jboss.org | http://twitter.com/maniksurtani | http://blog.infinispan.org

Infinispan

Map/Reduce

- In an early, tech preview state right now

Querying

- Using Lucene and Hibernate Search to index

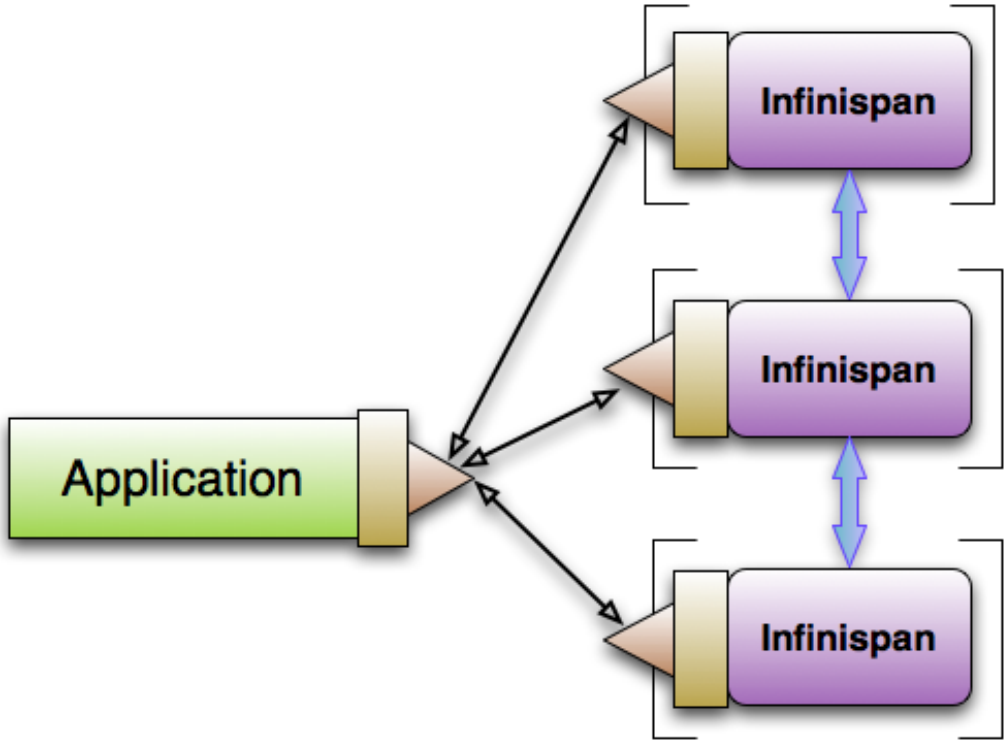
manik@jboss.org | http://twitter.com/maniksurtani | http://blog.infinispan.org

Infinispan

Client/Server Mode

manik@jboss.org | http://twitter.com/maniksurtani | http://blog.infinispan.org

Client/Server Architecture



Supported Protocols

- REST
- Memcached
- Hot Rod

manik@jboss.org | http://twitter.com/maniksurtani | http://blog.infinispan.org

What is Hot Rod?

- Wire protocol for client server communications
- Open
- Language independent
- Built-in failover and load balancing
- Smart routing

manik@jboss.org | http://twitter.com/maniksurtani | http://blog.infinispan.org

Server Endpoint Comparison

	Protocol	Client Libraries	Clustered ?	Smart Routing	Load Balancing/ Failover
REST	**Text**	N/A	**Yes**	**No**	Any HTTP load balancer
Memcached	**Text**	Plenty	**Yes**	**No**	Only with predefined server list
Hot Rod	**Binary**	Java Python Ruby	**Yes**	**Yes**	Dynamic

manik@jboss.org | http://twitter.com/maniksurtani | http://blog.infinispan.org

Starting an Infinispan Server

- Hot Rod or memcached server endpoint

```
$ bin/startServer.sh -r hotrod \
                     -c infinispan.xml
$ bin/startServer.sh -r memcached \
                     -c infinispan.xml
```


manik@jboss.org | http://twitter.com/maniksurtani | http://blog.infinispan.org

Starting an Infinispan Server

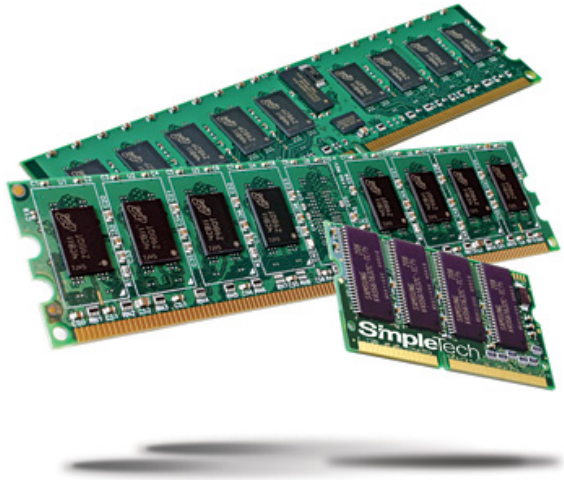
- REST endpoint
  - Deploy `infinispan-server-rest.war` in your favourite servlet container.

Caching

manik@jboss.org | http://twitter.com/maniksurtani | http://blog.infinispan.org

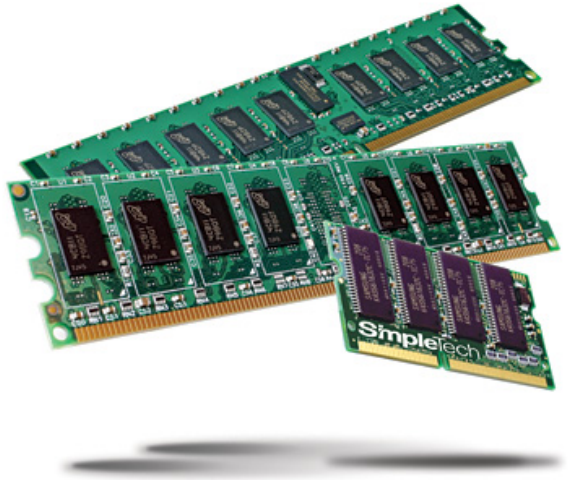
Why use a memory cache?

- Performance booster
- Good for data that is:
  - Hard to calculate
  - Expensive to retrieve
    - E.g., from a DB or a Web Service
  - Frequently accessed



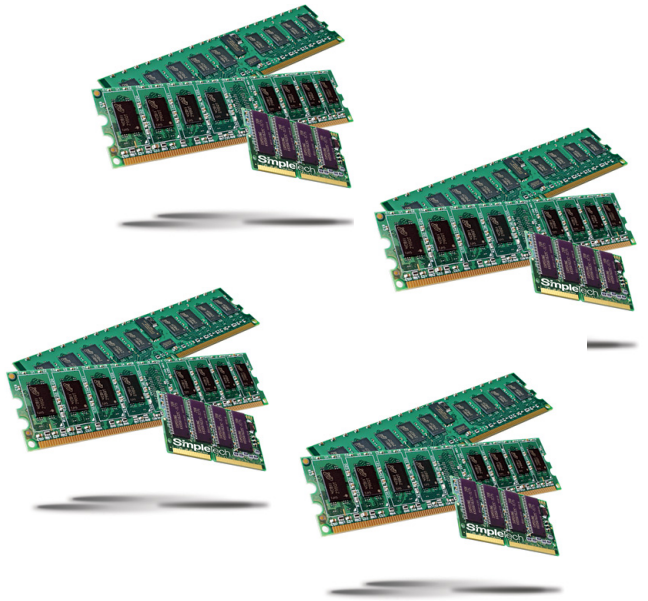
Local, in-memory object cache

- Better than a HashMap
  - Greater concurrency
  - Built-in eviction, prevents OOMs!
  - Overflow to disk
  - Warm starts, preloading
  - Events, notifications
  - Highly configurable locking strategies
  - JTA compatible
  - JMX monitoring



Clustered in-memory cache

- Performance booster for clustered apps
- Similar to a local cache
- Needs to be cluster-aware
- More shared-cache space!
- INVALIDATION mode if backed by DB?



Questions?

http://www.infinispan.org
http://blog.infinispan.org
http://github.com/infinispan
http://twitter.com/infinispan - #infinispan
#infinispan on FreeNode (IRC)

manik@jboss.org | http://twitter.com/maniksurtani | http://blog.infinispan.org