

# Caching

- notes thoughts and discussion starter on the current implementation found in <https://svn.apache.org/repos/asf/geronimo/sandbox/geronimo-cache>  
\*

## Vision

The idea is to provide a single API (essentially `java.util.Map`) that will allow Geronimo (and POJO's for that matter) to cache objects. Some aspects of the API and impl;

- The API will be the same for clients regardless of the configured implementation.
- The impl will be specified via configuration
- The impl will work for pojo's as well as within/part of Geronimo
- Transport, Storage, Topology and Cache are all loosely coupled

## Topologies Supported

A topology refers to the way the cache node(s) are laid out across a network and how the clients of the cache interact with the cache.

- Local - in memory, not replicated, no redundancy, very fast read & write
- Fully Replicated - locally in memory but replicated to other caches, redundant, very fast local read, write depends on transport impl chosen
- Distributed - partially replicated, partially remote

## Topologies Envisioned

- Fully Remote - a cache that is remote to all clients, use full in some situations for access to expensive to find or generate information that is accessed infrequently. Other use cases I'm sure apply here.

## Current State

The current state is very alpha and more or less in a proof of concept state but many things do work.

### Stuff that does work

- Local cache
- Fully Replicated Cache
  - JMS Transport (this is in the latest patch but not committed yet)
  - JIRA -> <http://issues.apache.org/jira/browse/GERONIMO-2046?page=all>
- Memory Storage

### Stuff that does not work but is in progress

- Fully Replicated Cache
  - UDP Transport
  - Active IO Transport