











## Cache Coherency is Also Important

- A cache that is "coherent" shows the same contents at every location within a distributed or clustered environment
- Caches of read-only data are automatically coherent!
- The choice for clustered caching of read/write data:
  Accept a certain amount of data staleness
  - Maintain cache data coherency across the cluster
- Clustered data coherency implies a means to *synchronize*:
  - Clustered concurrency control (like Java "synchronized")Distributed Transactional Caching
- Interposing the data caches *between* the application logic and the data source prevents loss of consistency











































## The Distributed Cache Scheme

- Distribution is invisible to application
  The application does not need to know the physical location of the data
- Recovery occurs in Parallel
   Not 1 to 1 like Active + Passive architectures
- Any Member can fail without data loss
- Configurable # backups
- No Developer or Infrastructure intervention

Copyright 2007

25

































## The Consensus Algorithm · Oracle Coherence clustering is very different All members of the cluster (JVMs) have consensus, or agreement, as to: - Who owns the primary copy of the data - Who owns the backup copy of the data

- Who has which responsibilities (by default, equal responsibility)
- All members maintain consensus at all times
- · Communication is peer-to-peer
- No central registry or hub-and-spoke architecture
- Non-blocking if a member fails (as opposed to "voting" protocols)

Copyright 2007

38















