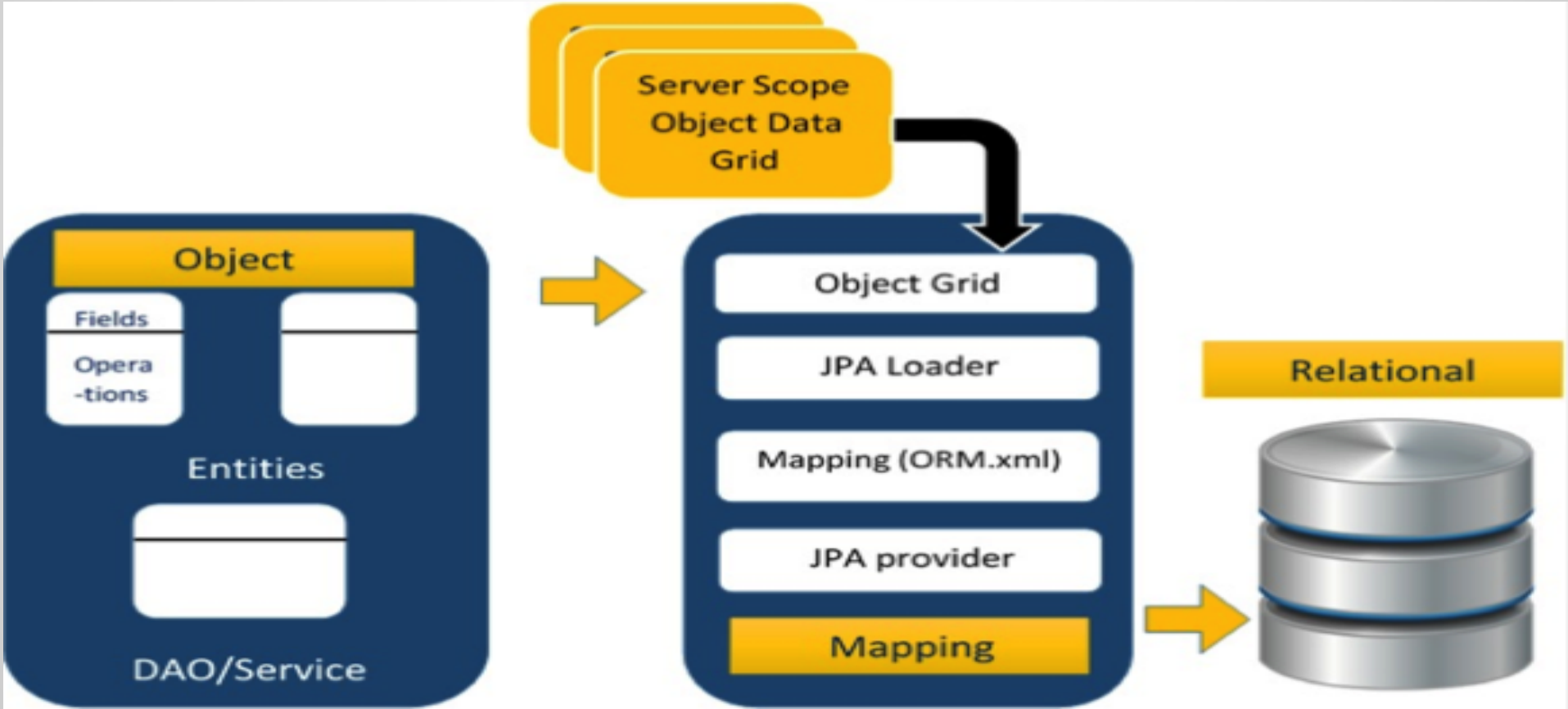
JPA

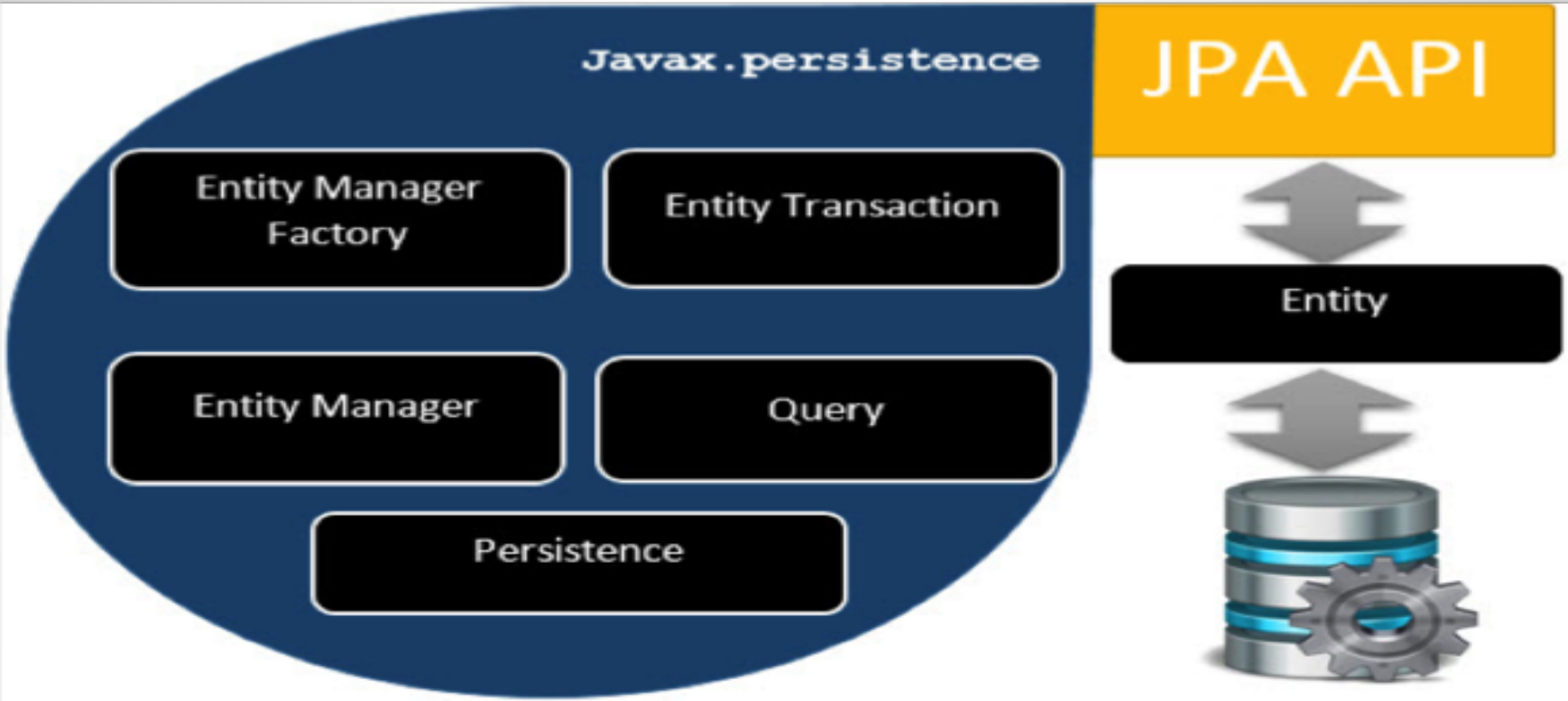
Agenda:

- ORM Components
- Jpa Class level Architecture
- Jpa Components description
- Jpa Components relation
- Entity lifecycle & Entitymanager Interface
- Getting started with JPA
- Using @Transactional with JpaSpringModule
- Examples, Best practises
- Caching
- Locking
- Transactions

ORM components

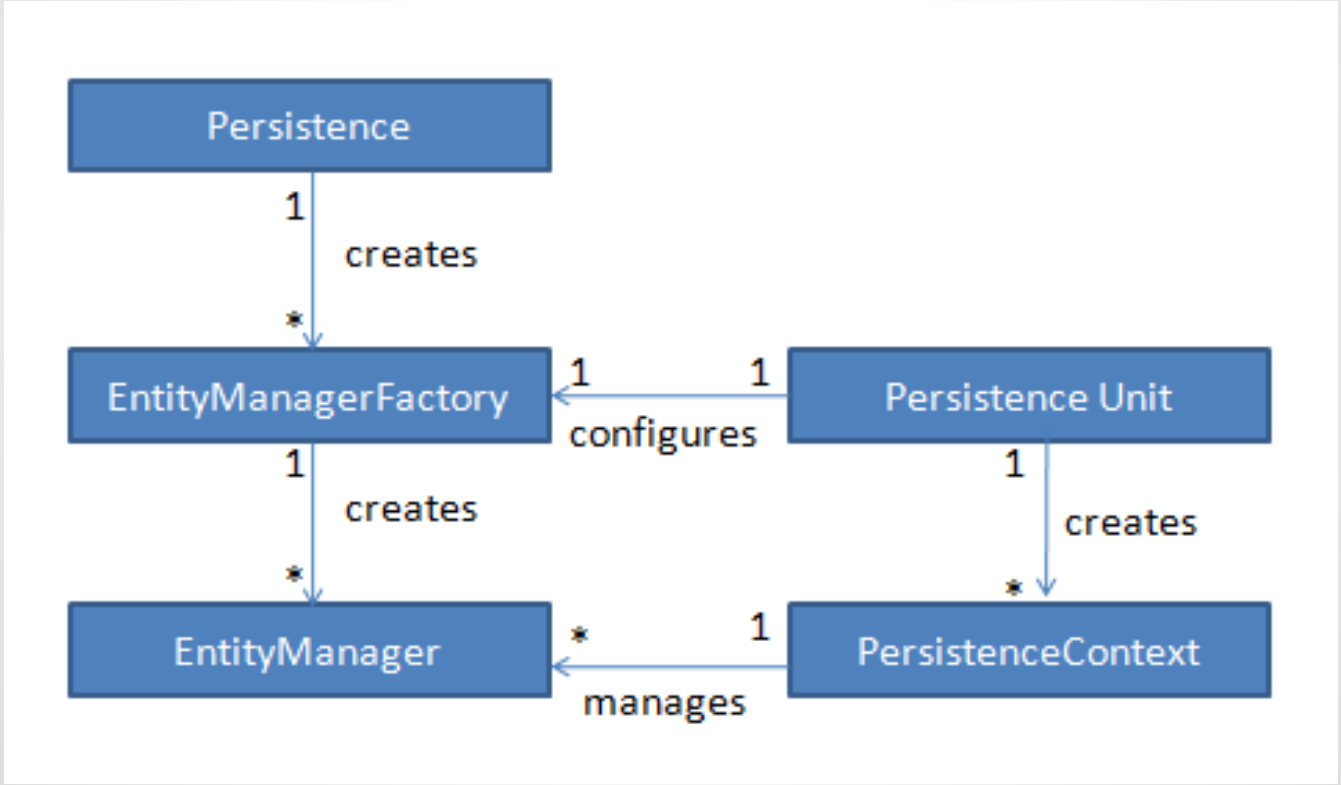


JPA Class Level Architecture

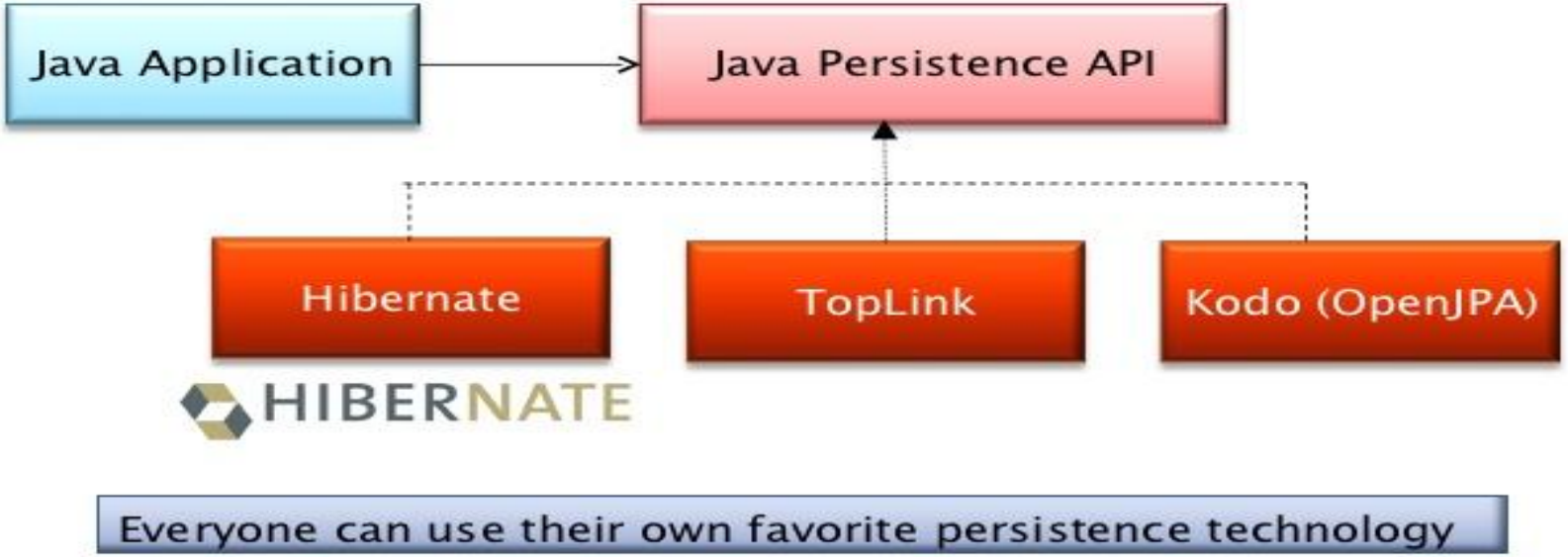


Units	Description
EntityManagerFactory	This is a factory class of EntityManager. It creates and manages multiple EntityManager instances.
EntityManager	It is an Interface, it manages the persistence operations on objects. It works like factory for Query instance.
Entity	Entities are the persistence objects, stores as records in the database.
EntityTransaction	It has one-to-one relationship with EntityManager. For each EntityManager, operations are maintained by EntityTransaction class.
Persistence	This class contain static methods to obtain EntityManagerFactory instance.
Query	This interface is implemented by each JPA vendor to obtain relational objects that meet the criteria.

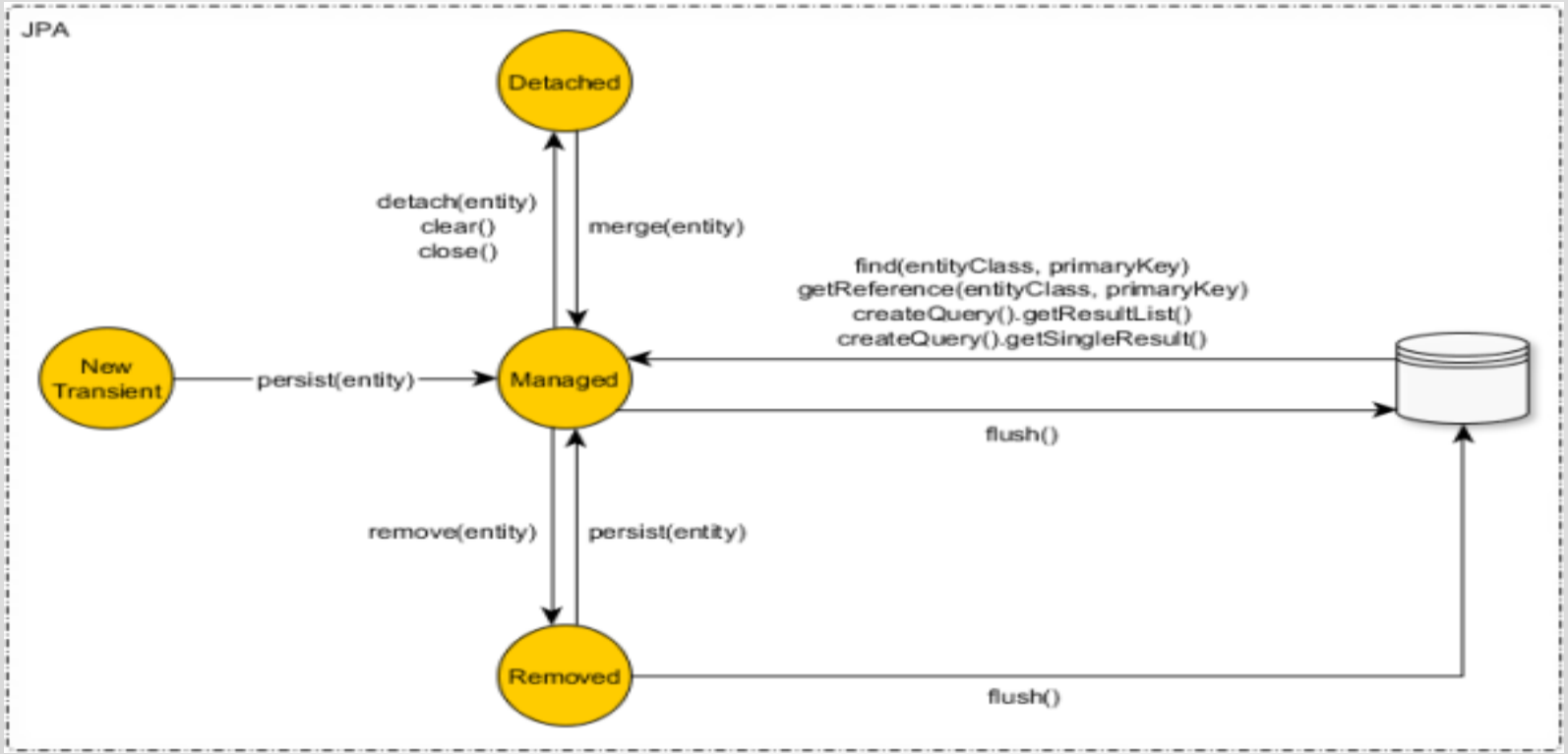
Components of JPA



JPA Architecture

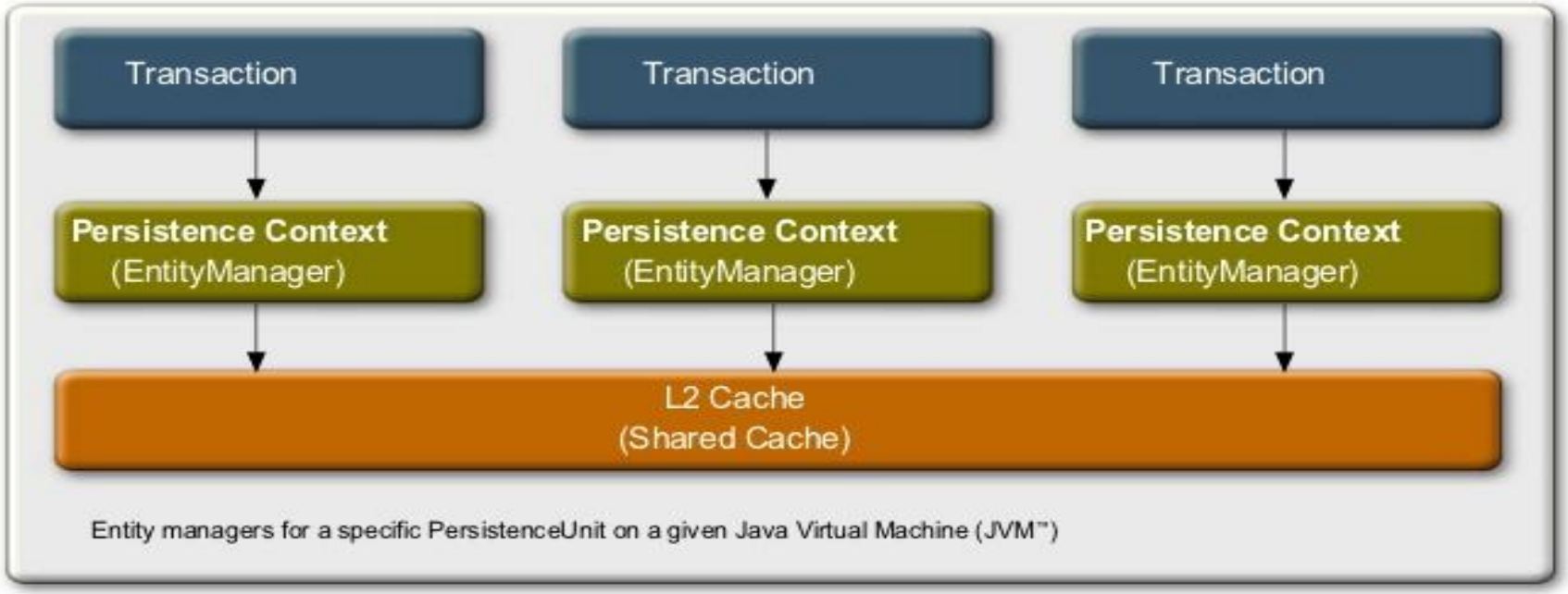


Entity Manager



Level1 and Level2 caches

Persistence Context is a Level 1 cache



The terms "Java Virtual Machine" and "JVM" mean a Virtual Machine for the Java™ Platform.
Source:http://weblogs.java.net/blog/guruwons/archive/2006/09/understanding_t.html

Lock Modes for Concurrent Entity Access

Lock Mode	Description
`OPTIMISTIC`	Obtain an optimistic read lock for all entities with version attributes.
`OPTIMISTIC_FORCE_INCREMENT`	Obtain an optimistic read lock for all entities with version attributes, and increment the version attribute value.
`PESSIMISTIC_READ`	Immediately obtain a long-term read lock on the data to prevent the data from being modified or deleted. Other transactions may read the data while the lock is maintained, but may not modify or delete the data. The persistence provider is permitted to obtain a database write lock when a read lock was requested, but not vice versa.
`PESSIMISTIC_WRITE`	Immediately obtain a long-term write lock on the data to prevent the data from being read, modified, or deleted.
`PESSIMISTIC_FORCE_INCREMENT`	Immediately obtain a long-term lock on the data to prevent the data from being modified or deleted, and increment the version attribute of versioned entities.
`READ`	A synonym for `OPTIMISTIC`. Use of `LockModeType.OPTIMISTIC` is to be preferred for new applications.
`WRITE`	A synonym for `OPTIMISTIC_FORCE_INCREMENT`. Use of `LockModeType.OPTIMISTIC_FORCE_INCREMENT` is to be preferred for new applications.
`NONE`	No additional locking will occur on the data in the database.