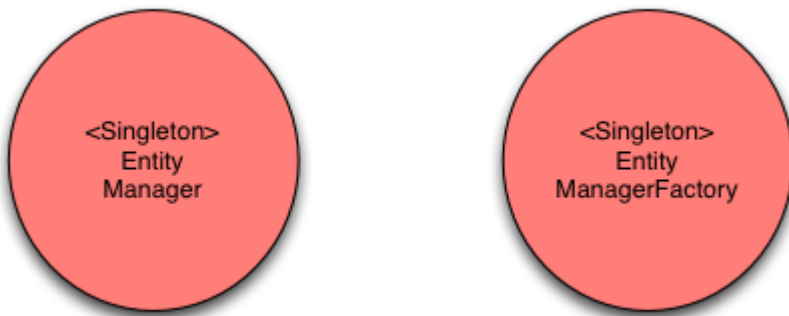


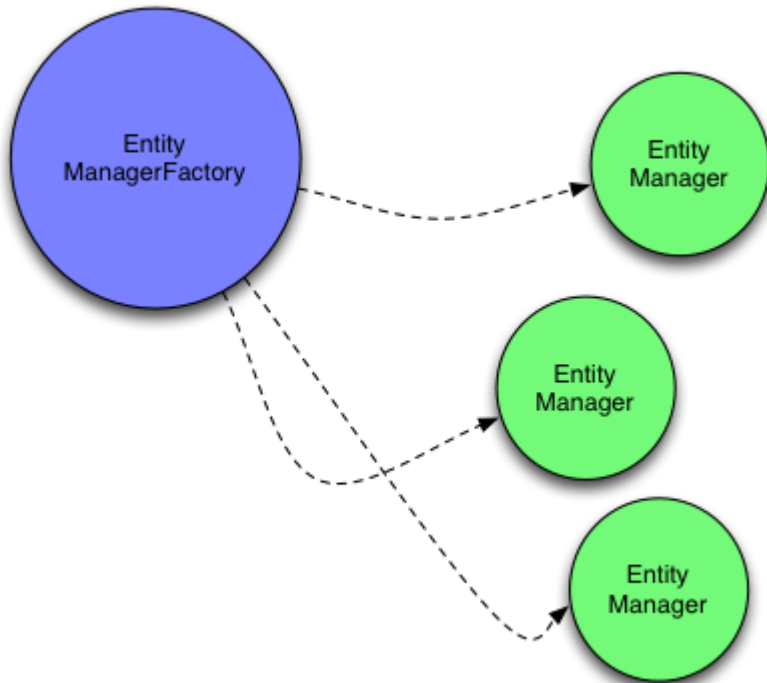
En muchas ocasiones me he encontrado con personas que tienen la siguiente duda .¿Son EntityManager y EntityManagerFactory Singleton?. O mejor aún aunque no sean Singleton ¿Se pueden tratar como si lo fueran? . Vamos a verlo un poco mas a detalle.



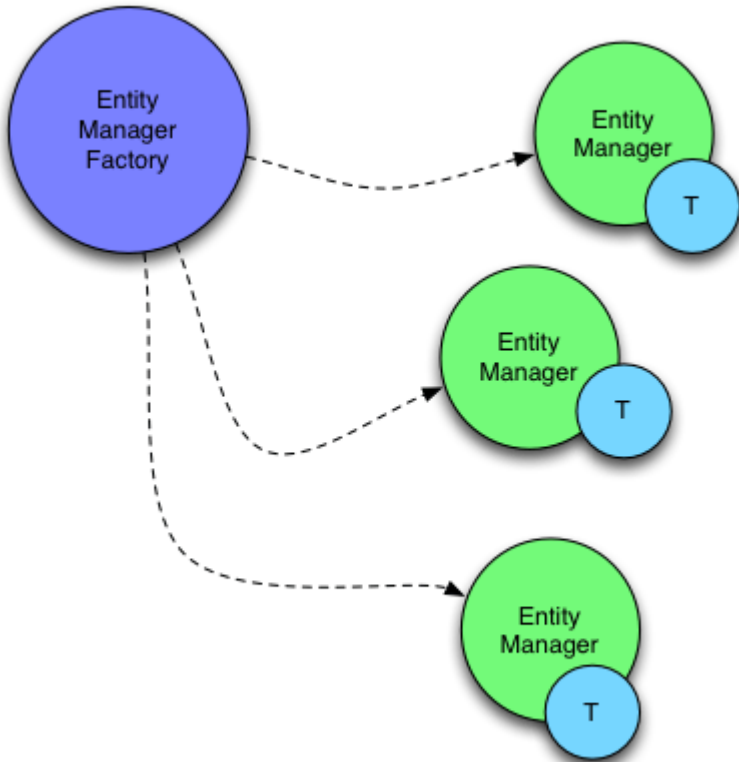
No son Singletons

Ninguna de estas dos clases son objetos de tipo Singleton . El EntityManagerFactory es la clase que se encarga de abrir la conexión a la base de datos y pone a nuestra disposición los distintos EntityManager que usemos .

EntityManager, EntityManagerFactory y Singletons

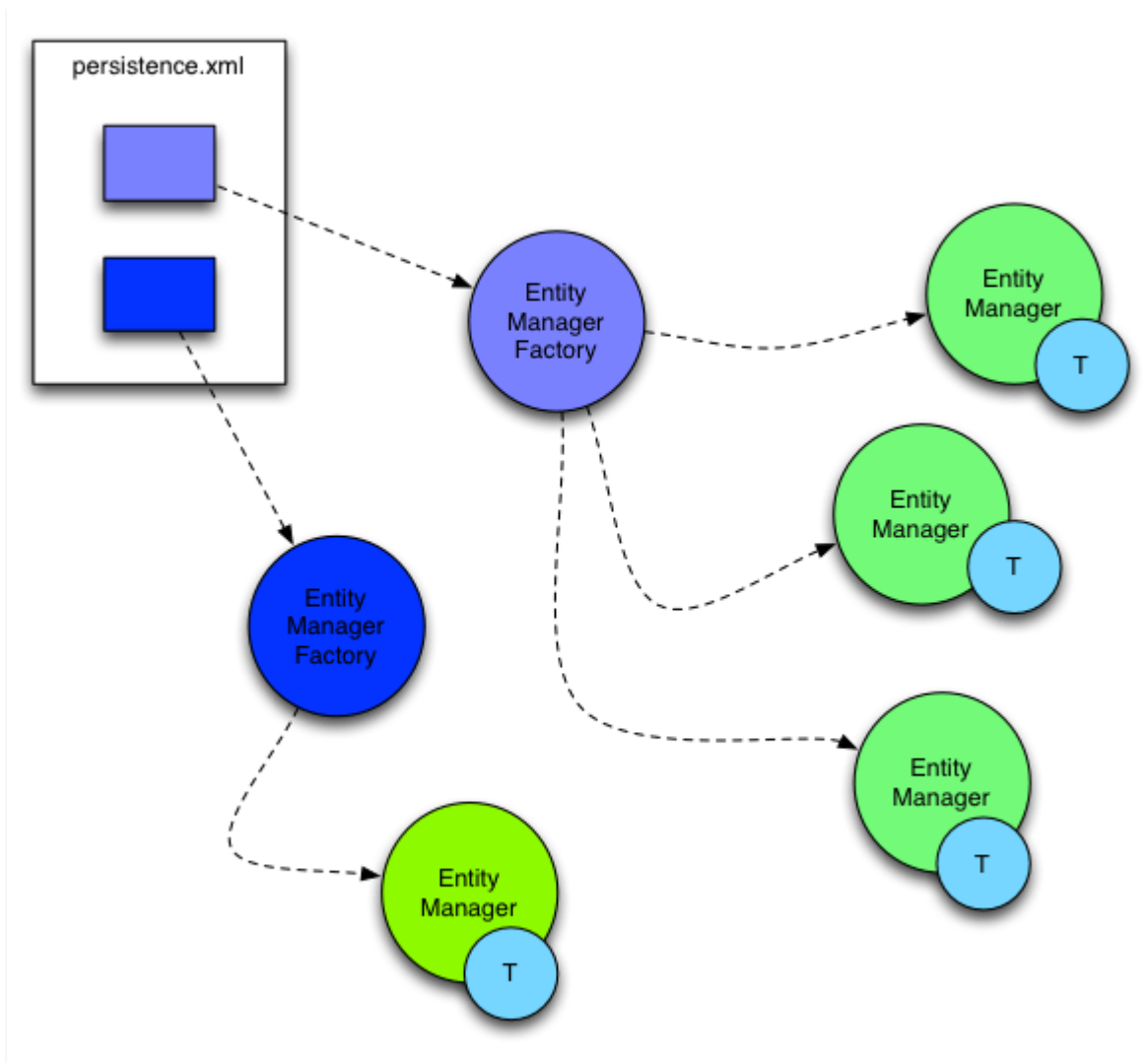


Después de ver este diagrama esta muy claro que los EntityManager no son clases Singleton ya que puede existir muchas instancias de ellas generadas a través del EntityManagerFactory. Un EntityManager esta ligado habitualmente aunque no siempre (Extended EntityManagers) a una transacción y tiene de vida lo que dura esa transacción.



EntityManagerFactory

Estos objetos son con los que mas dudas aparecen ya que se cargan una vez y se pueden configurar como si fueran Singletons pero realmente no lo son ya que podemos tener un EntityManagerFactory por cada unidad de persistencia que tengamos a nivel de persistence.xml . Cada una de estas unidades estará asociada a una base de datos concreta.



Por lo tanto hay que estar atento a como configuramos estos elementos ya que los EntityManager no deben tratarse como Singletons nunca. Mientras que los EntityManagerFactories depende mucho de la casuística de la aplicación.

Otros artículos relacionados

ArquitecturaJava : [Ejemplo de JPA](#)

ArquitecturaJava : [Ejemplo de uso EntityManager](#)

Arquitectura Java :EntityManager Métodos