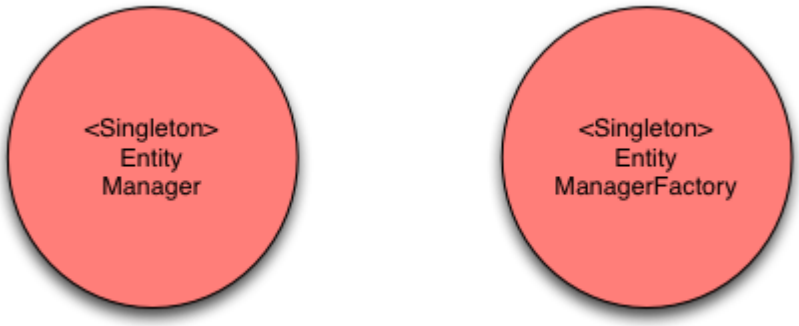
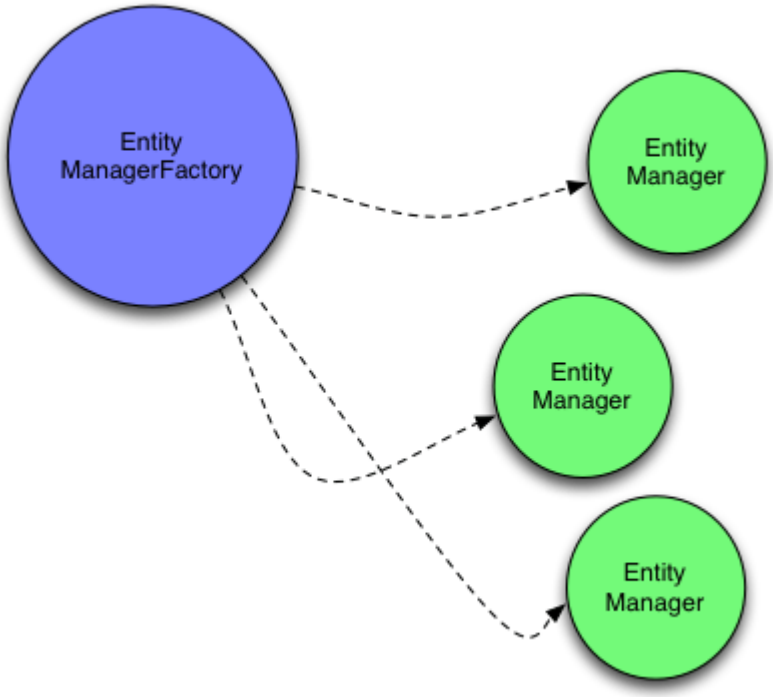
En muchas ocasiones me he encontrado con personas que tienen la siguiente duda .¿Son EntityManager y EntityManagerFactory Singleton?. O mejor aún aunque no sean Singleton ¿Se pueden tratar como si lo fueran? . Vamos a verlo un poco mas a detalle porque es un tema importante.

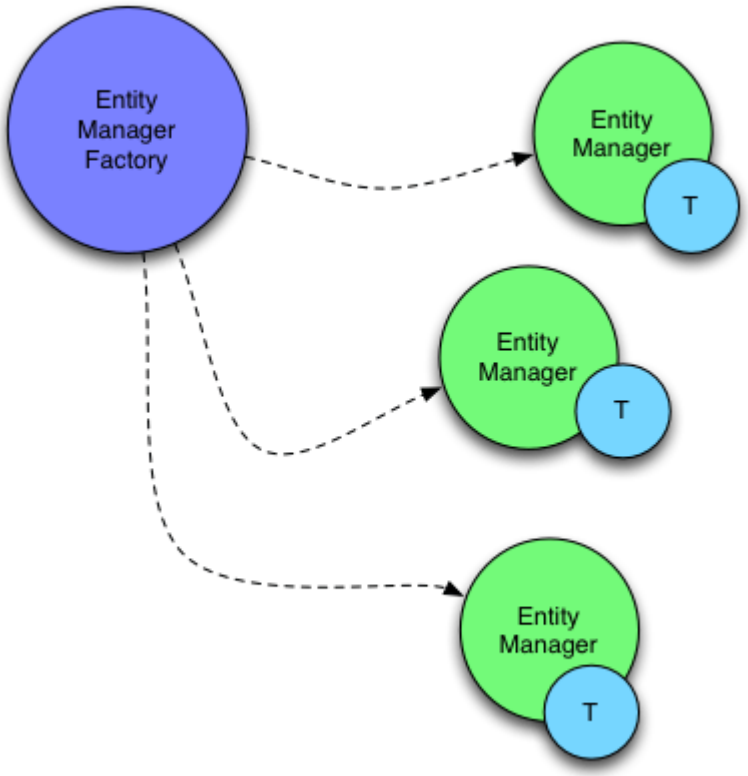


EntityManager EntityManagerFactory No son Singletons

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 . Es decir gestiona la conectividad con la base de datos en general

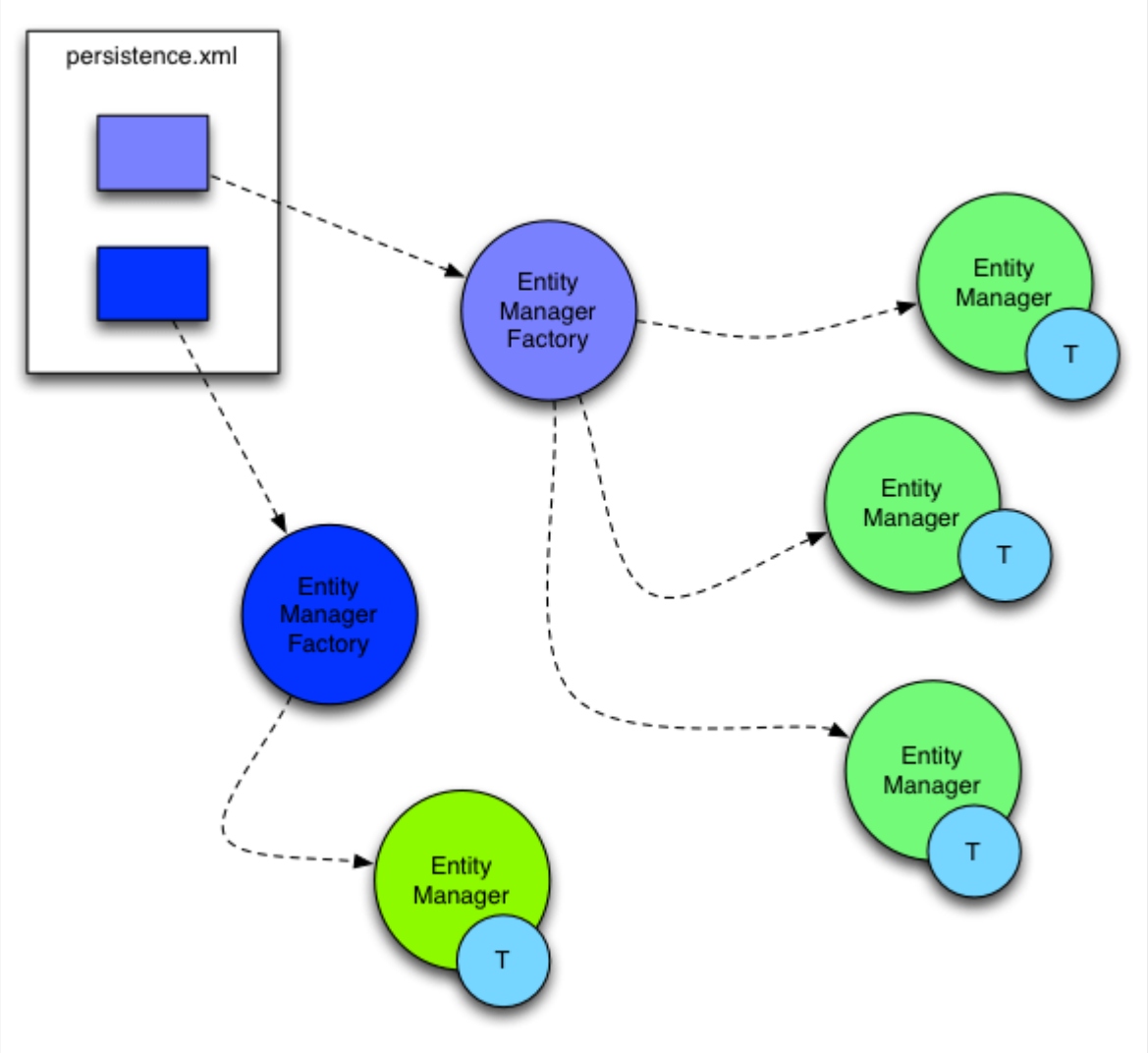


Después de ver este diagrama esta muy claro que los EntityManager no son clases Singleton ya que puede existir muchas intancias de ellas generadas a través del EntityManagerFactory. Un EntityManager esta ligado habitualmente aunque no siempre (Extended EntityManagers) a una transacción y tiene de vida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



Entrando a detalle

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 EntityManagers no deben tratarse como Singletons nunca. Mientras que los EntityManagerFactories depende mucho de la casuistica de la aplicación en una aplicación con una sola base de datos funcionan de forma similar a los objetos singleton.

Otros artículos relacionados

- Ejemplo de JPA (Java Persistence API)
- JPA Remove y Objetos gestionados
- JPA Merge y el manejo del EntityManager
- Utilizando Java Singleton Properties

- JPA