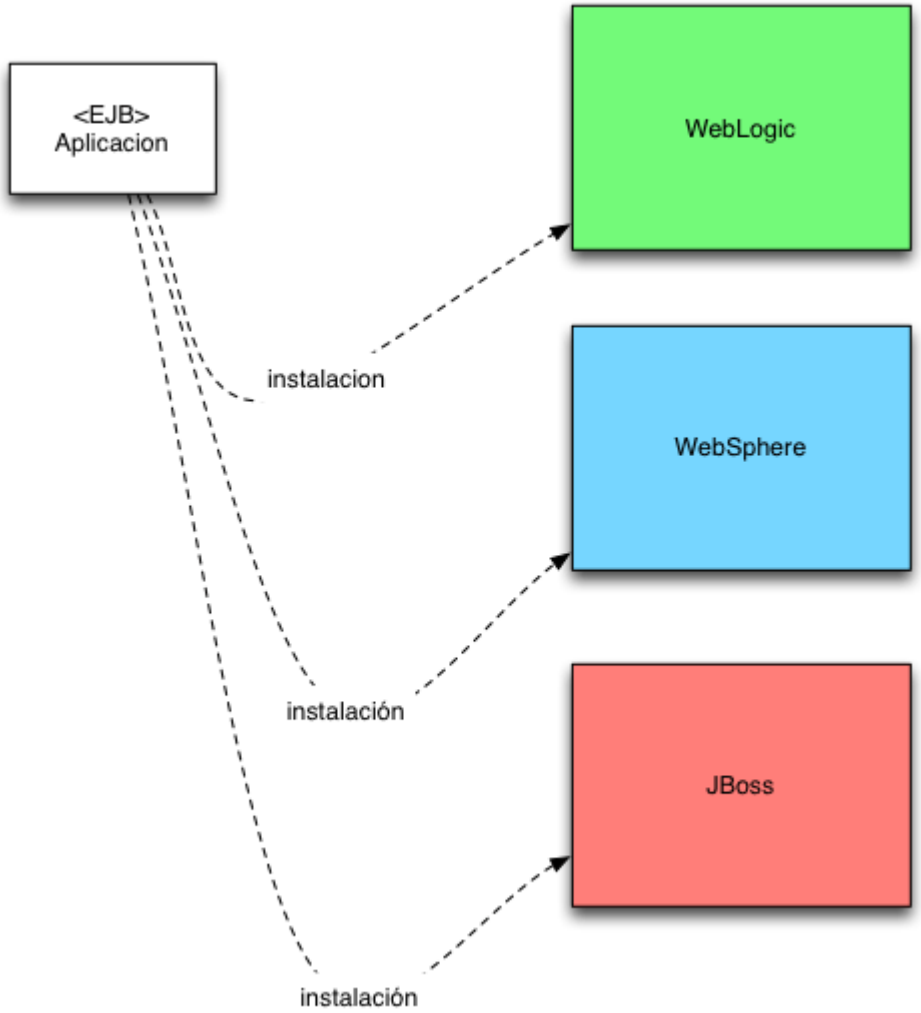
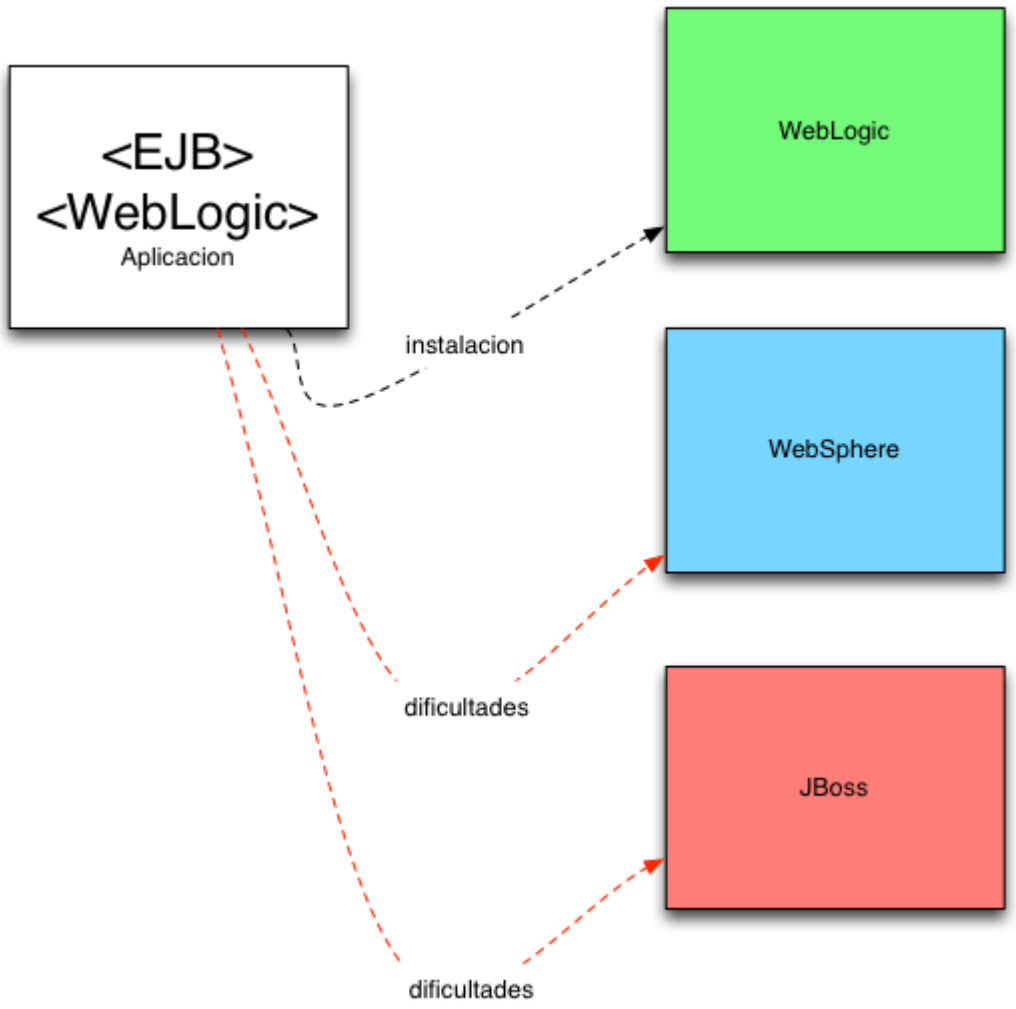
Acabamos de ver como usar Java equals y hashcode

Hoy en día dos de las tecnologías más habituales en el mundo Java para trabajar son los EJBs y Spring Framework. Entre ambas se reparten una cuota de mercado importante. En muchas ocasiones me he encontrado opiniones muy favorables al uso de EJB debido a que es el standard. Esto en teoría facilita sobre manera la migración de las aplicaciones entre un servidor de aplicaciones y otro.



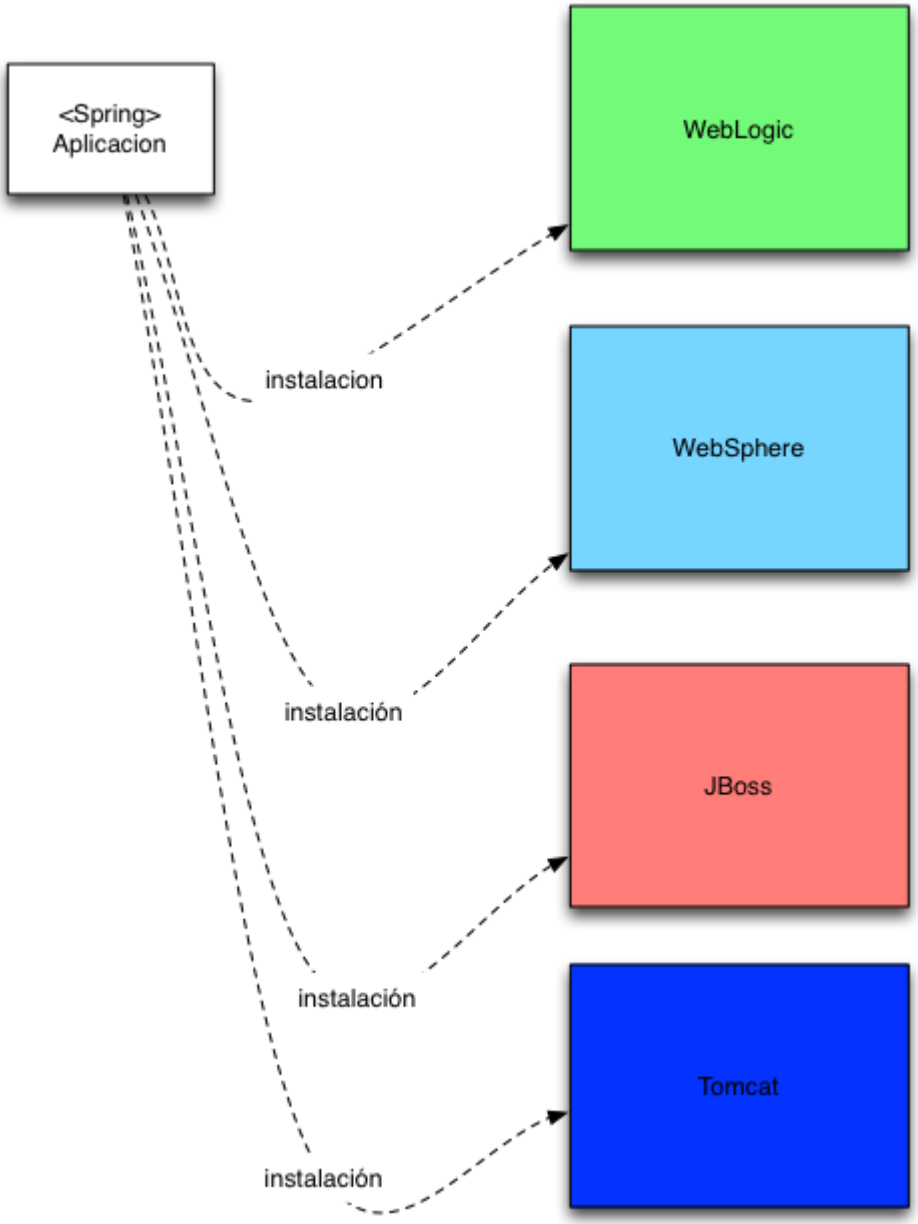
EJBs y Standards

Si bien lo anteriormente expuesto es cierto, la realidad es que es un “cierto” relativo ya que muchas veces los servidores de aplicaciones nos aportan funcionalidades aparte que les diferencian de la competencia. Temas como Caches, Persistencias Avanzadas, Cluster etc etc. Características que vamos a querer usar en muchas ocasiones y que pueden convertir a nuestra aplicación en una aplicación no tan “standard” como en un principio teorizábamos.



Spring Framework y Aislamiento

Las aplicaciones desarrolladas con Spring Framework no se basan en los standares, pero tienen una ventaja bastante peculiar. La aplicación misma se encarga de la Cache, Seguridad, Acceso a Datos etc. Por lo tanto necesitará el servidor de aplicaciones para muy poco. De hecho a veces los administradores de sistemas se quejan de que las consolas del servidor de aplicaciones les valen para poco al estar desarrollada la aplicación con Spring Framework. Esto quiere decir que la aplicación esta muy muy aislada. Este fuerte nivel de aislamiento hace que en muchos casos migrar la aplicación de servidor sea muy sencillo ya que no usa casi nada del servidor y no solo eso, sino que al ser el aislamiento tan fuerte la podremos desplegar también en un Servlet Container como Tomcat sin problemas ya que no usa EJBs.



Eso sí recordemos que si nos ceñimos al standard la migración será más sencilla que con Spring.

Otros Artículos relacionados:

Spring y Responsabilidades

Spring PropertyPlaceHolder