

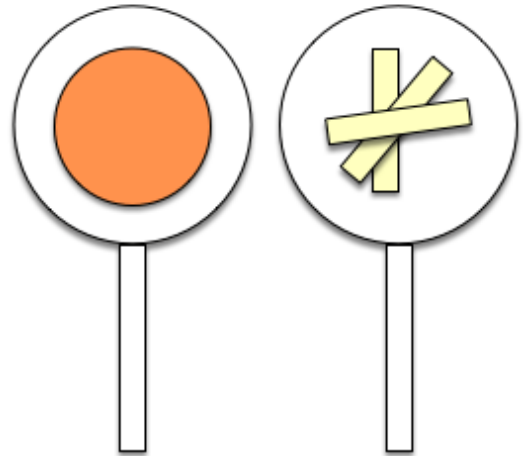
La diferencia entre framework vs librería es a veces difícil de entender sobre todo cuando uno está empezando. ¿Qué diferencia existe entre un framework vs librería? .Vamos a intentarlo explicar con un ejemplo ajeno a la programación .Supongamos que nosotros queremos cocinar una hamburguesa con patatas. En principio se trata de una operación muy muy sencilla . Necesitamos la hamburguesa , las patatas y si lo queremos hacer rápido un par de sartenes para freír cada una de ellas.

Hamburguesa

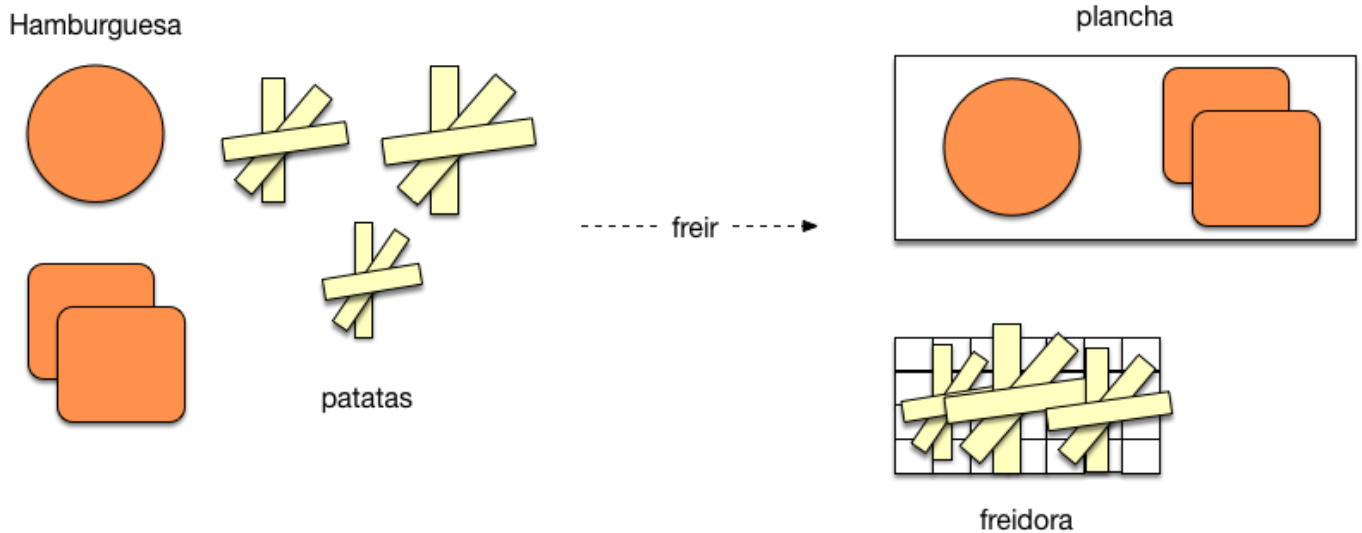


patatas

----- freir ----->



Hasta aquí todo es correcto , podemos cocinar nuestra hamburguesa con patatas. Aquí es donde podemos realizar un símil con el concepto de construir un programa, simplemente hemos usado los métodos standard para construirlo y punto. ¿Se puede hacer mejor?... uhhh quizás sí la verdad es que freír la hamburguesa quizás no es lo mas sano. La podemos hacer a la plancha y nos quedará mejor y mas sana. De igual forma quizás sea más limpio freír las patatas en una freidora con aceite nuevo ya que podemos freír unas cuantas más (estas no sobran nunca).



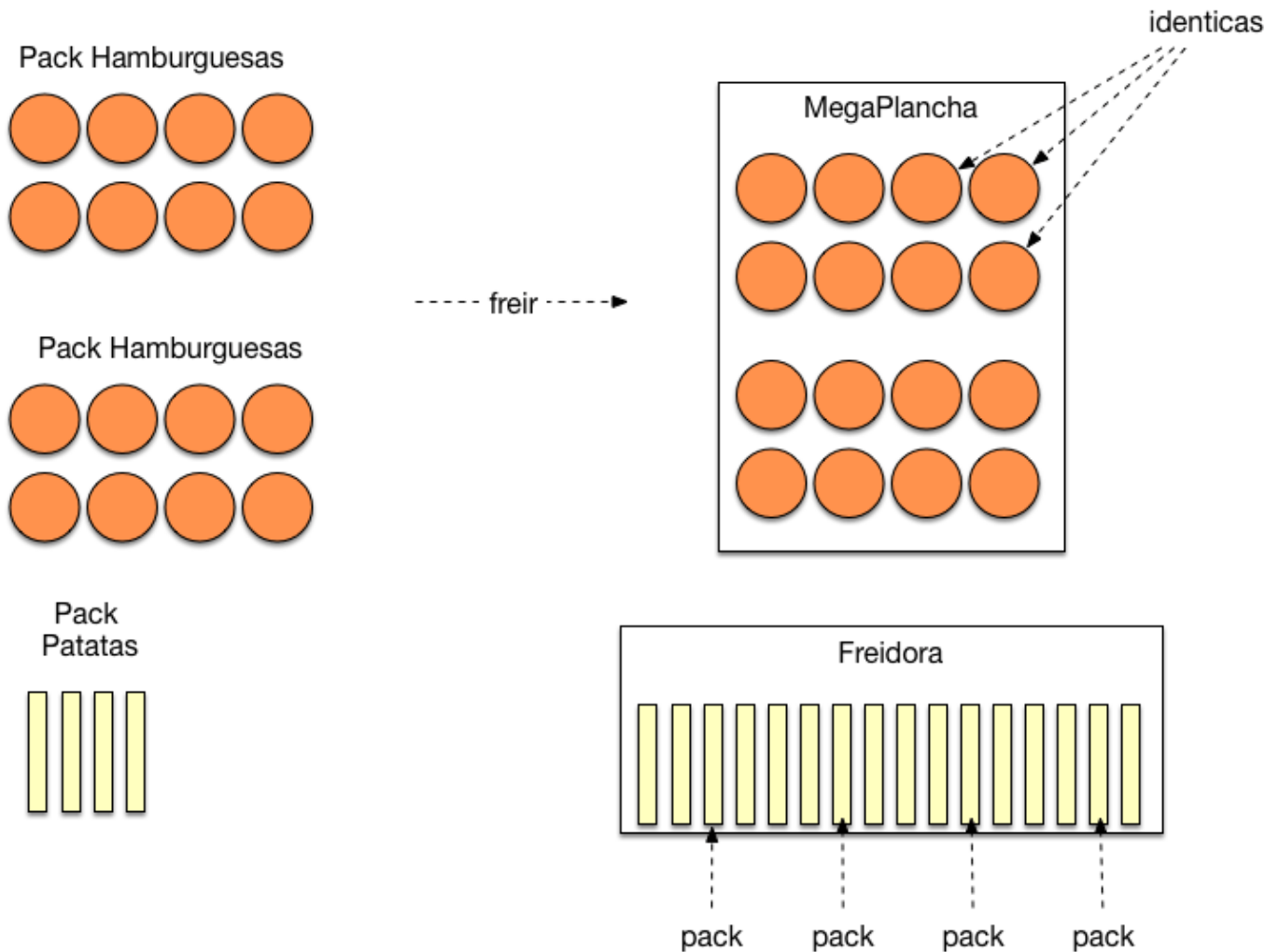
Framework vs Libreria (Libreria)

No solo eso sino que la plancha ademas admite bien hamburguesas cuadradas que están de moda. Ok todo es correcto , podemos usar la plancha y la freidora y mejorar la forma en la que cocinamos la hamburguesa. Este es el concepto de librería , un conjunto de código que nos facilita la construcción de nuestro programa un ejemplo sería jQuery , podemos hacer algo mejor y más rápido.

Framework vs Libreria (Frameworks)

Sin embargo esta es la forma en la cual la gente pro hace las hamburguesas con patatas en casa , e incluso puede ser la forma en la que un pequeño restaurante las hace. ¿ Ahora bien cómo las hace McDonalds? . Este tema es otro cantar , McDonalds es posible que use hamburguesas congeladas, todas del mismo tamaño y que las cocine por bloques. No solo eso sino que también usara las mismas patatas congeladas todas idénticas o muy similares que freirá en su freidora por packs.

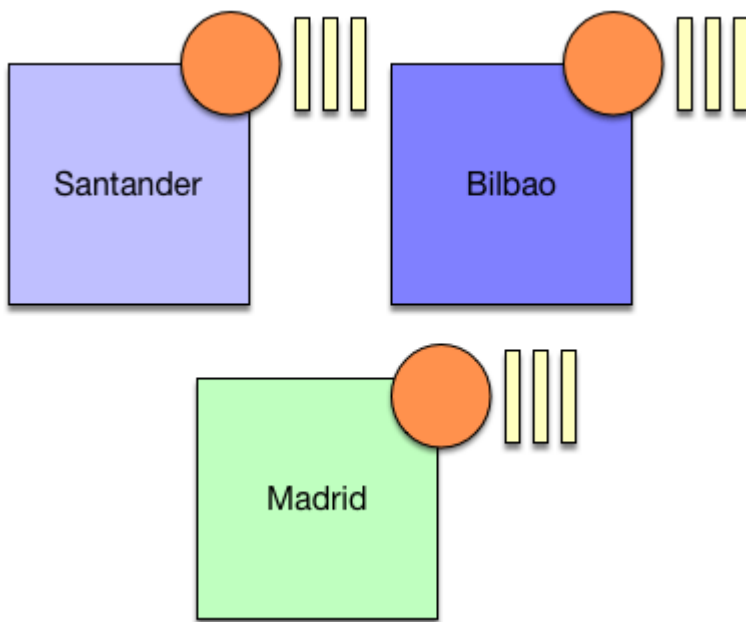
Framework vs Libreria dos conceptos importantes



Esto parece en un primer momento que se parece mucho a la solución anterior pero solo que a lo bestia. La realidad es que se parecen en poco . ¿Qué fue de nuestras hamburguesas cuadradas? . Muy sencillo McDonalds las hace solo redondas y punto. Oye y de esas patatas unas más grandes y otras más pequeñas .. tampoco existen McDonalds nos traen las patatas congeladas y todas son iguales. Oye que yo quería la hamburguesa muy hecha . Nada imposible la mega plancha las hace todas a la vez.¿En qué hemos ganado entonces? .

Pues así visto lo visto parece que hemos perdido. La realidad es un poco diferentes , hemos perdido en flexibilidad, esa es la clave, el proceso no es tan flexible. Sin embargo hemos ganado en productividad ya que en poco tiempo cocinamos mucho . Para lograrlo se asumen

varias limitaciones las hamburguesas solo son de un tipo igual que las patatas , la freidora y la megaplancha también son muy especificadas. ¿Qué más hemos logrado? Hemos logrado generar homogeneidad ya que todas las hamburguesas en todos los McDonalds nos saben igual .



Da lo mismo donde estemos ubicados. Acabamos de generar una “cultura” . Eso es lo que hace un framework , aportar productividad , define limitaciones y genera homogeneidad a la hora de trabajar dentro de una organización . Un ejemplo sería Spring o Angular.

Otros artículos relacionados

1. [Angular 5 Hello World y su funcionamiento](#)
2. [React vs Angular 2 , frameworks vs librerías](#)
3. [El patrón MVC , arquitectura cliente vs servidor](#)
4. [Framework Wikipedia](#)