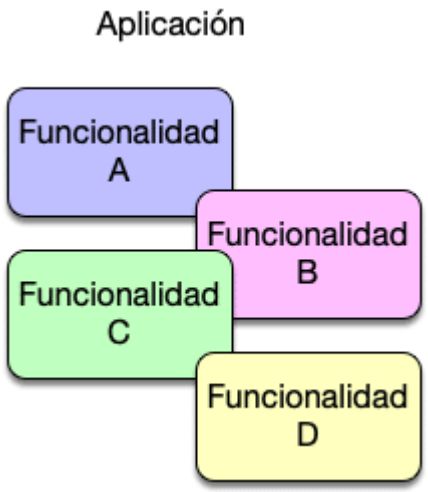
Tabla de Contenidos

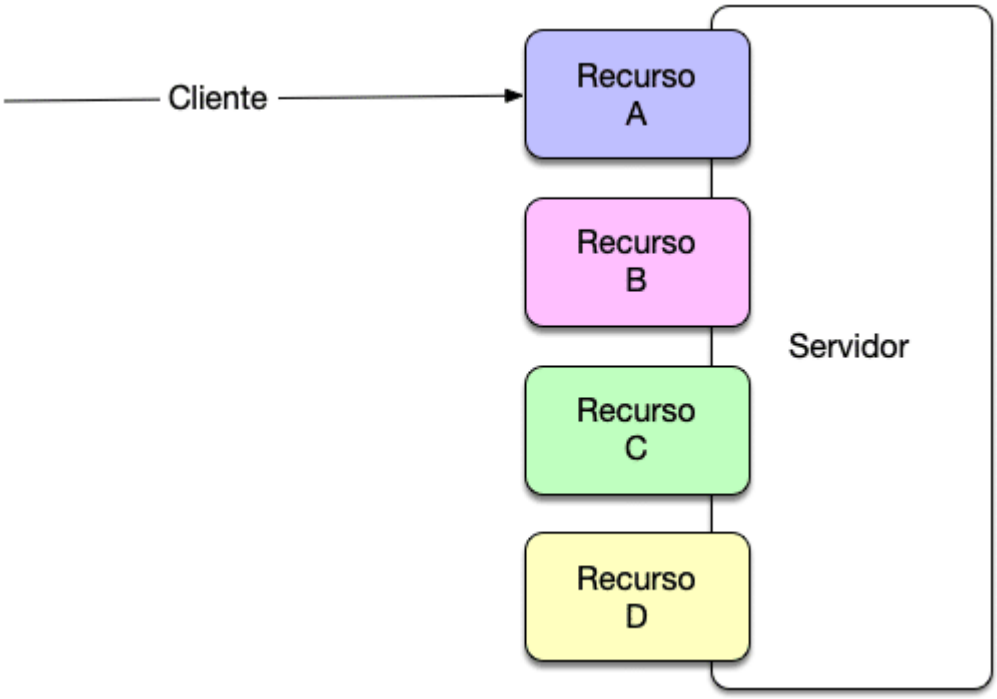
- REST API Design y Simplicidad
  - ¿Es el mejor camino?
  - Simplicidad y Negocio

El concepto de REST API Design es un concepto cuando menos “abierto”. En el sentido de que siempre podemos diseñar el API REST de la forma que deseemos o nos apetezca . La pregunta más importante en estos casos es si podemos hacer algo para simplificar el diseño o aportar sencillez. Vamos a hablar un poco de ello. Normalmente cuando queremos diseñar un API REST nos encontramos que tenemos un conjunto importante de funcionalidad que publicar y rápidamente nos queremos poner a ello y aportar capacidades a los usuarios que se conecten.



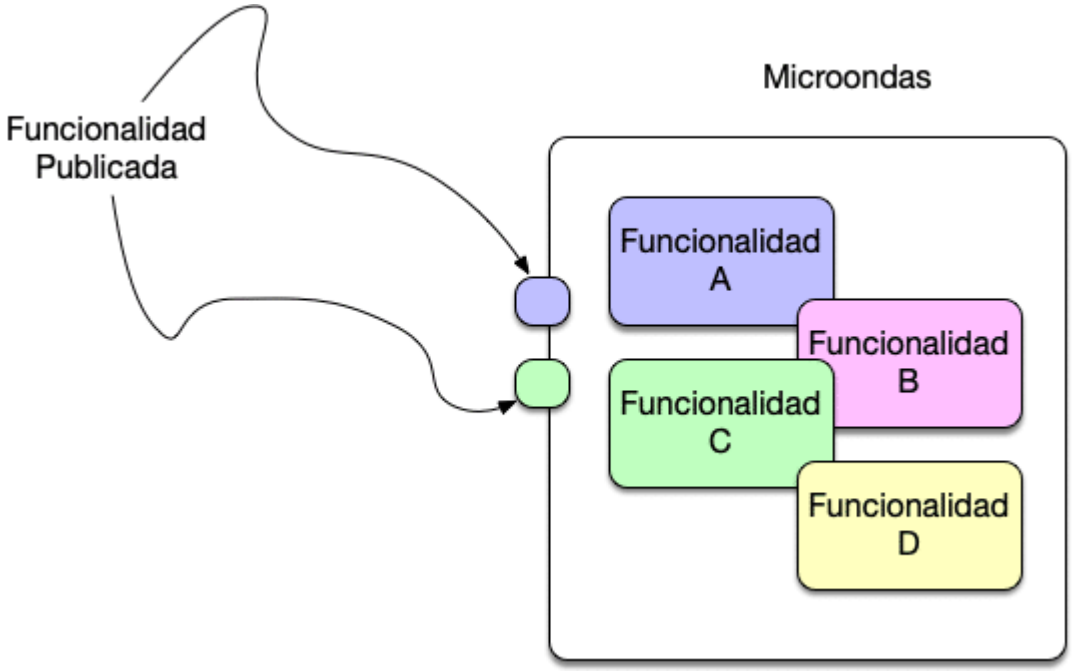
REST API Design y Simplicidad

Así pues en muchos casos los desarrolladores deciden publicar esa funcionalidad vía servicios REST de una forma no voy a decir directa pero sí muy similar a la que tienen en su código. El diagrama sería algo parecido a esto:



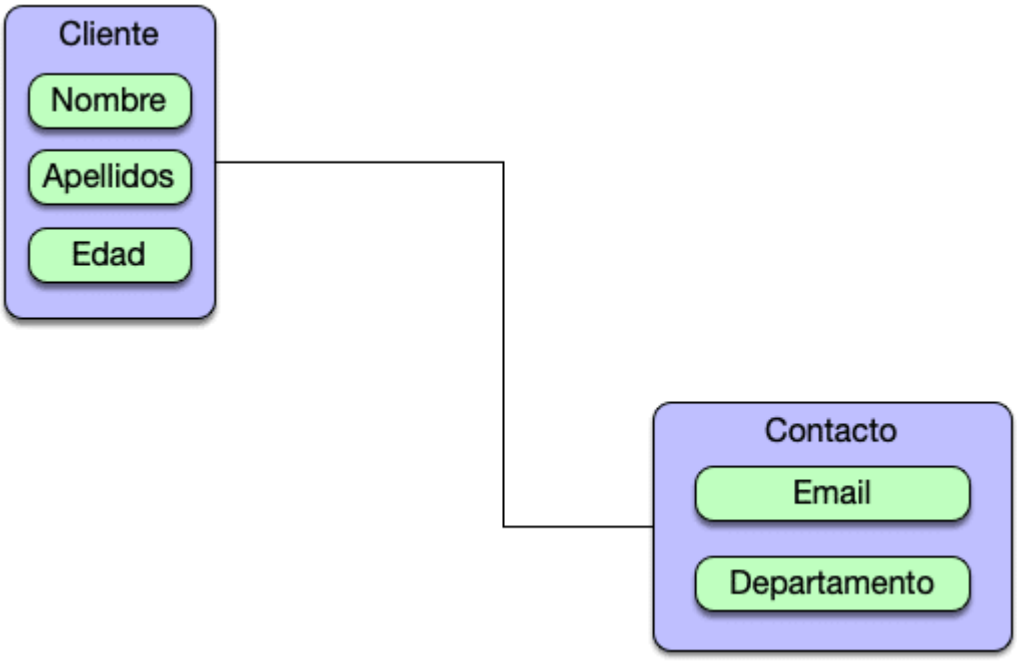
¿Es el mejor camino?

Esta es una buena pregunta . ¿Es el mejor camino en cuando a REST API Design? . La respuesta probablemente es No. Para eso nos podemos apoyar en conceptos mucho más sencillos de diseño como es el diseño por ejemplo de un Microondas. La funcionalidad que esconde tras de sí es amplia pero publica hacia fuera “muy poca” solamente aquella que considera que es vital para el cliente a la hora de sacar partido de todo lo que él puede hacer.

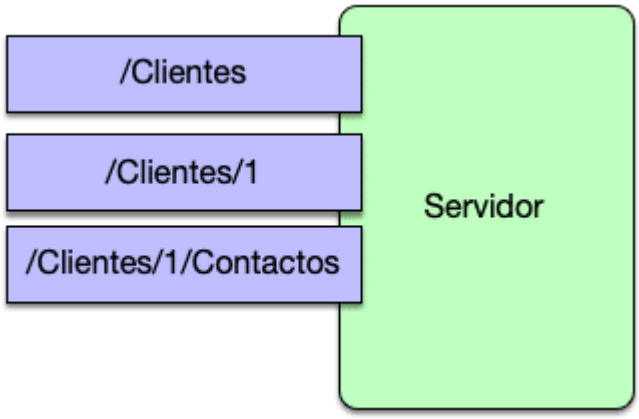


Simplicidad y Negocio

Esto nos puede parecer un poco extraño al principio pero vamos a apoyarnos en un ejemplo real para entender mejor los conceptos. Supongamos que tenemos el concepto de Cliente y se Cliente dispone de una lista de emails a los cuales le podemos enviar un correo. Probablemente la estructura de clases será :



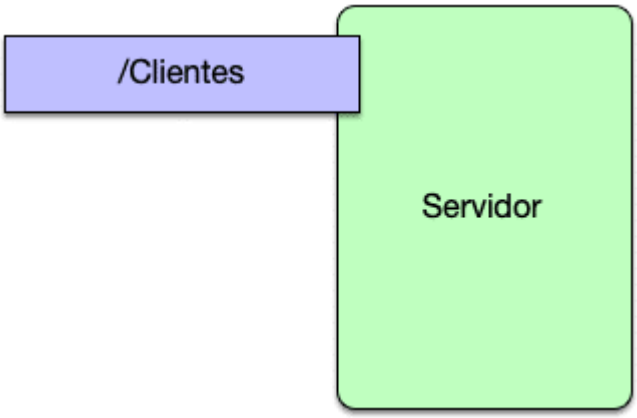
Así pues podríamos disponer de un Cliente con una lista de Contactos . Nos puede parecer interesante publicar toda esta información vía REST . Esto generaría una estructura de agregados :



Sin embargo es esta información la que necesita nuestro cliente o hubiera sido mejor en vez de publicar el Cliente publicar un ClienteDTO que incluyera el correo que consideramos el principal de este Cliente.



De esta de manera simplemente sería suficiente con publicar una única URL REST con la lista de Clientes .



Pensemos siempre en el usuario del API a la hora de abordar un REST API Design

1. Arquitecturas REST y Agregados
2. REST DTO
3. Spring REST Cors
4. REST