

Antes de empezar a pensar en el servicio, debemos responder las siguientes preguntas en orden:

- ¿Qué son las URIs? Las cosas identificadas por URIs son “recursos”. Aunque es más apropiado decir que los Recursos son identificados mediante URIs.

Si solo existe una única URI como punto de acceso, posiblemente se esté creando un protocolo RPC. En tal caso, se debe desmenuzar el problema en tipos de recursos que se quieren manipular. Dos lugares donde se debe considerar cuando se buscan los recursos potenciales son las colecciones y las interfaces de búsqueda. Una colección de recursos es en si mismo un recurso. Una interfaz de búsqueda también lo es, ya que el resultado de una búsqueda es otro conjunto de recursos.

A modo de ejemplo, supongamos un sistema de mantenimiento de una lista de contactos de empleados. En tal sistema cada usuario debería tener su propia URI con una apropiada representación. Además, la colección de recursos es otro recurso.

Por tanto, hemos identificado dos tipos de recursos, por tanto habrá dos tipos de URIs:

- Employee (Una URI por empleado).
  - AllEmployee (Listado de los empleados).
- ¿Cuál es el formato? Cuando hablamos informalmente de formato, estamos hablando de la representación. Como ya hemos comentado con anterioridad, no se puede acceder directamente al recurso, hay que obtener una representación. Esta representación puede ser un documento HTML, XML, una imagen,... dependiendo de la situación.

Para cada uno de los recursos, se tiene que decidir cual va a ser su representación. Si es posible, reutilizar formatos existentes, ayudará a incrementar la oportunidad de que nuestro sistema se componga con otros.

Continuando con nuestro ejemplo, el formato de representación va a ser XML, ya que es el principal formato para intercambio de información. El formato del empleado podría ser el siguiente:

```
<employee xmlns='HTTP://example.org/my-example-ns/'>
  <name>Full name goes here.</name>
  <title>Persons title goes here.</title>
  <phone-number>Phone number goes here.</phone-number>
</employee>
```

Para el listado de empleados podríamos tomar este otro:

```
<employee-list xmlns='HTTP://example.org/my-example-ns/'>
  <employee-ref href="URI of the first employee"/>
  Full name of the first employee goes here.</employee>
```