Documentos

Tiempo estimado: 15min

En esta lección, vamos a prestar especial atención a los documentos. Cada documento representa una unidad de datos a indexar.

Abrimos la lección introduciendo el concepto de documento. Junto a los índices, las piezas claves de almacenamiento y búsqueda. Y a continuación, describimos las operaciones de inserción, actualización y supresión de documentos de los índices de búsqueda.

Al finalizar la lección, el estudiante sabrá:

- Qué es un documento.
- Qué es y cuándo se realiza el análisis de los documentos indexados.
- Cómo insertar, actualizar y suprimir documentos indexados.

Introducción

Un documento (*document*) es la unidad básica de indexación en un motor de búsqueda. En RediSearch, un documento es un par clave-valor de tipo *hash*. Sólo se indexan aquellos documentos (o pares) que se registran explícitamente en uno o más índices. Y en caso de estarlo, sólo se indexarán los campos indicados en los esquemas de los índices.

Análisis de los documentos

Groso modo, podemos ver un índice como una tabla donde cada entrada representa un término, la cual mantiene una referencia a los documentos en los que aparece. Esta tabla la actualiza automáticamente el analizador (*analyzer*), el componente del motor que estudia el contenido de un documento al insertarse, concretamente los campos del esquema.

Por un lado, extrae los términos que aparecen, donde un término (*term*) representa un elemento o palabra buscable por los usuarios.

Por otro lado, realiza una reducción (*stemming*), operación mediante la cual se determina el término base o raíz de cada uno de ellos. Así para cada palabra en plural también se añadirá su término singular para facilitar la búsqueda a partir de términos en singular. Puede hacer algo similar con las conjugaciones de verbos y las palabras en femenino y masculino.

Lo importante es recordar que esta operación se hace durante la inserción de los documentos, no con cada consulta de búsqueda. Así las búsquedas serán más rápidas y fáciles para el motor.

A la hora de indexar un documento, es muy importante conocer:

- El idioma en que se encuentra su texto.
- La puntuación de usuario.

Idioma del documento

Recordemos que el objetivo de un índice es descomponer los documentos y extraer los términos que aparecen en ellos. Esta descomposición puede depender de cada idioma. Por lo que hay que añadir el idioma en que se encuentra el documento para que el analizador haga bien su trabajo. Si no se indica, tras cada inserción, se usará el predeterminado, que podría no coincidir con el idioma real del documento.

Puntuación del documento

En RediSearch, cada documento tiene una puntuación (*score*), un valor que le otorga mayor importancia que el resto. Esta puntuación la fija el propio usuario. Y hay que fijarlo en el momento de su inserción. Si

para el usuario todos los documentos son iguales, utilice siempre el mismo valor.

Generalmente, se utiliza la puntuación para dar mayor importancia a los documentos más recientes con respecto a otros que representan entidades anticuadas u obsoletas. También es útil para marcar documentos en listas tops: el top 10, el top 20, etc. Así el motor no recorrerá todos los documentos.

Inserción de documentos

Es muy importante tener claro que un documento es un par clave-valor de tipo *array* asociativo. El cual puede encontrarse ya en la base de datos o bien se puede añadir al índice y a la base de datos mediante una única operación.

Inserción de documento existente

Para añadir un documento ya existente en la base de datos a un índice, se usa el comando FT.ADDHASH, cuya sintaxis es como se muestra a continuación:

```
FT.ADDHASH índice clave puntuación
FT.ADDHASH índice clave puntuación LANGUAGE idioma
```

Como parece lógico, hay que indicar, por un lado, el nombre del índice al que añadir el documento y, por otra parte, su clave. Además, hay que indicar la puntuación de usuario (*score*), cuyo valor debe encontrarse entre 0.0 y 1.0. Muy importante es el idioma, que en caso de omisión será english. La lista de idiomas actual es la siguiente: arabic, dutch, english, finnish, french, german, hungarian, italian, norwegian, portuguese, romanian, russian, spanish, swedish, tamil y turkish.

Veamos un ejemplo:

```
127.0.0.1:6379> HMSET band:gin-blossoms:en name "Gin Blossoms" origin US
OK
127.0.0.1:6379> FT.ADDHASH bands band:gin-blossoms:en 1 LANGUAGE english
OK
127.0.0.1:6379>
```

Inserción de documentos inexistentes

Si el documento no existe todavía, podemos realizar su añadidura a la base de datos y al índice mediante una única operación. El comando FT.ADD es el encargado de esta función:

```
FT.ADD índice clave puntuación FIELDS campos
FT.ADD índice clave puntuación LANGUAGE idioma FIELDS campos
```

Ejemplo:

```
127.0.0.1:6379> HGETALL band:phoenix:en
(empty list or set)
127.0.0.1:6379> FT.ADD bands band:phoenix:en 1 LANGUAGE english FIELDS name Phoenix origin
FR
OK
127.0.0.1:6379> HGETALL band:phoenix:en
1) "name"
2) "Phoenix"
3) "origin"
4) "FR"
127.0.0.1:6379>
```

Mediante la cláusula FIELDS, se indica los campos del documento. Tanto aquellos que se indexarán como los que no.

Documentos sólo indexados

Aunque hemos dicho que un índice requiere que sus documentos se encuentren en la base de datos, esto no es del todo cierto. Ha sido una pequeña mentira pedagógica. Tengamos en cuenta que el índice mantiene sus propios datos de los documentos. No mantiene una copia de los documentos, pero sí los términos. Y la clave del documento.

Cuando añadimos un documento de la base de datos al índice, las consultas de búsqueda podrán acceder a él y devolverlo. Es lo más habitual.

Pero no tiene por qué ser así siempre. Podemos añadir un documento a un índice sin necesidad de

crearle un par clave-valor asociado en la base de datos. Habrá que indicar su clave, a pesar de todo. Pero al no estar en la base de datos, las consultas no devolverán más que la clave, no así su contenido. En estos casos, los documentos deben añadirse con el comando FT.ADD y el parámetro NOSAVE.

Vamos a ver ambos casos mediante unos sencillos ejemplos. En primer lugar, un documento con copia mantenida en la base de datos:

```
127.0.0.1:6379> HGETALL band:attic-lights:en
(empty list or set)
127.0.0.1:6379> FT.ADD bands band:attic-lights:en 1 LANGUAGE english FIELDS name "Attic
Lights" origin UK
OK
127.0.0.1:6379> HGETALL band:attic-lights:en
1) "name"
2) "Attic Lights"
3) "origin"
4) "UK"
127.0.0.1:6379> FT.SEARCH bands attic
1) (integer) 1
2) "band:attic-lights:en"
3) 1) "name"
   2) "Attic Lights"
   3) "origin"
   4) "UK"
127.0.0.1:6379>
```

Observe que el resultado de la búsqueda devuelve el contenido completo del documento.

Ahora, veamos un documento sin homólogo en la base de datos:

```
127.0.0.1:6379> HGETALL band:neuman:en
(empty list or set)
127.0.0.1:6379> FT.ADD bands band:neuman:en 1 NOSAVE LANGUAGE english FIELDS name Neuman
origin ES
OK
127.0.0.1:6379> HGETALL band:neuman:en
(empty list or set)
127.0.0.1:6379> FT.SEARCH bands neuman
1) (integer) 1
2) "band:neuman:en"
3) (empty list or set)
127.0.0.1:6379>
```

En el resultado, sólo se devuelve la clave del documento, sin su contenido. Aunque no se añada el documento a la base de datos, sólo al índice, es necesario indicar siempre su clave.

Actualización de documentos

Supongamos que modificamos un documento indexado, ¿se actualizan automáticamente los índices? Tristemente, tenemos que decir que *no*. En estos casos, hay que realizar una inserción con reemplazo mediante el comando FT.ADDHASH. Su sintaxis es la misma, pero hay que añadir REPLACE al final.

Veamos un ejemplo ilustrativo completo que muestra como cuando realizamos una actualización de documento mediante HSET, no se actualiza también el índice:

```
127.0.0.1:6379> FT.ADD bands band:dinosaur-jr:en 1 LANGUAGE english FIELDS name "Dinosaur
Jr." origin US
OK
127.0.0.1:6379> FT.SEARCH bands US
1) (integer) 1
2) "band:dinosaur-jr:en"
3) 1) "name"
   2) "Dinosaur Jr."
   3) "origin"
   4) "US"
127.0.0.1:6379> FT.SEARCH bands USA
1) (integer) 0
127.0.0.1:6379> HSET band:dinosaur-jr:en origin USA
(integer) 0
127.0.0.1:6379> FT.SEARCH bands US
```



```
1) (integer) 1
2) "band:dinosaur-jr:en"
3) 1) "name"
   2) "Dinosaur Jr."
   3) "origin"
   4) "USA"
127.0.0.1:6379> FT.SEARCH bands USA
1) (integer) 0
127.0.0.1:6379> FT.ADDHASH bands band:dinosaur-jr:en 1 LANGUAGE english REPLACE
OK
127.0.0.1:6379> FT.SEARCH bands US
1) (integer) 0
127.0.0.1:6379> FT.SEARCH bands USA
1) (integer) 1
2) "band:dinosaur-jr:en"
3) 1) "name"
   2) "Dinosaur Jr."
   3) "origin"
   4) "USA"
127.0.0.1:6379>
```

El REPLACE transforma la inserción en un *upsert*, esto es, inserta si no existe o bien actualiza si ya existe. También es posible realizar un *upsert* con FT.ADD. Use también REPLACE pero indíquelo antes de FIELDS.

Supresión de documentos

Para suprimir un documento de un índice, hay que utilizar el comando FT.DEL:

```
FT.DEL índice clave
```

Es importante tener muy claras las cosas. Si se suprime un par clave-valor indexado de la base de datos mediante el comando DEL de Redis, no se suprimirá también sus entradas de los índices. Y si se suprime un documento de un índice, lo que se está haciendo es suprimir sus entradas del índice. Nada más. No se suprime también el par clave-valor.

Se recomienda utilizar *scripts* Lua para suprimir claves o documentos indexados. Si añade nuevos índices, actualice también el *script*. No es plan de tener entradas en los índices de búsqueda que hagan referencia a documentos borrados.

