Shell e interfaz web de ArangoDB

Tiempo estimado: 10min

Antes de adentrarnos en la parte puramente de datos, hay que detenerse primero en dos herramientas muy utilizadas en ArangoDB, su *shell* y su interfaz web.

Comenzamos la lección presentando brevemente arangosh y la interfaz web. A continuación, se introduce el *shell* de ArangoDB. Y finalmente, se muestra cómo acceder a la interfaz web de una instancia de bases de datos.

Al finalizar la lección, el estudiante sabrá:

- Cómo usar arangosh.
- Cómo acceder a la interfaz web de ArangoDB.

Introducción

Todo sistema de bases de datos que se precie debe tener buenas herramientas con las que acceder *ad-hoc* a los datos. En el caso de ArangoDB, se dispone de un *shell* y una interfaz web. El *shell* se utiliza para acceder mediante línea de comandos, usando JavaScript. Mientras que la interfaz web es una interfaz gráfica de administración de una instancia de bases de datos. Ambas son muy usadas.

Shell de ArangoDB

El shell de ArangoDB (*ArangoDB shell*) es un programa o herramienta de línea de comandos con la que ejecutar o consultar instancias de bases de datos. Concretamente, es un *shell* de JavaScript, por lo que las consultas se realizan mediante este lenguaje de programación.

Lo implementa el comando arangosh. Este programa se conecta a una instancia usando una cuenta de usuario y permite consultar una determinada base de datos de manera interactiva.

Para obtener su lista de opciones, se utiliza la opción --help:

```
arangosh --help
```

Información de conexión

Para poder conectar a una instancia, hay que indicar la información de un extremo o punto de conexión donde escucha la instancia. Para ello, se usa la opción --server.endpoint. De manera predeterminada, se usa tcp://127.0.0.1:8529.

Por otra parte, hay que indicar la base de datos de la instancia a la que conectar. Su nombre se indica mediante la opción --server.database. Si no se especifica una explícitamente, se intentará conectar a la base de datos de sistema _system.

Además, también hay que indicar la cuenta de usuario con la que conectar. El nombre se indica mediante la opción --server.username y la contraseña mediante --server.password. Si no se especifica nombre de usuario, el *shell* usará root.

He aquí un ejemplo ilustrativo:

```
arangosh --server.endpoint tcp://localhost:8529 --server.username root --server.database
test
```

Prompt

El prompt es un texto mediante el cual el *shell* indica que está esperando que le proporcionemos una

operación a ejecutar. El formato actual es:

```
servidor:puerto@baseDeDatos>
```

Como,. por ejemplo:

```
127.0.0.1:8529@_system>
```

Se puede indicar un *prompt* particular mediante la opción --console.prompt. Ejemplo:

```
$ arangosh --console.prompt "esto es el prompt> "
esto es el prompt>
```

Archivo .arangosh.rc

El archivo .arangosh.rc, ubicado en el directorio *home* del usuario, es un archivo JavaScript que ejecuta el *shell* automáticamente al arrancar. Se utiliza principalmente para crear variables o funciones personalizadas utilizables durante la sesión de *shell*. Cualquier nueva función o variable global que se necesite añadir al contexto de ejecución global del motor de JavaScript del *shell*, se puede añadir mediante el objeto global. Ejemplo:

```
global.fn = function() {
  …
};
```

Definición de variables

Recordemos que arangosh es un *shell* de JavaScript para trabajar con una instancia de bases de datos ArangoDB. Para añadir una variable al contexto de ejecución del *shell*, se puede utilizar la siguiente sintaxis:

```
nombre = valor
```

Ejemplo:

```
127.0.0.1:8529@_system> x = "ejemplo"
ejemplo
127.0.0.1:8529@_system>
```

Definición de funciones

Para definir funciones, no hay más que utilizar la sentencia function de JavaScript. Veamos un ejemplo ilustrativo:

```
127.0.0.1:8529@_system> function sum(x, y) { return x + y; }
127.0.0.1:8529@_system> sum(1, 2)
3
127.0.0.1:8529@_system>
```

Objeto predefinido db

En ArangoDB, toda conexión a la instancia debe hacerse a una de sus bases de datos. El *shell* proporciona el objeto predefinido db para identificar la base de datos activa o actual. Para obtener una pequeña ayuda del objeto db, se puede utilizar su método _help():

```
127.0.0.1:8529@_system> db._help()
```

Ayuda del shell

Si deseamos obtener una breve ayuda, usar el comando help:

```
127.0.0.1:8529@_system> help
```

Terminación del shell

Para terminar una sesión de *shell*, se usa *dos* Ctrl+C o el comando exit. He aquí un ejemplo:

```
127.0.0.1:8529@_system> exit
$
```

Cancelación de línea

Si lo que se desea es cancelar la línea actual introducida en el *prompt*, se usa *un* Ctrl+C.

Limpieza de la consola

Para limpiar la consola, utilice la función clear().

Presentación de mensajes

Para presentar un mensaje al usuario, utilice la función print():

```
function print(msg)
```

Parámetro	Tipo de datos	Descripción
msg	string	El mensaje a mostrar.

Ejecución de scripts.

arangosh puede ejecutar *scripts* de *shell* escritos, recordemos, en JavaScript. No hay más que indicar el archivo mediante la opción --javascript.execute. Ejemplo ilustrativo:

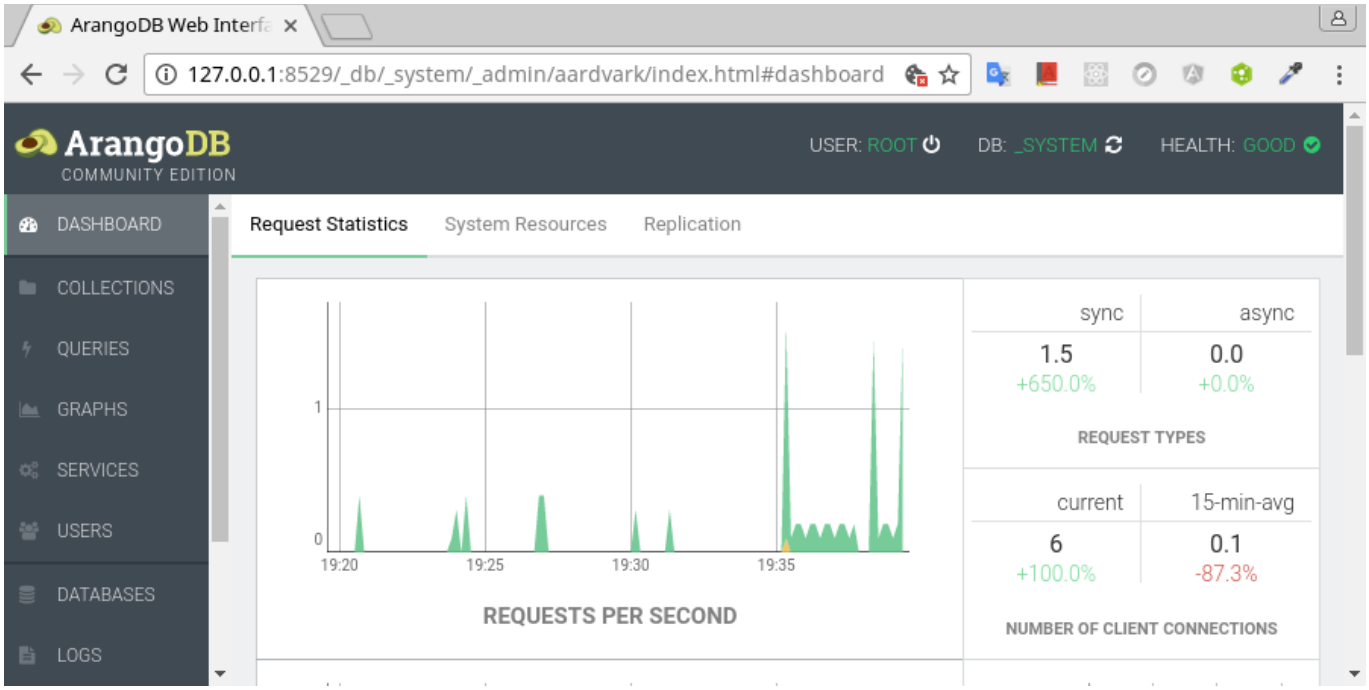
```
$ cat listado.js
print("Listado de bases de datos");
print("------------------------");

for (let name of db._databases()) {
  print(name);
}
$ arangosh --javascript.execute listado.js
Please specify a password:
Listado de bases de datos
------------------------
_system
test
$
```

También es posible utilizar la opción --javascript.execute-string para ejecutar código JavaScript proporcionado mediante la línea de comandos. Ejemplo:

```
arangosh --javascript.execute-string "print(db._databases())"
```

Interfaz web de ArangoDB

La interfaz web de ArangoDB (*ArangoDB web inteface*) es una consola gráfica para la administración de una instancia de bases de datos. Basta con abrir un navegador y conectar a un punto de conexión en el que escuche la instancia.



Mediante esta aplicación, se puede crear bases de datos y colecciones, registrar servicios Foxx, ejecutar consultas de datos, ver el estado de la instancia y mucho más.