

# Taller Fiware - Smart Campus - Guía de instalación de componentes

---

Encuentro ANUIES-TIC 2018

## Introducción

---

(Los tutoriales para empezar a trabajar con Fiware se encuentran en: <https://fiware-tutorials.readthedocs.io/en/latest/>)

Tenemos tres opciones para trabajar con los componentes de Fiware:

1. Instalándolos localmente en máquinas virtuales con ayuda de contenedores.
2. Instalándolos en un servidor
3. Accediendo a instancias de los componentes en el ambiente de desarrollo Fiware Lab Cloud

En este documento se muestra cómo instalar localment los componentes de Fiware.

## 1. Componentes de Fiware en equipo local

---

### Requerimientos de equipo

- Memoria 8 GB o 12 GB si es Windows (para la virtualización)
- Espacio en disco duro 16 GB (para las máquinas virtuales)

### Requerimientos de software

- [Docker y Docker compose](#)
- Un cliente REST, por ejemplo, [Insomnia](#)

### 1.1 Instalación de Docker

En esta guía se muestra la instalación de docker en un ambiente Windows 10 siguiendo las instrucciones de [esta página](#). La guía instalación para MAC se encuentra [aquí](#) y la de Linux, [aquí](#).

Si su computadora no cumple los requerimientos para instalar docker directamente, instalamos [su toolbox](#).

Una vez instalado el toolbox, abrimos su terminal y verificamos que se haya instalado correctamente ejecutando el comando `docker --version`:

```
> docker --version
Docker version 18.03.0-ce, build 0520e24302
```

La primera vez que se instala docker, es común verificar que todo funciona correctamente instalando la imagen "[Hello World](#)" del repositorio de contenedores Docker Hub, y ejecutándolo:

```
> docker run hello-world
```

```
docker : Unable to find image 'hello-world:latest' locally
...

latest:
Pulling from library/hello-world
ca4f61b1923c:
Pulling fs layer
ca4f61b1923c:
Download complete
ca4f61b1923c:
Pull complete
Digest: sha256:97ce6fa4b6cdc0790cda65fe7290b74cfebd9fa0c9b8c38e979330d547d22ce1
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.
...
```

## 1.3 Instalación de los contenedores para Orion Context Broker

Como se verá en la práctica, el componente central de Fiware, es el [Orion Context Broker](#) (OCB), que utiliza una base de datos de software libre [MongoDB](#) para asegurar la persistencia de los datos que recibe e información adicional. Por ello, debemos descargar dos imágenes de Docker y crear la interconexión entre ellos.

Empezamos por recuperar las imágenes del Docker Hub y definir una red para interconectar los contenedores. No utilizaremos un archivo de configuración \*.yaml para familiarizarnos con algunos de los comandos de Docker.

- Desde una terminal (o desde la terminal del toolbox si así lo instaló), ejecute:

```
$ docker pull mongo:3.6
$ docker pull fiware/orion
$ docker network create fiware_default
```

- El contenedor Docker con MongoDB se inicializa y se conecta a la red con los siguientes comandos:

```
$ docker run -d --name=mongo-db --network=fiware_default --expose=27017 mongo:3.6 --
bind_ip_all --smallfiles
```

- El OCB se inicializa y se conecta a la red así:

```
$ docker run -d --name=fiware-orion -h orion --network=fiware_default -p 1026:1026
fiware/orion -dbhost mongo-db
```

Si todo se ha instalado correctamente, deberá haber generado algunos mensajes como los siguientes:

```

Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

PS C:\WINDOWS\system32> docker --version
Docker version 18.03.0-ce, build 0520e24302
PS C:\WINDOWS\system32> docker image ls
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
PS C:\WINDOWS\system32> docker pull mongo:3.6
3.6: Pulling from library/mongo
802b00ed6f79: Pull complete
0ea0d89f82e8: Pull complete
d97c58720bef: Pull complete
be6bb1bae15f: Pull complete
e23dff40214a: Pull complete
8ccae196e47f: Pull complete
3b9e62932ff1: Pull complete
ec4bea918171: Pull complete
3faf18351b5d: Pull complete
2450c47236e6: Pull complete
Digest: sha256:535d886eccab2ed8efd2ca95b5185e2b0489aeb26b07eff2321f08b11b5c2ef1
Status: Downloaded newer image for mongo:3.6
PS C:\WINDOWS\system32> docker pull fiware/orion
Using default tag: latest
latest: Pulling from fiware/orion
18b8eb7e7f01: Pull complete
2ccf1caf547c: Pull complete
Digest: sha256:3b315c3d4640a446d16bf6524fc967c0fcc14b52ecda02bf21d8979f76f6e3a3
Status: Downloaded newer image for fiware/orion:latest
PS C:\WINDOWS\system32> docker network create fiware_default
8317197ba8ebf8239a86062ab764feaa3d8714a9334764cb96df396fea5093b4
PS C:\WINDOWS\system32> docker run -d --name=mongo-db --network=fiware_default --expose=27017 mongo:3.6 --bind_ip_all --smallfiles
7770233db78eb5b2e84887510fe87c545f27c771e6590ec3b5118906f728ae31
PS C:\WINDOWS\system32> docker run -d --name=fiware-orion -h orion --network=fiware_default -p 1026:1026 fiware/orion -dbhost mongo-db
b06679c70923bfe4555a5cb4de36eb2c6a53d74834a1e027a6063cbf781c97b
PS C:\WINDOWS\system32>

```

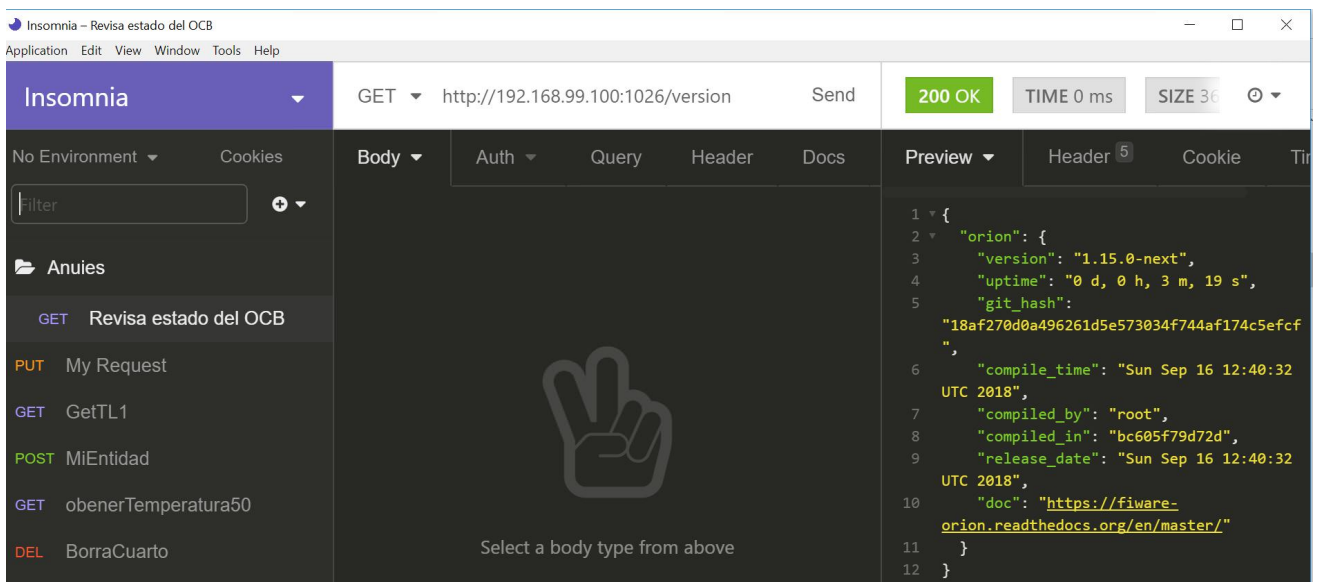
**Nota:** Si desea borrar todo e iniciar de nuevo, puede ejecutar los siguientes comandos:

```

$ docker stop fiware-orion docker
$ rm fiware-orion
$ docker stop mongo-db
$ docker rm mongo-db
$ docker network rm fiware_default

```

- Verificamos que el servidor está activo consultando la versión del OCB en el puerto 1026, que es donde espera las solicitudes de servicio. Esto se puede hacer ejecutando el comando `curl -X GET 'http://localhost:1026/version'`, invocando el URL en un navegador Web, o con nuestro cliente REST:



Insomnia - Revisa estado del OCB

Application Edit View Window Tools Help

Insomnia GET http://192.168.99.100:1026/version Send 200 OK TIME 0 ms SIZE 36

No Environment Cookies Body Auth Query Header Docs Preview Header 5 Cookie

Filter

Anuies

GET Revisa estado del OCB

PUT My Request

GET GetTL1

POST MiEntidad

GET obenerTemperatura50

DEL BorraCuarto

Select a body type from above

```

1 {
2   "orion": {
3     "version": "1.15.0-next",
4     "uptime": "0 d, 0 h, 3 m, 19 s",
5     "git_hash":
6     "18af270d0a496261d5e573034f744af174c5efcf",
7     "compile_time": "Sun Sep 16 12:40:32
8     UTC 2018",
9     "compiled_by": "root",
10    "compiled_in": "bc605f79d72d",
11    "release_date": "Sun Sep 16 12:40:32
12    UTC 2018",
13    "doc": "https://fiware-
14    orion.readthedocs.org/en/master/"
15  }
16 }

```

Si todo está correcto, obtendrá una respuesta en el panel del extremo derecho de Insomnia:

```
{
  "orion": {
    "version": "1.15.0-next",
    "uptime": "0 d, 0 h, 3 m, 21 s",
    "git_hash": "e2ff1a8d9515ade24cf8d4b90d27af7a616c7725",
    "compile_time": "Wed Apr 4 19:08:02 UTC 2018",
    "compiled_by": "root",
    "compiled_in": "2f4a69bdc191",
    "release_date": "Wed Apr 4 19:08:02 UTC 2018",
    "doc": "https://fiware-orion.readthedocs.org/en/master/"
  }
}
```

- **Nota:** En Windows, quizás deba utilizar la dirección privada que reportó docker al instalarse, por ejemplo: <http://192.168.99.100:1026/version>