

Laboratorio 1 – RDF

Objetivos:

1. Entender el modelo RDF.
2. Ser capaz de crear archivos RDF manualmente.
3. Ser capaz de gestionar una Triple Store.

Recursos necesarios:

- Ubuntu (Usuario/password: lsi/lsi) o servidor Google Cloud.

1.- Introducción

RDF (Resource Description Framework) es un estándar para modelar datos en forma de grafos en el contexto de la Web. En este laboratorio aprenderéis a crear datos en RDF y a cargarlos en una Triple Store.

2.- Instalación de GraphDB

GraphDB¹, de la empresa OntoText², es una Triple Store de alta calidad que incluye razonamiento automático, conectores para servicios como Lucene y Kafka, e incluso virtualización de bases de datos relacionales (Ontop).

GraphDB³ se puede instalar tanto en la instancia remota Google Cloud como en local (En ambos casos las instrucciones que siguen asumen un Sistema Operativo Ubuntu 22.04). Se recomienda usarlo en local pero probar también la instalación en Google Cloud, ya que el examen se hará sobre la instancia Google Cloud.

- Descargar archivo **graphdb-10.8.2-dist.zip** de eGela. Para instalarlo en Google Cloud hay que copiarlo a la instancia remota: **\$ scp graphdb-10.8.2-dist.zip IP_INSTANCIA_GOOGLE_CLOUD:/home/usuario**
- Instalar dependencias:
 - **\$ sudo apt update**
 - **\$ sudo apt install default-jdk**
 - **\$ sudo apt install unzip**
- Descomprimir archivo y ejecutar⁴:

1 <https://www.ontotext.com/products/graphdb/>

2 <https://www.ontotext.com/>

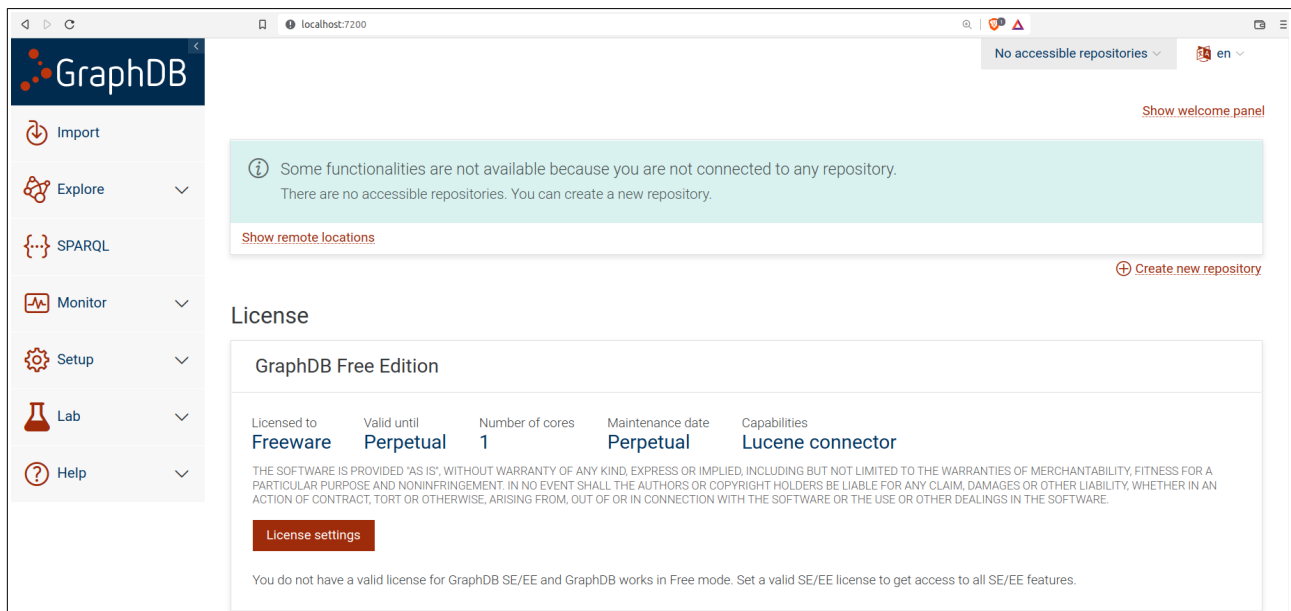
3 <https://graphdb.ontotext.com/documentation/10.8/graphdb-standalone-server.html#>

4 Para poder ejecutar más fácilmente, puedes poner el Path de Graphdb en la variable **\$PATH** del archivo **.bash_profile**

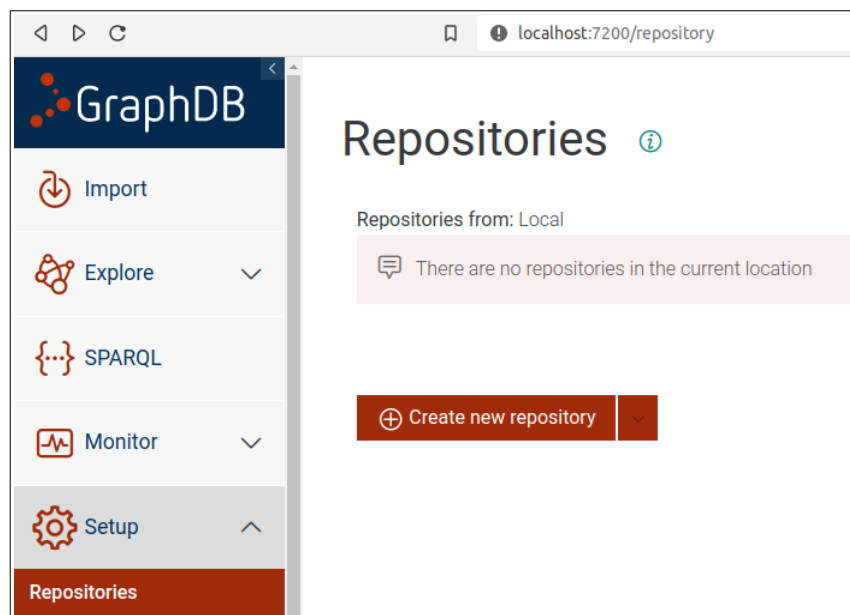
Administración de Bases de Datos 2024/2025

- \$ **unzip graphdb-10.8.2-dist.zip**
- \$ **cd graphdb-10.8.2-dist/bin/**
- \$ **./graphdb**

En la dirección <http://localhost:7200> o http://IP_INSTANCIA_GOOGLE_CLOUD:7200 debería aparecer la siguiente pantalla⁵:



Para poder trabajar con GraphDB hay que crear un repositorio mediante *Setup ; Repositories ; Create new repository*:



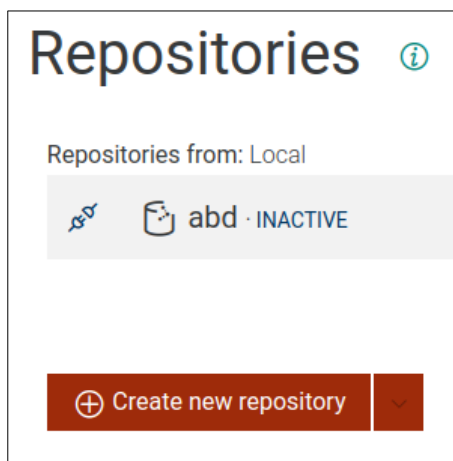
⁵ Si estáis ejecutando GraphDB en Google Cloud tendréis que abrir el puerto 7200 en el firewall. *Menu Burger* (Tres líneas horizontales); *Red de VPC*; *Firewall*: añadir regla para permitir tráfico en el puerto 7200 desde 0.0.0.0/0, con el tag "http-server".

Administración de Bases de Datos 2024/2025

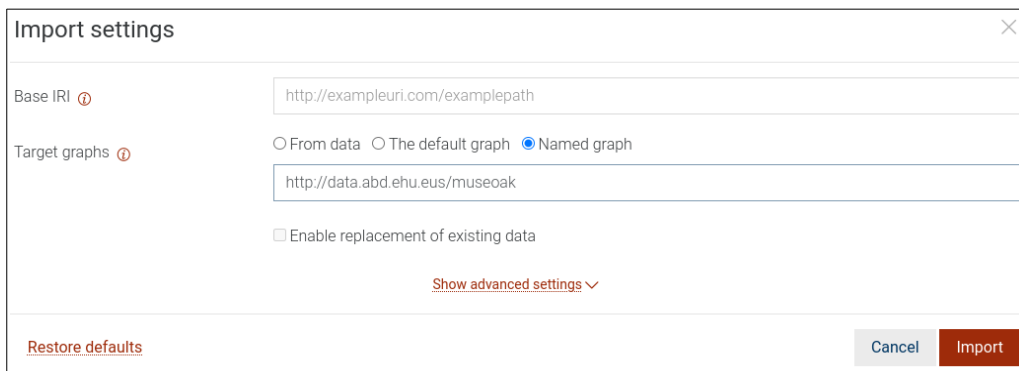
Hay tres tipos de repositorios en GraphDB:

- GraphDB repository: el repositorio local normal.
- Ontop Virtual SPARQL: un endpoint SPARQL virtual (Sin datos) que traduce las consultas SPARQL a consultas SQL, pudiendo conectar así bases de datos relacionales y consultarlas como si fuesen RDF.
- FedX virtual SPARQL: federa diferentes endpoints SPARQL bajo uno solo.

Elegir **GraphDB repository** y crear un repositorio con todas las opciones por defecto. Una vez creado el repositorio, conectarlo pinchando en el enchufe de la izquierda:

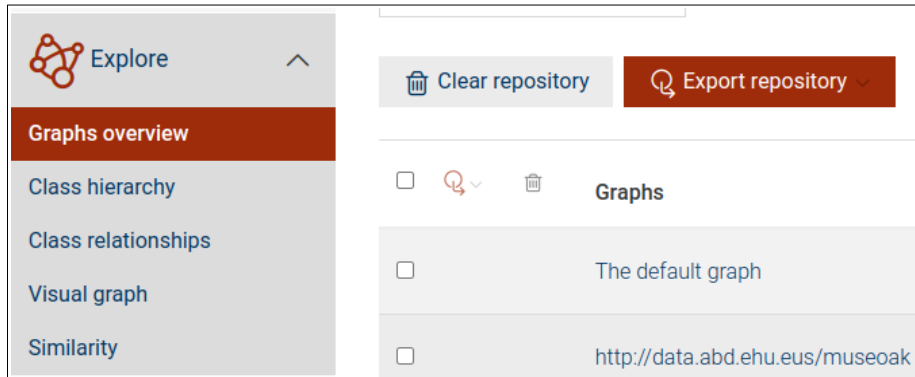


Para cargar el RDF, ir a la pestaña **Import** y ahí se puede cargar RDF eligiendo un archivo, pegándolo en un formulario, desde una URL, o desde archivos del servidor. Es importante cargar los datos en un Named Graph para organizar la información adecuadamente. La URI del Named Graph puede ser cualquiera:



Administración de Bases de Datos 2024/2025

Si los datos han sido cargados adecuadamente, en *Explore* ; *Graphs overview* debería aparecer la URI del Named Graph:



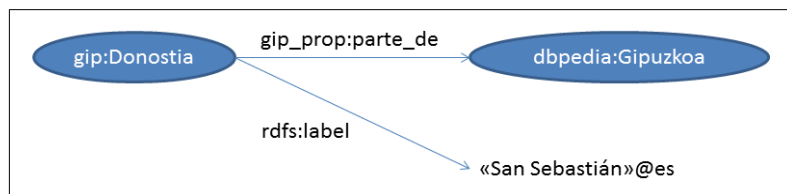
Pinchando en la URI del Named Graph deberían aparecer los triples RDF recién cargados. A través de la opción **Visual Graph** se pueden ver los triples en forma de grafo.

Para apagar GraphDB, **Ctrl-c** en la terminal en la que se ejecuta. Para que siga ejecutándose aun habiendo salido de la terminal hay dos opciones:

- **\$ nohup ./graphdb &** (Anotar el número de proceso y luego usar **\$ kill numero_proceso** para apagar GraphDB cuando sea necesario).
- **\$./graphdb -d**

3.- Crear archivos RDF

Normalmente el origen de los datos RDF son otras fuentes de datos (Bases de datos relacionales, archivos XML, JSON, ...), pero vamos a crear archivos RDF manualmente para entender el modelo RDF. Crea un archivo RDF que represente el siguiente modelo RDF, usando como guía los ejemplos vistos en clase:



El archivo tiene que estar en Turtle (**Archivo RDF 1**) y NTriples (**Archivo RDF 2**). Puedes usar los siguientes Prefix:

- PREFIX gip:<http://gipuzkoa.eus/resource/>
- PREFIX gip_prop:<http://gipuzkoa.eus/prop/>
- PREFIX dbpedia:<http://dbpedia.org/resource/>
- PREFIX rdfs: **??? [Pista: buscar en prefix.cc]**

Administración de Bases de Datos 2024/2025

Ahora crea un archivo RDF a partir de los datos que se encuentran en el archivo **datos.csv** (**Archivo RDF 3**). Lo puedes crear manualmente o puedes usar una herramienta como Morph-KGC⁶ para crear una conversión ejecutable⁷.

Una vez creado el RDF, crea otro archivo diferente con la siguiente información en un triple (**Archivo RDF 4**): *Bilbo (nuestro dataset) es igual a (owl:sameAs) "Bilbao" (Wikidata)*.

Entrega:

- Archivos RDF (1,2,3,4).
- URL GraphDB en Google Cloud con RDF cargado, y URI Named Graph.

⁶ <https://morph-kgc.readthedocs.io/>

⁷ La extensión Copilot VS Studio Code también hace un trabajo aceptable, pero tened en cuenta que en el examen no váis a poder usar Copilot ni ningún LLM y que el objetivo de este ejercicio es entender el modelo RDF, cosa que sólo se puede hacer convirtiendo datos a mano.