

Reto planteado por Denodo para HackUDC

Este documento contiene la descripción de la propuesta de reto de Denodo para el HackUDC.

Jan 9, 2025

Contents

Contents	2
El reto	3
Definición	3
Datos	3
Conceptos importantes	4
RAG (Retrieval-Augmented generation)	4
Denodo Platform	5
Denodo Express	5
Design Studio	5
Data Catalog	8
Denodo AI SDK	9
Referencias útiles	10

El reto

Definición

El reto consistirá en crear un software que consiga solventar la problemática de falta de conocimiento de los LLMs usando la Denodo Platform y el AI SDK.

Un ejemplo que se puede seguir y presentar sería la elaboración de un chatbot que consiga contestar a preguntas sobre los Juegos Olímpicos de París de 2024 de manera correcta. La temática de las preguntas es libre dentro del marco de los JJOO. Podréis preguntar por resultados, medallas, tiempos, equipos, marcas... Pero una vez elegida esta temática tendréis que asegurarnos que el modelo de datos que diseñéis en Design Studio sirve para contestar a estas preguntas. Y es que la importancia de los datos que se seleccionen es crucial, puesto que en ello estriba la finalidad de la aplicación. Por ejemplo, si solo construís una base de datos con los ficheros que tienen que ver con los partidos de fútbol de los JJOO de París, la aplicación final de nuestro ejemplo solo podrá contestar a preguntas de fútbol. Del resto el AI SDK no podrá obtener conocimiento.

Si no optáis por escoger este conjunto de datos, podéis optar por desarrollar la misma solución pero con otros datos diferentes, siempre y cuando se cumpla que el LLM no sepa contestar de antemano a preguntas sobre estos datos.

El enunciado como véis es abierto, el ejemplo no se tiene por que seguir y podéis hacer otra cosa que no sea un chatbot.

Todo se valorará, no os preocupéis por hacer poco, lo importante para nosotros es que comprendáis el problema, hagáis un buen diseño de los datos e intentéis crear algo a posteriori que utilice el AI SDK. De partida os proporcionamos un frontal web con un chatbot ya hecho, pero se valoraría la creación de uno propio que utilice la API del AI SDK, o cualquier otra cosa que utilice estas APIs, no tiene por que ser un frontal. Aquí es donde os dejamos a vosotros crear lo que queráis.

¡Ánimo y mucha suerte!

Datos

Los datos sobre los Juegos Olímpicos de París de 2024 los encontraréis en este enlace:

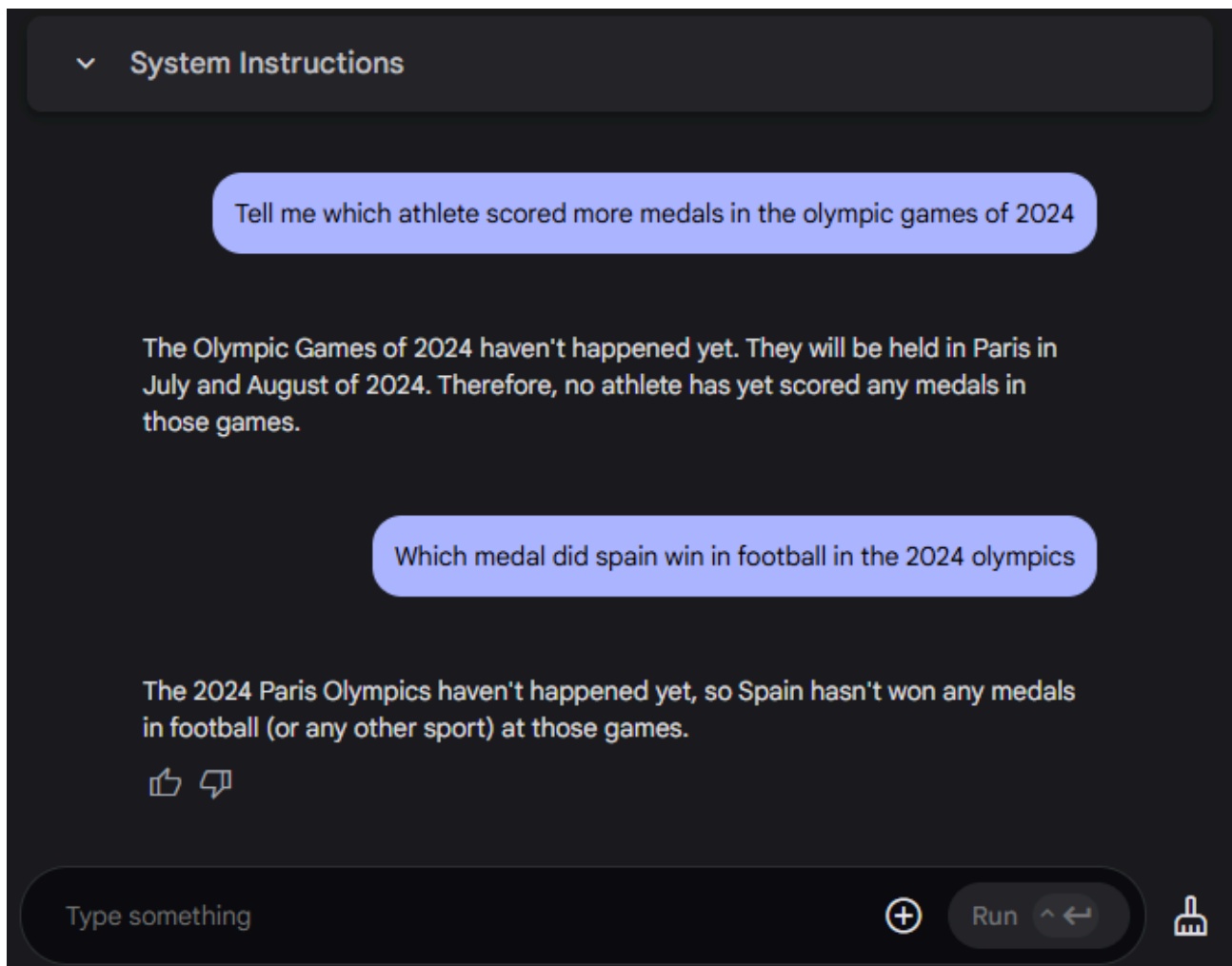
<https://www.kaggle.com/datasets/piterfm/paris-2024-olympic-summer-games/data>

Conceptos importantes

RAG (Retrieval-Augmented generation)

Hay veces en donde los conocimientos que nos ofrecen los LLMs no satisfacen nuestras necesidades, o bien por que no han sido entrenados con la información que nosotros necesitamos que conozcan o bien por que esta información no es pública y por tanto las empresas que diseñan estos LLMs no pueden entrenarlos con ella. Para estos casos existen varias alternativas para solucionar el problema, nosotros nos centraremos en una de ellas, su nombre es retrieval-augmented generation o simplemente RAG. El RAG consiste en que nosotros le demos esa información al LLM como parte de nuestra pregunta para que el LLM pueda utilizar esa información para contestarnos de manera efectiva.

Por ejemplo, ¿qué pasa si le preguntamos al LLM Gemini Flash 1.5 preguntas sobre los juegos olímpicos de París de 2024? Pues resulta que para este LLM, ¡aún no han tenido lugar! Tenemos que solucionar esto.



Denodo Platform

Denodo Platform, impulsada por la virtualización de datos, es una solución de integración de datos, gestión de datos y entrega de datos lógica que proporciona una capa de acceso a datos centralizada que permite a todos los usuarios encontrar, consultar, integrar y compartir conjuntos de datos de forma segura y en tiempo real.

Para este reto será especialmente interesante el concepto de integración de datos. Esto nos permite, entre otras cosas, integrar datos fragmentados en diferentes ficheros para posteriormente poder crear una base de datos y hacer que haya relaciones entre ellos.

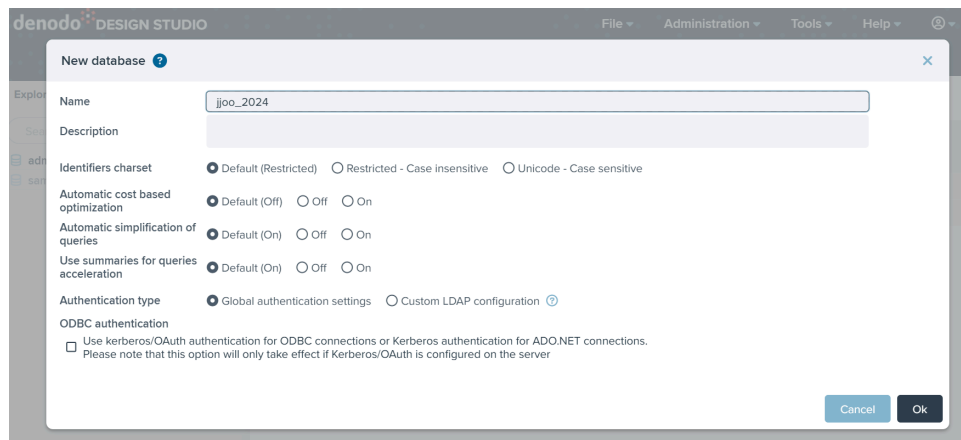
Por otra parte, también es interesante otra cualidad de la Denodo Platform como es la entrega de datos. Gracias a lo comentado anteriormente, podremos construirnos una base de datos y posteriormente nos permite compartir esta información de manera efectiva gracias a que estos datos se pueden exponer mediante APIs, las cuales serán usadas por herramientas externas como Denodo AI SDK, el cuál se explicará más adelante.

Denodo Express

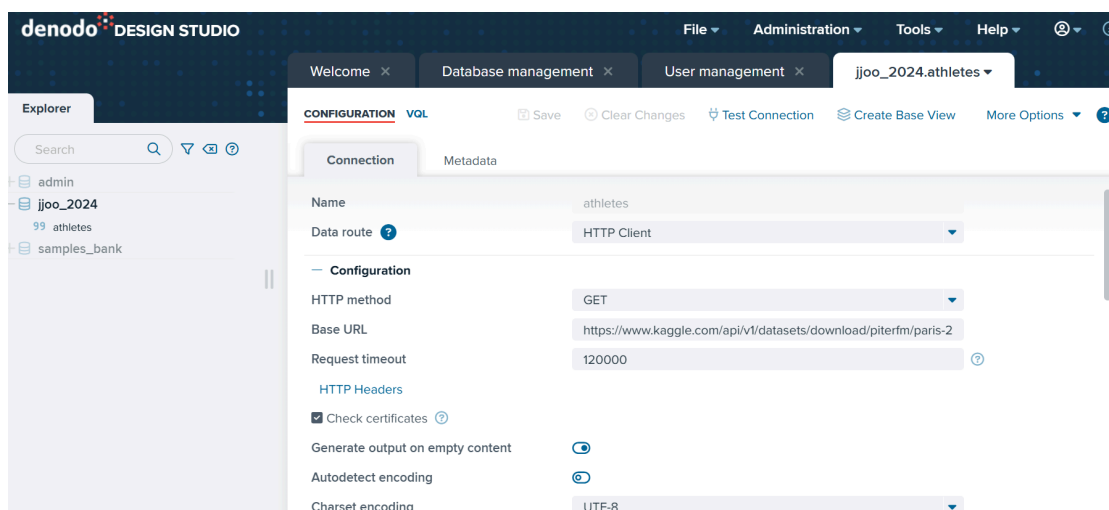
Denodo Express es una solución diseñada para usuarios que desean explorar y aprovechar rápidamente Denodo Platform. Ofrece funcionalidades estándar para integrar diversas fuentes de datos, facilitando un acceso práctico a la gestión lógica de datos. Es por ello que esta versión de la Denodo Platform encaja perfectamente para este reto. Una tendencia emergente es combinar Denodo Express con Denodo AI SDK para impulsar proyectos de IA Generativa (GenAI), que es exactamente lo que vamos a hacer.

Design Studio

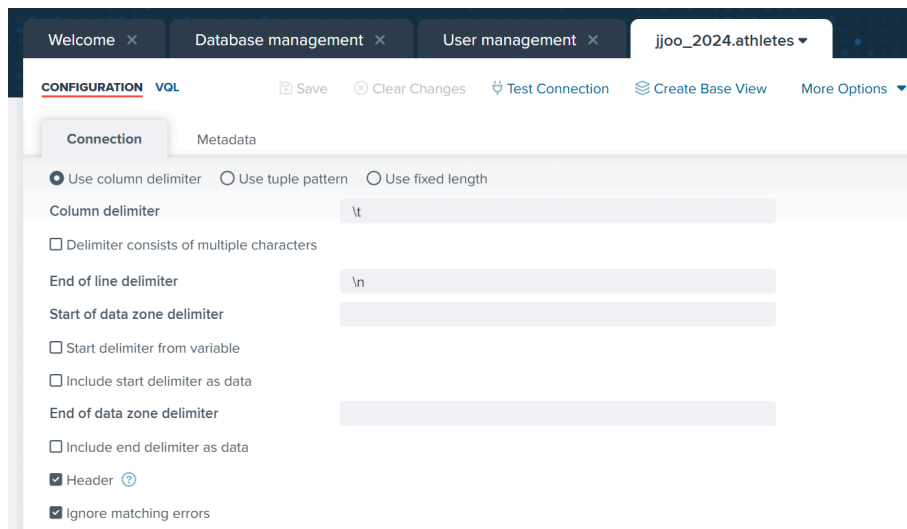
Design Studio es un producto dentro de Denodo Platform. En concreto, será el software que vamos a utilizar para importar los datos que queremos que use el LLM mediante la creación de una base de datos: por ejemplo, una base de datos para almacenar la información relativa a los Juegos Olímpicos de 2024 (conjunto de datos de referencia que puede utilizarse en el reto [Reto > Datos de referencia](#)). Haremos esto a través de Administration > Database management > new):



En esa base de datos creada seleccionaremos los ficheros que la compondrán y cómo se relacionarán entre ellos. Para ello, importamos los conjuntos de datos a utilizar en esta BD que acabamos de crear. ¿Cómo lo haremos? Creando los denominados *data sources*. Estos data sources (de ahora en adelante, DS) pueden ser de diversos tipos (ficheros JSON, delimited files, etc..). Por ejemplo, si queremos inyectar a nuestro LLM datos relativos a los Juegos Olímpicos de 2024 (ejemplo de datos que puede ser utilizado en el reto > [Reto > Datos de referencia](#)), podemos crear un nuevo DS de tipo *delimited file* que obtenga los datos relativos por ejemplo de los atletas a través de la URL donde están alojados ([athletes.csv](#)). Haremos esto situándonos en la BD deseada > new > data source > delimited file):



En este caso hemos seleccionado *HTTP Client* como ruta de los datos a importar, pues los vamos a obtener del link mencionado anteriormente ([athletes.csv](#)). Dentro de la definición de este DS podemos indicar la transformación que queremos hacer sobre estos datos, indicando por ejemplo el *column delimited* (\t) o indicando que la primera fila del archivo se corresponde con las *headers*, de esta manera detectará automáticamente el nombre de las columnas del fichero (check en *header*):



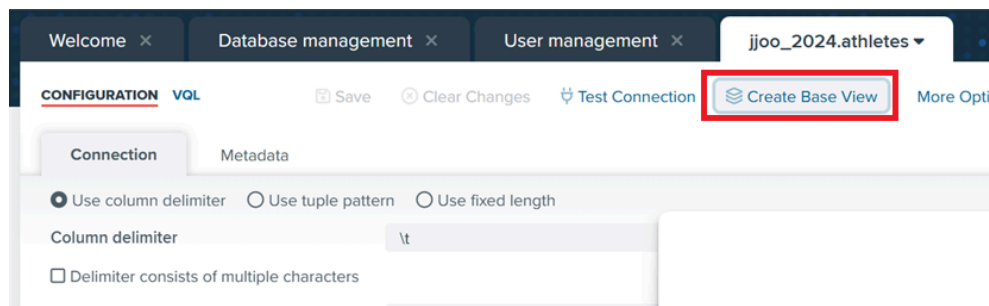
Otra opción sería descargar toda la base de datos como un zip y descomprimir el fichero. Luego, la carpeta se copia al contenedor de docker en donde está la denodo platform con el siguiente comando:

Unset

```
docker cp <nombre de la carpeta> <contenedor>:<ruta dentro del contenedor>
```

Por ejemplo: `docker cp archive 9241d8c73ce6:/home`

Una vez creado este DS, podemos crear una *base view* (una vista virtual, entendida como una representación de los datos de una fuente externa, en este caso, del fichero .csv) para obtener datos de esta fuente: por ejemplo, podremos una vista para hacer consultas sobre el DS importado, es decir, consultas relacionadas con los atletas. Para ello, desde el DS:



Y una vez creada la base view, ya podremos consultar datos sobre esta vista a través de *queries*:

The screenshot shows the Denodo Design Studio interface. The 'Query' tab is active, displaying a SQL query: `SELECT * FROM jjoo_2024.athletes CONTEXT ('cache_wait_for_load' = 'true') TRACE`. The 'Execute' button is highlighted with a red box. Below the query, there are checkboxes for various execution options like 'Retrieve all rows', 'Open results in new tab', etc. The 'Query Results' tab is also visible, showing a table with 150 rows and 6 columns: code, current, name, name_short, name_tv, and gender.

code	current	name	name_short	name_tv	gender
1532872	True	ALEKSANYAN Artur	ALEKSANYAN A	Artur ALEKSANYAN	Male
1532873	True	AMOYAN Malkhas	AMOYAN M	Malkhas AMOYAN	Male
1532874	True	GALSTYAN Slavik	GALSTYAN S	Slavik GALSTYAN	Male
1532944	True	HARUTYUNYAN Arsen	HARUTYUNYAN A	Arsen HARUTYUNYAN	Male

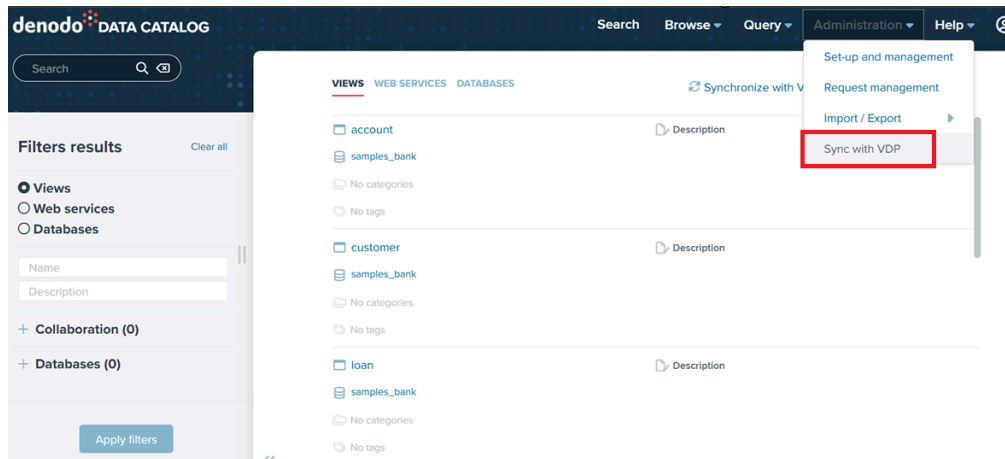
Podemos establecer relaciones entre estos datos. Para ello, utilizaremos *JOINS* entre las vistas que creamos. Puedes consultar cómo hacer *JOINS* utilizando Design Studio → [Joining Views from Disparate Sources](#).

Una vez importados los datos y sus relaciones (joins), debemos sincronizar estos datos utilizando la siguiente herramienta Data Catalog, la cual conectará con el chatbot.

Data Catalog

Data Catalog es otro producto de Denodo Platform. Es el software que permite a los usuarios consumir los datos creados por la empresa o, en este caso, los grupos de este HackUDC. Gracias a Data Catalog estos datos pueden ser compartidos posteriormente con el Denodo AI SDK, que será nuestra interfaz entre los datos y los LLMs.

Para ello, simplemente sincronizamos los datos que hemos diseñado previamente en la web de [Data Catalog](#):



Una vez hecho esto, reiniciamos el container de docker para que recree los metadatos que utiliza el chatbot:

Unset

```
docker rm -f denodo-ai-sdk-chatbot
```

```
docker compose -f docker-compose-sample-chatbot.yml up -d
```

Nota: los pasos de sincronización de datos utilizando el Data Catalog y el posterior recreado de los clusters los repetiremos cada vez que añadamos nuevos datos en el Design Studio, si no, el chatbot no conocerá esta información nueva.

Denodo AI SDK

Denodo AI SDK nos va a proveer con una interfaz para hacer que estos datos que hemos diseñado previamente sean preparados para ser utilizados por un LLM de manera eficiente y eficaz. Básicamente, nos va a permitir no tener que preocuparnos de tener que transformar los datos en embeddings, de orquestarlos ni de crear una base de datos vectorial. Esto permite acelerar enormemente el proceso de crear nuestra aplicación de IA.

El Denodo AI SDK nos va a proporcionar unas APIs que serán de vital importancia para luego crear nuestro propio chatbot o cualquier otro software que queráis realizar. Las APIs las encontraréis en el siguiente enlace: <http://localhost:8008/docs>.

Denodo AI SDK 0.3 OAS 3.1

/openapi.json

Be fearless.

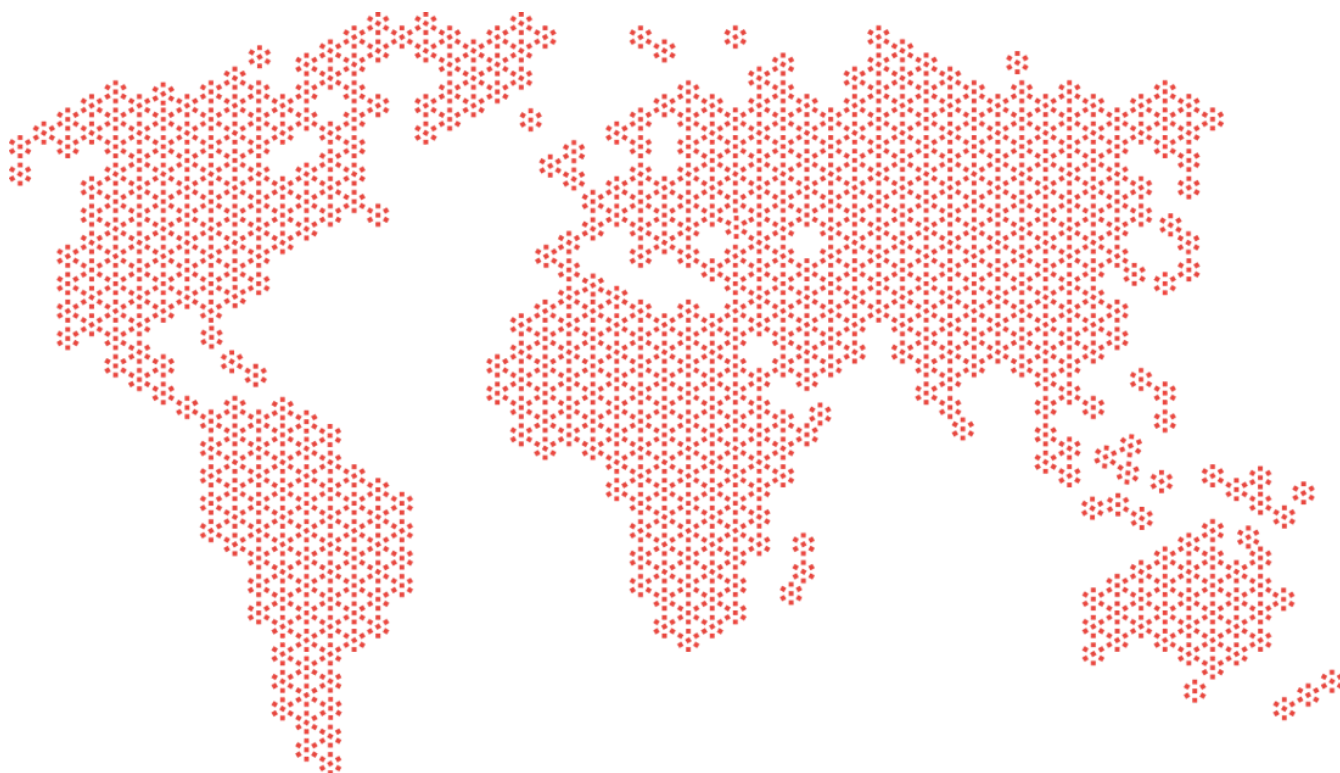
Authorize

default

GET	/getMetadata	Getmetadata	🔒	▼
GET	/getConcepts	Getconcepts	🔒	▼
GET	/similaritySearch	Similaritysearch	🔒	▼
GET	/streamAnswerQuestion	Streamanswerquestion	🔒	▼
GET	/streamAnswerQuestionUsingViews	Streamanswerquestionusingviews	🔒	▼
GET	/answerQuestion	Answerquestion	🔒	▼
GET	/answerDataQuestion	Answerdataquestion	🔒	▼
GET	/answerMetadataQuestion	Answermetadataquestion	🔒	▼
GET	/answerQuestionUsingViews	Answerquestionusingviews	🔒	▼

Referencias útiles

- [Denodo Basics](#)
- [Building an AI Chatbot with Denodo in Minutes](#)



NOTE: *This document is confidential and proprietary of Denodo. No part of this document may be reproduced in any form by any means without prior written authorization of Denodo.*



Visit www.denodo.com Email info@denodo.com Discover community.denodo.com



Control name of the document