Enhancing Denodo with Interzoid APIs

+

Example #2 - Using Interzoid's Global Telephone Information API with Denodo to add generic demographic data to global phone numbers

September 2020

All examples performed with Denodo Express v7

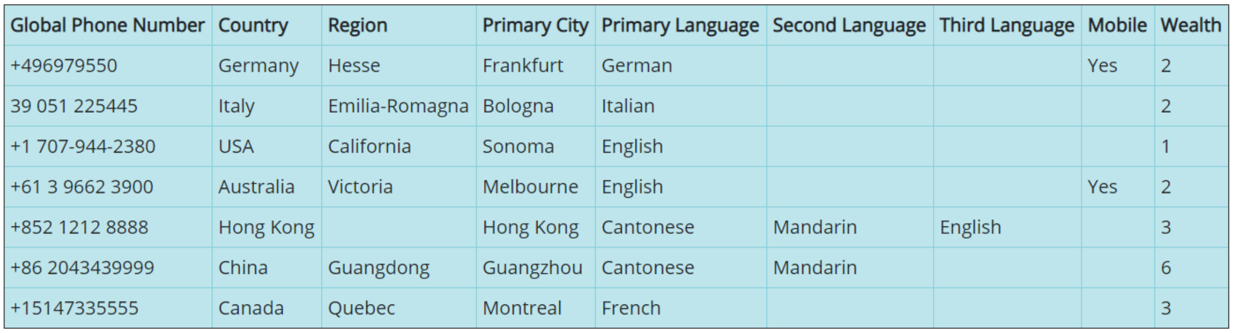
Introduction

In this walkthrough example, **Denodo** is used to perform a Join on a virtual database with Interzoid's Cloud-based Global Telephone Information API. It shows how source data can be augmented and enhanced by utilizing a third-party data source to create new and interesting "views" of existing data. This provides considerably more actionable information to one or more data sources using data virtualization tools, making resultant data integration and analysis activities more valuable and useful.

If you do not have Denodo, a limited-version called Denodo Express is available as a free download: https://www.denodo.com/en/denodo-platform/denodo-express

Interzoid's Global Telephone Information JSON API provides several data points that have been aggregated together for a telephone number anywhere in the world. Additional knowledge about international customers, prospects, and site visitors can provide a better communication experience for an organization, ultimately providing more business value.

Here is an example of enrichment data collected from Interzoid's Global Telephone Information API keyed from only an international phone number:



Global Phone Number	Country	Region	Primary City	Primary Language	Second Language	Third Language	Mobile	Wealth
+496979550	Germany	Hesse	Frankfurt	German			Yes	2
39 051 225445	Italy	Emilia-Romagna	Bologna	Italian				2
+1 707-944-2380	USA	California	Sonoma	English				1
+61 3 9662 3900	Australia	Victoria	Melbourne	English			Yes	2
+852 1212 8888	Hong Kong		Hong Kong	Cantonese	Mandarin	English		3
+86 2043439999	China	Guangdong	Guangzhou	Cantonese	Mandarin			6
+15147335555	Canada	Quebec	Montreal	French				3

The additional international information can be useful in marketing campaigns, in sales scenarios, data science initiatives, and other advanced analytics.

How to enhance international telephone number data in Denodo using data virtualization:

There are **five steps** in this example:

Step #1: Create a Data Source and Base View within Denodo for the source data

Step #2: Configure the Interzoid Global Telephone Information API as a Data Source

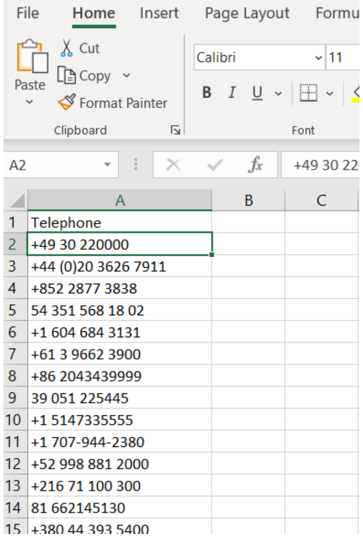
Step #3: Configure the Interzoid API Base View

Step #4: Define the Derived View using a Join function

Step #5: Execute the Join function to display results

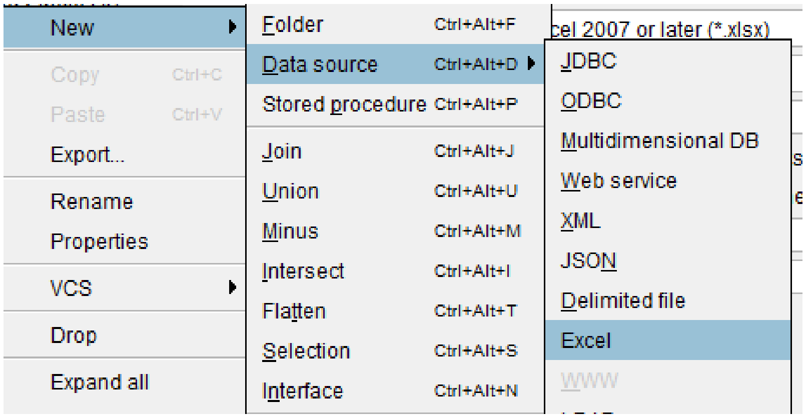
Step #1 – Create a Data Source and Base View within Denodo for the source data

In this example, for simplicity we will use an Excel workbook file as the data source (we could have just as easily used JDBC, ODBC, or one of the several other data sources available within Denodo). It is a list of international telephone numbers. *Filename:* “Example 2 – GlobalPhones.xlsx”

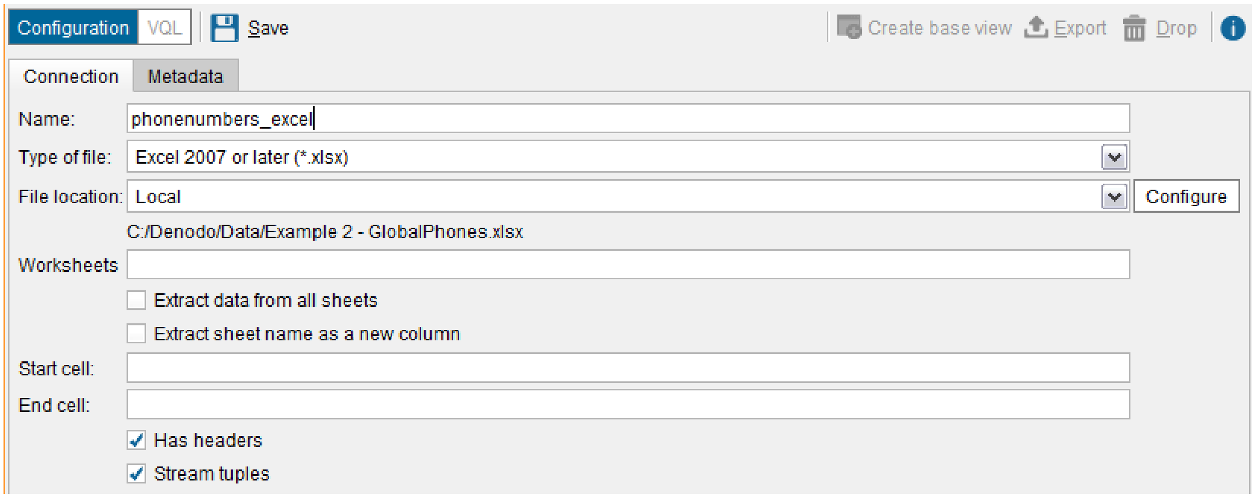


	A	B	C
1	Telephone		
2	+49 30 220000		
3	+44 (0)20 3626 7911		
4	+852 2877 3838		
5	54 351 568 18 02		
6	+1 604 684 3131		
7	+61 3 9662 3900		
8	+86 2043439999		
9	39 051 225445		
10	+1 5147335555		
11	+1 707-944-2380		
12	+52 998 881 2000		
13	+216 71 100 300		
14	81 662145130		
15	+380 44 393 5400		

Within the Virtual DataPort Administration Tool (VDP) of Denodo, select "Database Administration" and create a new Server for this walkthrough. Call it "Example2." Then, from the "Example2" folder in the Server tree, create a new folder called "Data Sources" and another one called "Base Views." Within the "Data Sources" folder, create a new Excel Data Source.



Next, change the File Location setting to "Local" and click the "Configure" button. Provide the path of the included Excel file and click "Ok." Name the Data Source "phonenumbers_excel". Check the "has headers" option and click "Save":



Name: phonenumbers_excel

Type of file: Excel 2007 or later (*.xlsx)

File location: Local

C:/Denodo/Data/Example 2 - GlobalPhones.xlsx

Worksheets

☐ Extract data from all sheets

☐ Extract sheet name as a new column

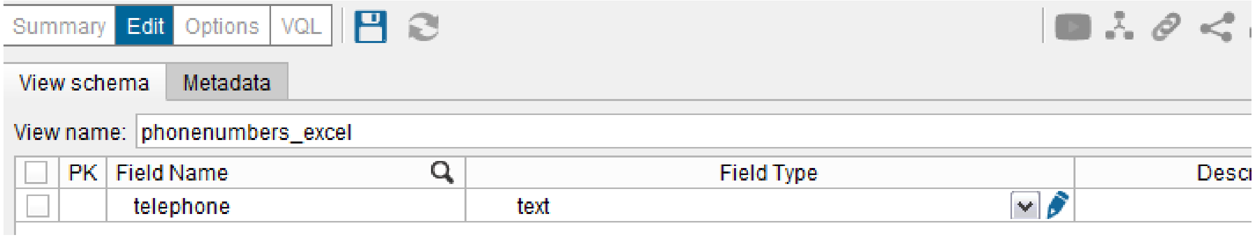
Start cell:

End cell:

☑ Has headers

☑ Stream tuples

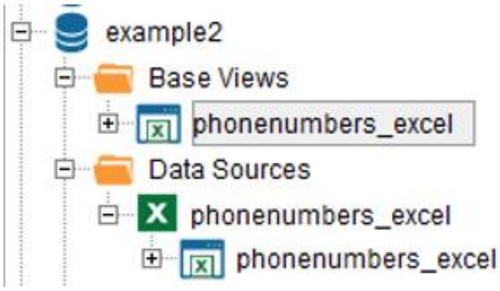
Next, click the "Create base view" button on the top. Once the Base View has been created and is displayed, click "Save":



View name: phonenumbers_excel

□	PK	Field Name	Field Type	Desci
□		telephone	text	

You should now have both a Data Source and the corresponding Base View in your Server tree (you might need to drag the new Base View to the Base View folder):



- example2
  - Base Views
    - phonenumbers_excel
  - Data Sources
    - phonenumbers_excel
      - phonenumbers_excel

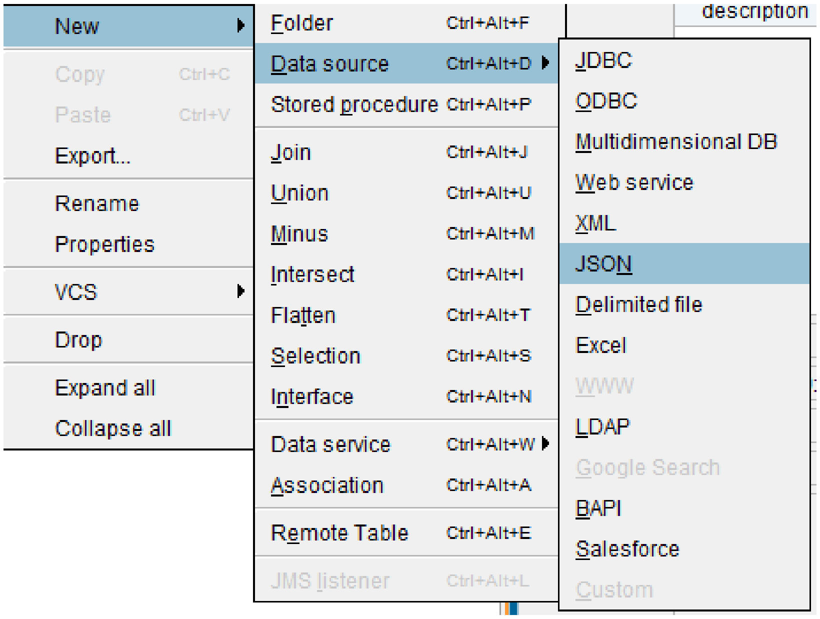
Step #2 – Configure the Interzoid Global Telephone Information API as a Data Source

The API we will use is a REST based API that we will access using query parameters with a URL location. Data is returned from this API in JSON format, so we will set the data source up in Denodo accordingly.

For more technical information about the API:

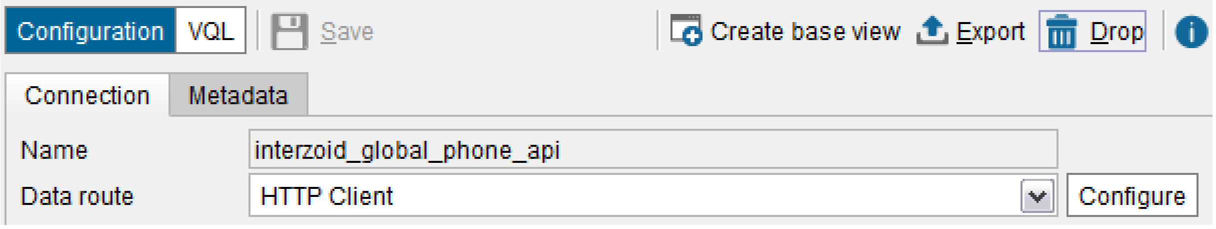
https://interzoid.com/services/getglobalnumberinfo

Create a new data source of type "JSON":



Name the data source: "interzoid_global_phone_api"

For the "Data Route", choose "HTTP Client" and then click the "Configure" button:



Name: interzoid_global_phone_api

Data route: HTTP Client

The end point for this particular API is:
https://api.interzoid.com/getglobalnumberinfo

Calling the API requires two parameters:

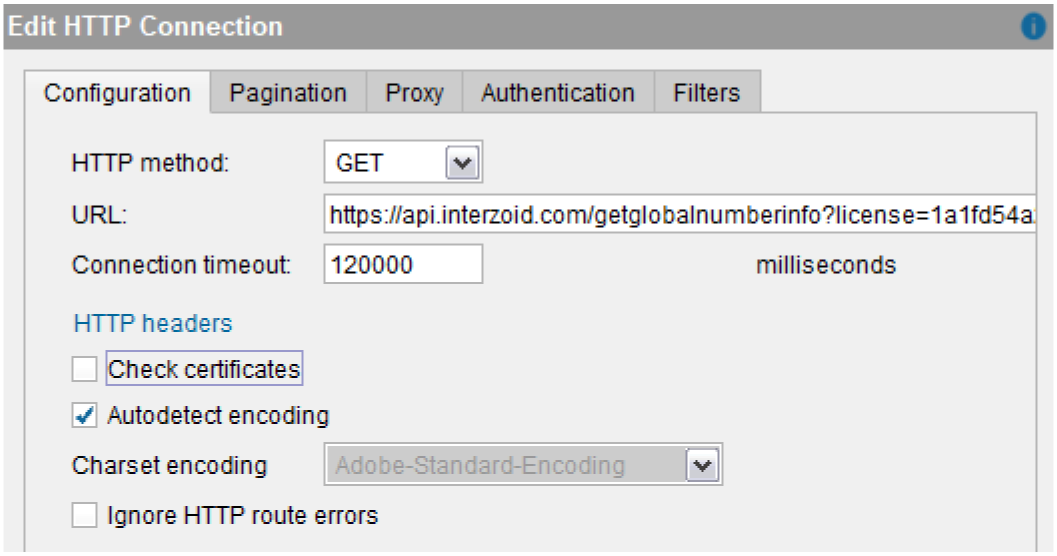
License: A license key issued from Interzoid (a limited key is provided for this example, additional license keys can be obtained from Interzoid)

International Phone Number: Phone number, starting with country code, to retrieve data for

In this example, we will hard-code a value for the “license” parameter. For the other parameter, “intlnumber”, we will use an “interpolated value”, as this tells Denodo to use a variable when calling the API. In our case, the variable will be phone number values from the phone number list Base View that we have already set up with the Excel file as the data source.

To achieve this, in the URL field when configuring the HTTP client, we will provide the endpoint along with the query parameters. You can see the hard-coded value for “license”, and then also how the interpolated variable is denoted within the API call URL using an @ symbol and brackets:

https://api.interzoid.com/getglobalnumberinfo?license=1a1fd54a2f8908405e8b5726c280560a&intlnumber=@{intl_number}



Edit HTTP Connection

HTTP method: GET

URL: https://api.interzoid.com/getglobalnumberinfo?license=1a1fd54a

Connection timeout: 120000 milliseconds

HTTP headers

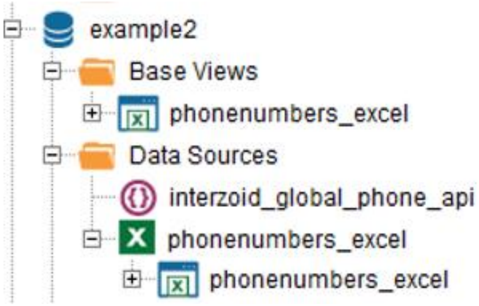
□ Check certificates

☑ Autodetect encoding

Charset encoding Adobe-Standard-Encoding

□ Ignore HTTP route errors

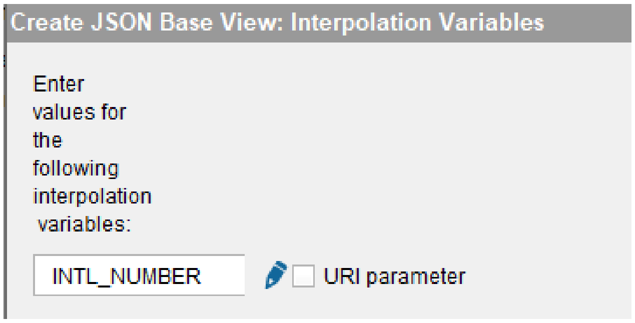
Click “OK”, and then “Save.” The configured Data Source will appear on the Server tree:



```
example2
  Base Views
    phonenumbers_excel
  Data Sources
    interzoid_global_phone_api
    phonenumbers_excel
      phonenumbers_excel
```

Step #3 – Configure the Interzoid API Base View

On the Configuration Panel for the "Interzoid_glopal_phone_api" Data Source, click "Create Base View." You will see the following:

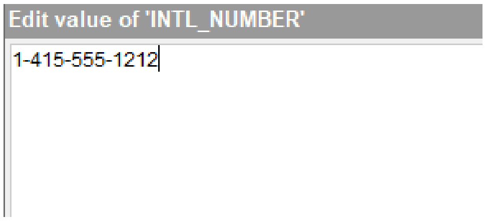


Create JSON Base View: Interpolation Variables

Enter values for the following interpolation variables:

INTL_NUMBER ☐ URI parameter

Click the blue edit pen and then enter the interpolation variable notation below including the brackets(any phone number will do):

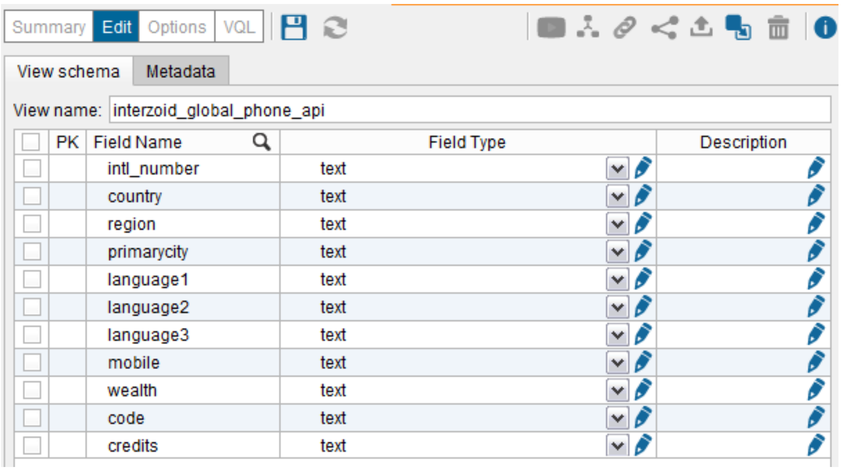


Edit value of 'INTL_NUMBER'

1-415-555-1212

Then click "OK", then "Next", and then "OK" again at the "Configure JSON Wrapper" panel without any changes.

You should then see the following:

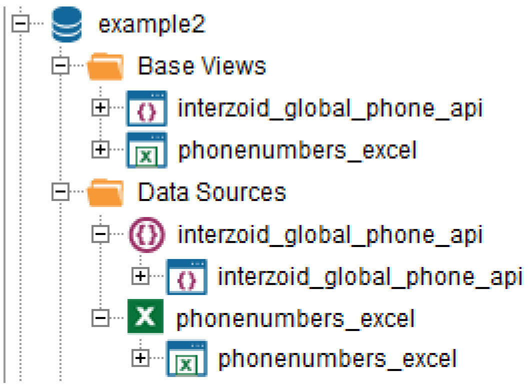


View name: interzoid_global_phone_api

☐	PK	Field Name	Field Type	Description
☐		intl_number	text	
☐		country	text	
☐		region	text	
☐		primarycity	text	
☐		language1	text	
☐		language2	text	
☐		language3	text	
☐		mobile	text	
☐		wealth	text	
☐		code	text	
☐		credits	text	

The “intl_number” field is the input parameter to the API. The other three are all output parameters that will comprise the output results. All other data fields are listed here.

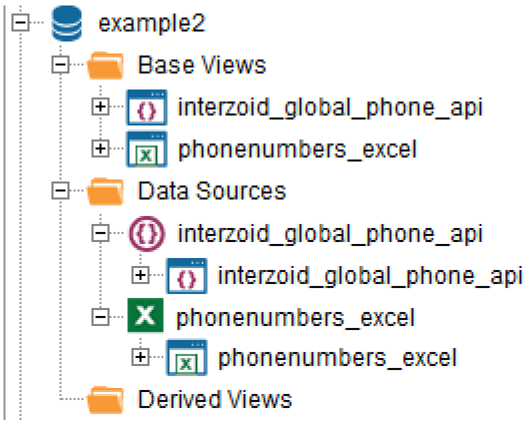
Click the blue “Save” disk to save the Base View. You should now have the new Base View available on the Server tree as shown:



```
example2
  Base Views
    interzoid_global_phone_api
    phonenumbers_excel
  Data Sources
    interzoid_global_phone_api
      interzoid_global_phone_api
    phonenumbers_excel
      phonenumbers_excel
```

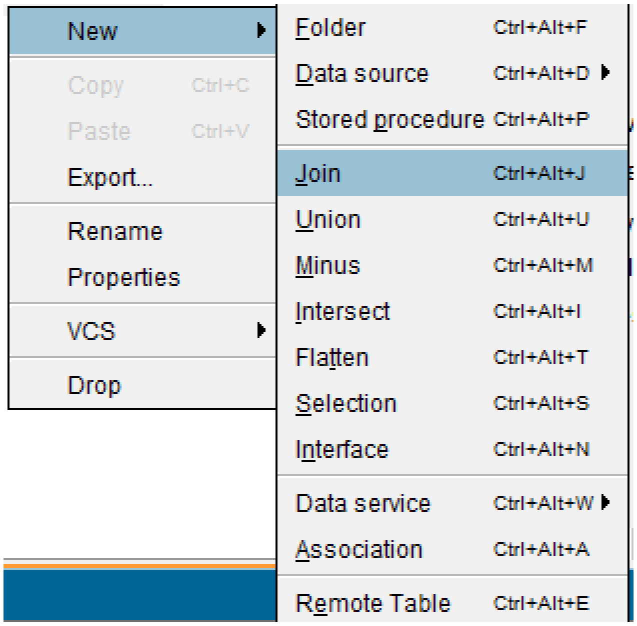
Step #4 – Create the Derived View using a Join function

Create a new folder called “Derived Views” to follow Denodo guidelines. This is where the Data Virtualization will occur and from where we can execute our Join function. This will call the API with each value in the phone number list data source, augmenting them with the demographic data for each and making the enhanced data available within a Derived View (the logical data layer).

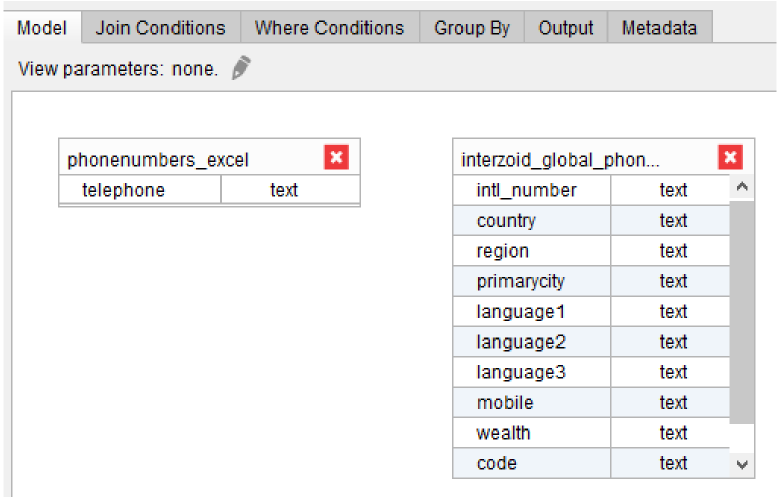


```
example2
  Base Views
    interzoid_global_phone_api
    phonenumbers_excel
  Data Sources
    interzoid_global_phone_api
      interzoid_global_phone_api
    phonenumbers_excel
      phonenumbers_excel
  Derived Views
```

In this folder, create a new Join function:



Then, drag the two Base Views from the tree onto the Join configuration panel. The panel should look like this:

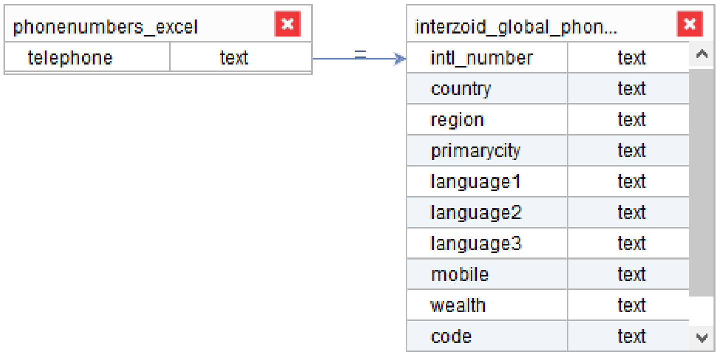


View parameters: none.

phonenumbers_excel	
telephone	text

interzoid_global_phon...	
intl_number	text
country	text
region	text
primarycity	text
language1	text
language2	text
language3	text
mobile	text
wealth	text
code	text

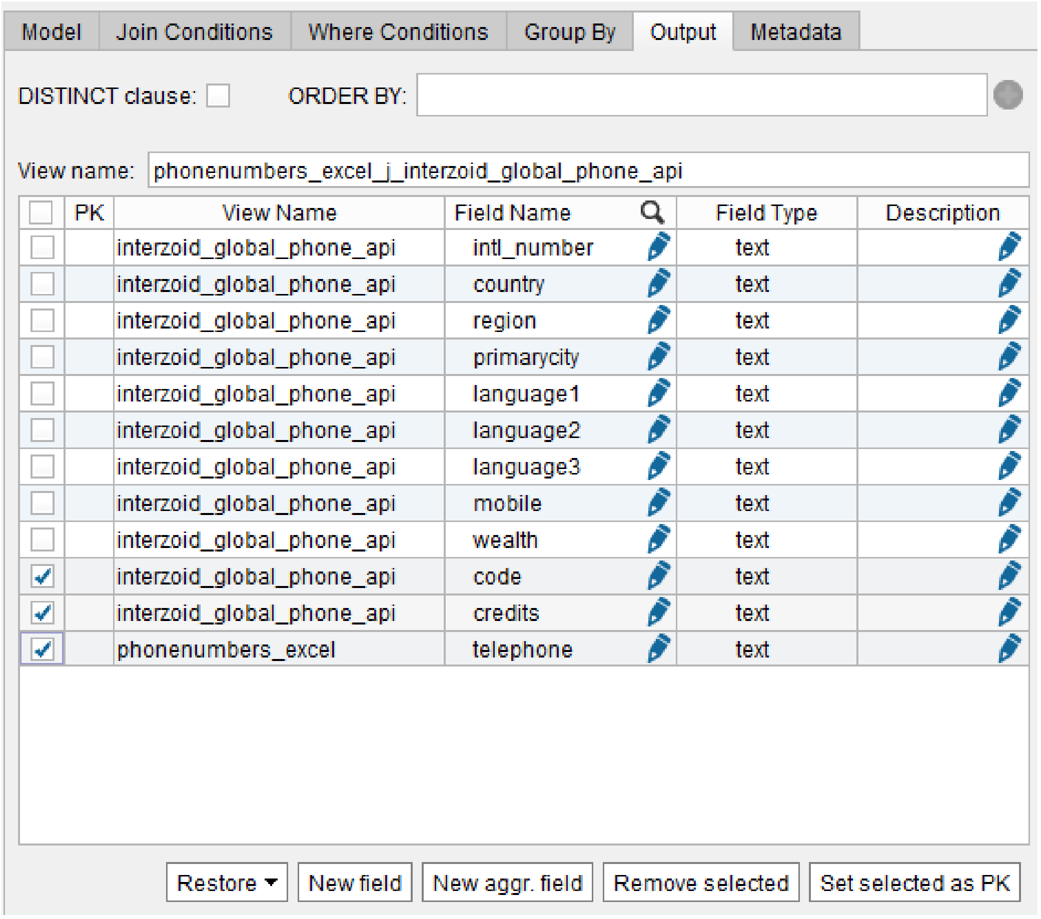
Next, drag the “telephone” field of the phone number list Base View to the “intl_number” field of the Interzoid API Base View. You should see a line connecting the two fields (defining the Join) like this:



phonenumbers_excel	
telephone	text

interzoid_global_phon...	
intl_number	text
country	text
region	text
primarycity	text
language1	text
language2	text
language3	text
mobile	text
wealth	text
code	text

Next, define the output of the Derived View. Click the "output" tab. Then, check the three fields as shown below.

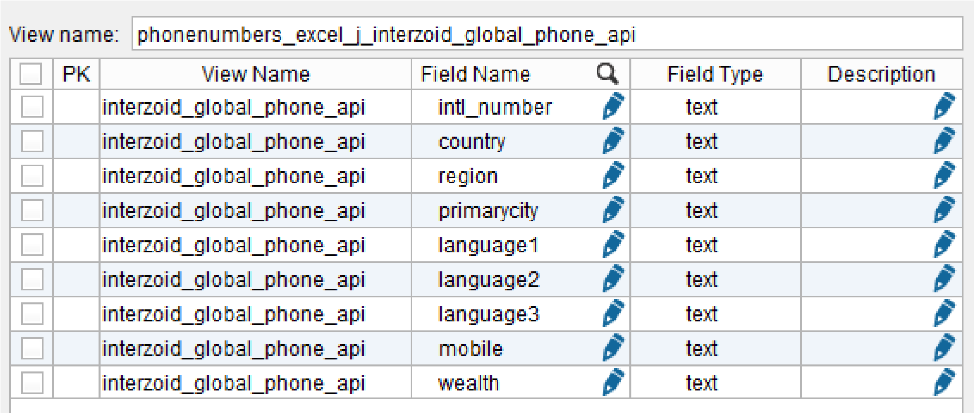


DISTINCT clause: □ ORDER BY:

View name: phonenumbers_excel_j_interzoid_global_phone_api

□	PK	View Name	Field Name	Field Type	Description
□		interzoid_global_phone_api	intl_number	text	
□		interzoid_global_phone_api	country	text	
□		interzoid_global_phone_api	region	text	
□		interzoid_global_phone_api	primarycity	text	
□		interzoid_global_phone_api	language1	text	
□		interzoid_global_phone_api	language2	text	
□		interzoid_global_phone_api	language3	text	
□		interzoid_global_phone_api	mobile	text	
□		interzoid_global_phone_api	wealth	text	
☑		interzoid_global_phone_api	code	text	
☑		interzoid_global_phone_api	credits	text	
☑		phonenumbers_excel	telephone	text	

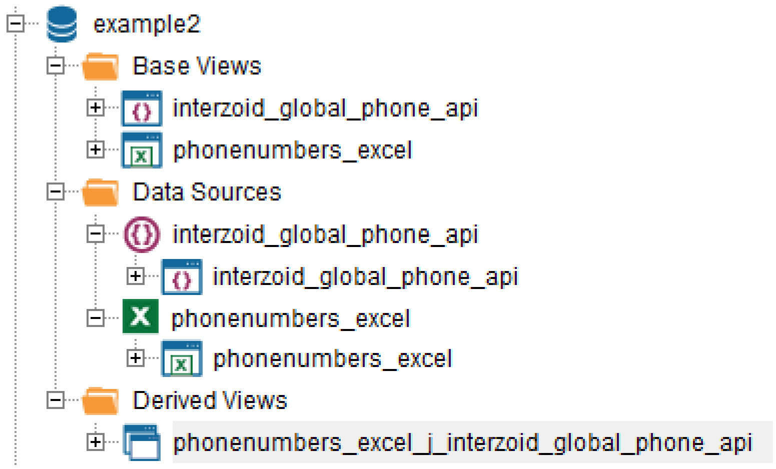
The third field selected is the input to the API. This will be the same as the intl_number field, so we do not need to see it twice. Also, “code” and “credits” are administrative data coming from the API that are not relevant to the Join, so check those as well. After the fields are checked, click the “Remove selected” button at the bottom of the panel. You should now see this:



View name: phonenumbers_excel_j_interzoid_global_phone_api

□	PK	View Name	Field Name	Field Type	Description
□		interzoid_global_phone_api	intl_number	text	
□		interzoid_global_phone_api	country	text	
□		interzoid_global_phone_api	region	text	
□		interzoid_global_phone_api	primarycity	text	
□		interzoid_global_phone_api	language1	text	
□		interzoid_global_phone_api	language2	text	
□		interzoid_global_phone_api	language3	text	
□		interzoid_global_phone_api	mobile	text	
□		interzoid_global_phone_api	wealth	text	

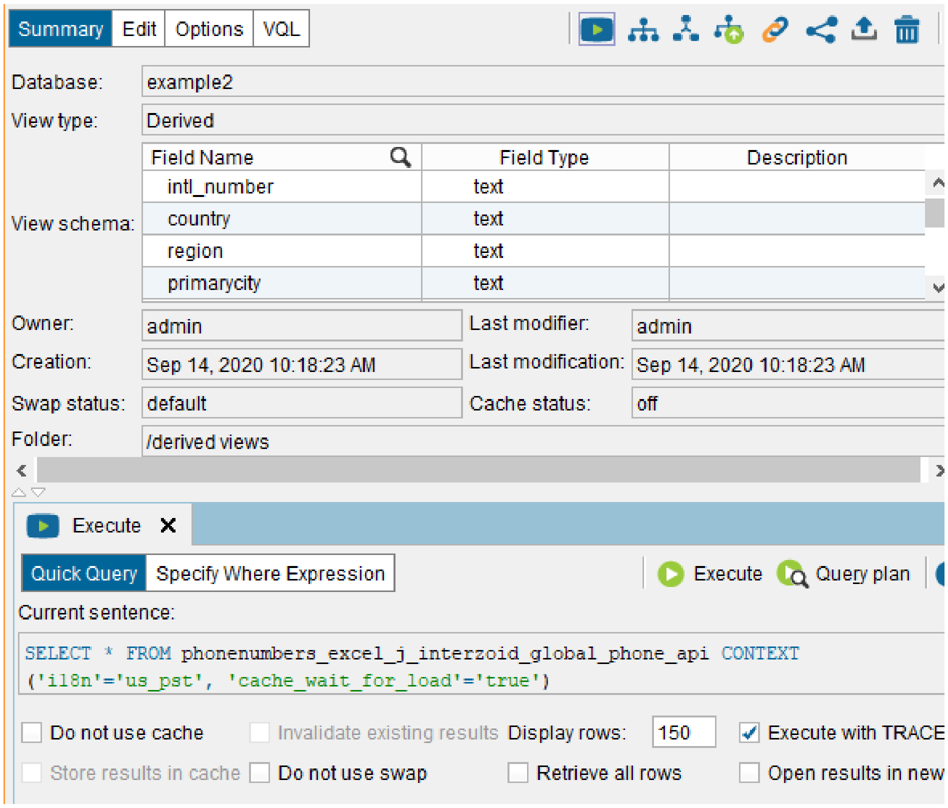
Click the blue disk “Save” button. The Join function’s Derived View will now appear in the Derived Views folder on the Server tree:



- example2
  - Base Views
    - interzoid_global_phone_api
    - phonenumbers_excel
  - Data Sources
    - interzoid_global_phone_api
      - interzoid_global_phone_api
    - phonenumbers_excel
      - phonenumbers_excel
  - Derived Views
    - phonenumbers_excel_j_interzoid_global_phone_api

Step #5 – Execute the Join function to display results

Within the configuration panel for the Derived View, click the “Execution Panel” arrow at the top. You should see the following:



Database: example2

View type: Derived

View schema:

Field Name	Field Type	Description
intl_number	text	
country	text	
region	text	
primarycity	text	

Owner: admin Last modifier: admin

Creation: Sep 14, 2020 10:18:23 AM Last modification: Sep 14, 2020 10:18:23 AM

Swap status: default Cache status: off

Folder: /derived views

Current sentence:

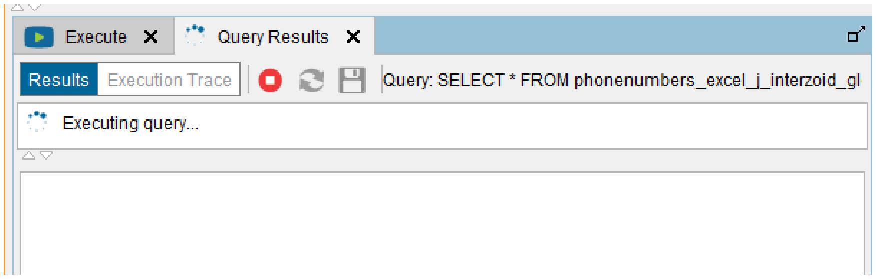
```
SELECT * FROM phonenumbers_excel_j_interzoid_global_phone_api CONTEXT
('i18n'='us_pst', 'cache_wait_for_load'='true')
```

□ Do not use cache □ Invalidate existing results Display rows: 150 ☑ Execute with TRACE

□ Store results in cache □ Do not use swap □ Retrieve all rows □ Open results in new

We are ready to execute the Join function that will build the Derived View by calling the API once for each record in the telephone number list Base View. To execute, click the white arrow in the green circle “Execute” button.

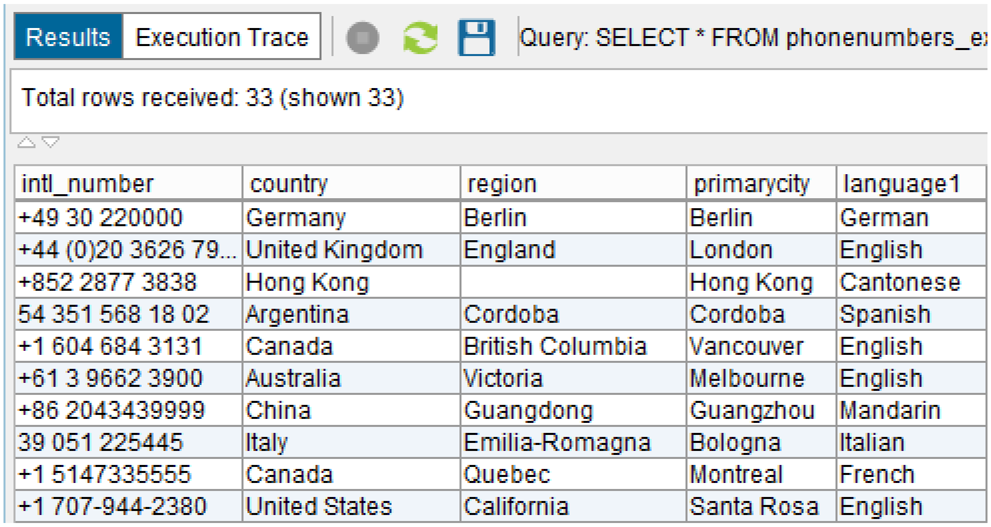
You should see the “Executing query…” message. This will take a few seconds as Denodo is going to the Web to execute the Interzoid Global Telephone API for the purposes of retrieving the enrichment data.



Query: SELECT * FROM phonenumbers_excel_j_interzoid_gl

Executing query...

You will then see the results as shown. For each phone number, you will see corresponding demographic data:



Query: SELECT * FROM phonenumbers_e

Total rows received: 33 (shown 33)

intl_number	country	region	primarycity	language1
+49 30 220000	Germany	Berlin	Berlin	German
+44 (0)20 3626 79...	United Kingdom	England	London	English
+852 2877 3838	Hong Kong		Hong Kong	Cantonese
54 351 568 18 02	Argentina	Cordoba	Cordoba	Spanish
+1 604 684 3131	Canada	British Columbia	Vancouver	English
+61 3 9662 3900	Australia	Victoria	Melbourne	English
+86 2043439999	China	Guangdong	Guangzhou	Mandarin
39 051 225445	Italy	Emilia-Romagna	Bologna	Italian
+1 5147335555	Canada	Quebec	Montreal	French
+1 707-944-2380	United States	California	Santa Rosa	English

Where from here?

Now that our global telephone numbers have been enriched, there are several potential courses of action to take, depending on business requirements.

For example, data from this logical data layer can be exported for further analysis and use. Within Denodo, results can be saved/exported to a delimited file for review or additional processing.

This walkthrough is only one example of integrating Interzoid APIs within Denodo. There are other data enrichment APIs that can augment existing sources of data with new information. There are also matching APIs that include specific matching algorithms for different types of data such as individual names (Bob = Robert, Johnson = Jonsen) or company names (IBM = International Business Machines).

For more information, contact support@interzoid.com or visit https://www.interzoid.com