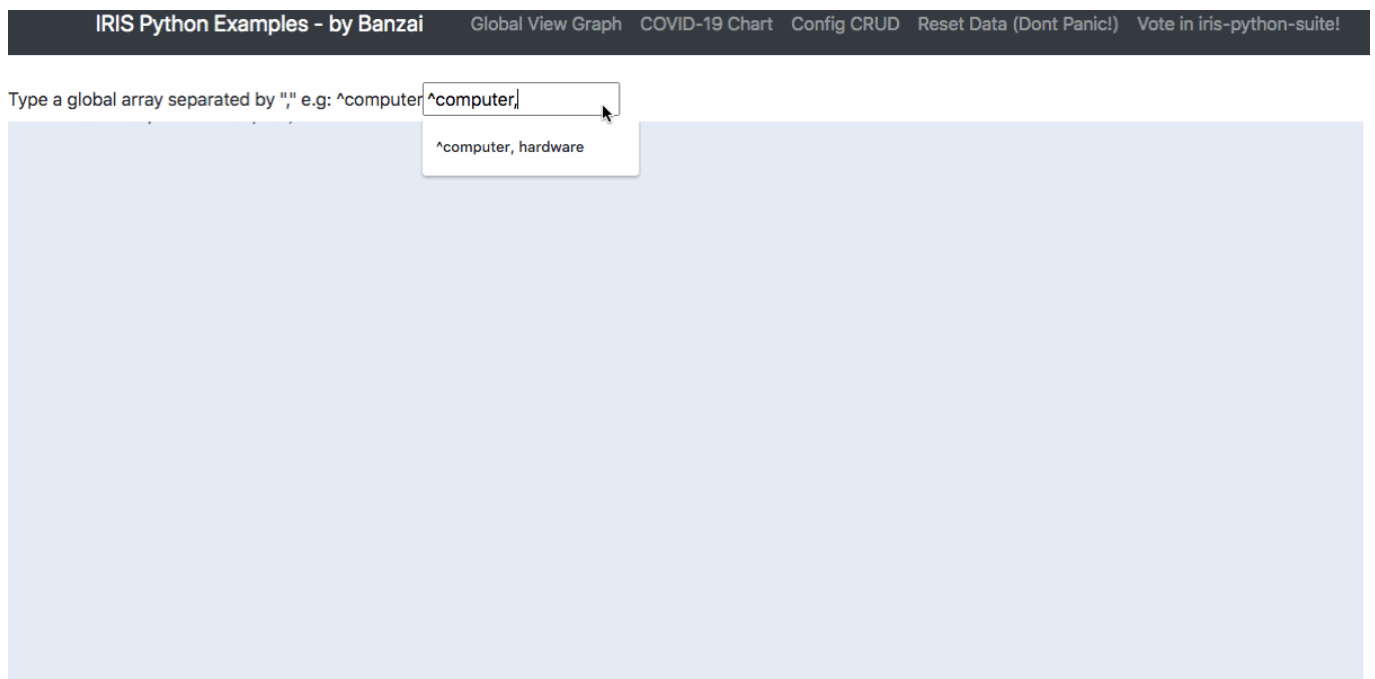


Article

[Renato Banzai](#) · Jun 2, 2020 4m read

[Open Exchange](#)

Using Python to Represent Globals in a Network Chart



Globals as a Graph Data Structure in Python

How to use the IRIS Native API + Python to see globals as a Graph Network Chart.

Iris Globals

Reading the documentation these topics are related to globals:

- A global consists of a set of nodes (in some cases, only one node), identified by subscripts.
- Each node can contain a value.
- ObjectScript includes functions to iterate through the nodes of a global and quickly access values.
- A global is automatically stored in the database. When you assign a value to a node of a global variable, the data is written immediately to the database.
 - You can see the contents of a global via an ObjectScript command or via the Management Portal.

A Python Way to See Globals

As one of representations of globals can be a Graph Data Structure there are some modules in Python that can transform these globals in a visualizable graph.

The Chart Application

Demo - Try it yourself

I have deployed the application as a demo here, my IRIS Database has one global to test ^computer:
(<http://iris-python-suite.eastus.cloudapp.azure.com/global-chart>)

Into the code

Clone my repository to see all the code implementation.

```
$ git clone https://github.com/renatobanzai/iris-python-covid19.git
```

What did I use in Python

In this application environment I use Python 3.7 with these modules.

- PyYAML==5.3.1
- dash==1.12.0
- plotly==4.7.1
- networkx==2.4
- numpy==1.18.4
- dash-bootstrap-components==0.10.1
- irisnative-1.0.0-cp34-abi3-linux_x86_64.whl

Project Structure

This project has a simple structure to be easy to understand. On the main folder we have 3 most important subfolders:

- ./app: with all the application code and installing configuration.
- ./iris: with the InterSystems IRIS dockerfile preparing to serve the application.
- ./data: with the files from Johns Hopkins University to ingest and a YAML to change configuration outside the container environment by a volume

Application Structure

Now inside the ./app directory we can see some files:

- __main__.py : with the implementation of the web application
- iris_python_suite.py : a class performing all data transformation to convert the globals into a networkx graph.

Database Structure

This application uses InterSystems IRIS as a repository, the globals used are:

-^computer : A global to test the graph. If you want, you can test with all other globals default in the USER Namespace.

There are some other globals created by the application that can be used as a test too:

- ^config : with some config data
- ^raw.covid19 : where the raw data (Source of Data) are ingested
- ^countrydetails : to get the population of each country

-^end.date.deaths : to serve the chart requisitions and here is the goal, Its fast!

-^end.timeless.deaths : to server another kind of chart requisition

App Structure

iris_python_suite.py: Inside this file are 2 classes that makes the job:

- irisdomestic: Has the same features of irishnative + creates instances of irisglobalchart, irisglobal, etc (factory pattern)
- irisglobalchart: Make a recursive track into the global, converting all the data in a Graph Networkx.

Why do I need to convert the Graph into a Networkx Object?

If are you asking yourself, the module networkx has a function position nodes using Fruchterman-Reingold force-directed algorithm.

Algorithm Fruch... WHAT?

As a graph can have any shape is too hard to represent it in a generic way. This is on algorithm to represent graphs without a lot of confusion.

The line that perform the use of this algorithm is on python_suite_global.py:

```
def get_fig(self):
    _nx = self.obj_nx
    pos = nx.spring_layout(_nx)
```

Running the application by yourself

Prerequisites

- git
- docker and docker-compose
- access to a terminal in your environment

Steps

With docker-compose you can easily up one environment with all the pieces and configurations go to the iris-python-covid19 folder and type this:

```
$ docker compose build
$ docker compose up
```

Estimated time to up containers

1st time running will depend of your internet link to download the images and dependencies. If it last more than 15 minutes probably something goes wrong feel free to communicate here. After the 1st time running the next ones will perform better and take less then 2 minutes.

If is everything ok

After a while you can open your browser and go to the address:

```
http://localhost:8050/global-chart
```

You should look at IRIS Admin Portal

I'm using for now the USER namespace

```
http://localhost:9092  
user: _SYSTEM  
pass: theansweris42
```

If this article help you or you like the content please vote:

This application is at the current contest on open exchange, you can vote in my application iris-python-suite here(<https://openexchange.intersystems.com/contest/current>)

[#Globals](#) [#Graph](#) [#Python](#) [#InterSystems IRIS](#) [#InterSystems IRIS for Health](#)
[Check the related application on InterSystems Open Exchange](#)

Source URL: <https://community.intersystems.com/post/using-python-represent-globals-network-chart>