
Article

[Robert Cemper](#) · Feb 9, 2022 2m read

GlobalToJSON-embeddedPython#2

I will give you some additional information on my first embedded Python package. it is written as a mix of python and ObjectScript to take the best of both worlds.

1. The target was to create an easy-to-read and easy-to-understand JSON file that also allows editing the contents within some limits with an external editor.
2. The result should be independent of 3.4 MB limit of strings in IRIS. So the result should go to a %Stream. Which is in case of Python an external file.
3. Also, the processing should be easy to follow and avoid tricky libraries.
4. And most imported I wanted to show the interaction between methods written in ObjectScript and methods written using embedded Python inside a common class.

So my choice was to keep all code close to Global access in ObjectScript

- Checking global existence with \$DATA()
- Running along the Global nodes with \$QUERY()
- Converting %LISTBUILD() items into a readable format
- Converting and storing \$LISTBUILD() content during load

So left to Python were these tasks

- Creating the result file
- Packing node reference and content into a JSON Object
- Reading and checking the input file during global load
- Unpacking the JSON objects
- Writing the global nodes for 'simple contents as strings and numbers
- Initiating write of \$LISTBUILD() content in Objectscript.

So there is a lot of interactions initiated from both sides. This was the major purpose of this exercise.

Of course, you may argue that all this could have been done just by using ObjectScript. And this is correct, but not the purpose.

[Video](#)

[GitHub](#)

[#Embedded Python #Globals #JSON #ObjectScript #InterSystems IRIS](#)

Source URL: <https://community.intersystems.com/post/globaltojson-embeddedpython2>
