Purpose: to better visualize and understand the full hierarchy of a Cube Application (i.e. a travel demand model in Cube) on a web-browser

Requirements: Python 2.7, modern web-browser

Context: One of the best features (atleast in my opinion) of Cube Base is the simple yet powerful flow-chart interface that it provides to visualize the contents of a travel demand model written in Cube Voyager. However, the interface could use a few updates to make it more functional. One such feature upgrade would be the ability to view the entire model structure on a single screen-shot, as opposed to navigating through the different levels of hierarchy one at-a-time.

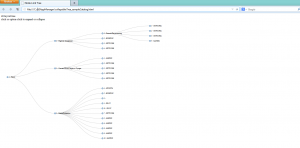
So, I took a crack at it with a python script. This python script extracts the relevant information from the thankfully human readable .app files (you can open them with notepad++) and writes it to a JSON file. Further, this JSON file is passed through a [D3.js](http://d3js.org/) template to produce a beautiful [collapsible tree layout](http://mbostock.github.com/d3/talk/20111018/tree.html).

Example:

Input:

[](http://akomma.soapmelon.com/wp-content/uploads/2013/02/2-12-2013-11-40-56-AM.png)

Output:

[](http://akomma.soapmelon.com/wp-content/uploads/2013/02/2-12-2013-11-51-55-AM.png)

Download: <http://github.com/akomma/d3AppManager.git>

Usage:

1. Copy all the .app files from your model into the ‘sample Catalog’ folder
2. Make sure you point to the main parent application in line#10 of the python script. Save .py file
3. Run the python program (written on ver 2.7.3), it produces ‘sampleCatalog.JSON’ file
4. Now open the ‘collapsibleTree\_sampleCatalog.html’ file using your browser (it reads the .JSON file created)
5. You should be able to the see the Cube model structure as a interactive collapsible tree now

Future improvements: Please note that this is fairly a basic implementation. I plan on updating it further. If you would like to improve it, please fork the repo on github and have at it. Some ideas are as following:

* Ability to visualize loop controls present in the model
* Add mouse-over events to show text-snippets including runtime, execution order
* Filter the tree layout my program type (eg: display on matrix program modules)