Jupyter Tool Deployment Styles

How do you want your Jupyter Notebook-based tool to behave when a user runs it? You have several choices, which we describe here:

1. Tool or App Mode style--code cells are hidden, UI widgets are visible
2. Notebook style--all code cells are shown
3. Autorun style--shows and runs code cells on load

If you're working out how to develop and deploy the notebook, please refer to the other articles in this series. Continue reading to choose and set the tool display style.

Using Jupyter on Deb10 version

If you are running the newer, Jupyter on Deb 10 tool (alias "jupyter6deb10") to develop notebooks, you can achieve the dashboard-style effects by just selecting the App Mode in the Jupyter menu bar. Toggle back to edit your notebook using the Edit App button.

This screenshot shows the Jupyter on Deb 10 tool menu bar, with its Appmode button

Deploying as App Mode

Then, to deploy the tool in App Mode, specify an -A in your invoke script, as follows:

```
/usr/bin/invoke_app "$@" -C "start_jupyter -A -T @tool MyNotebook.ipynb" 
-t mytoolname 
-u anaconda-6 -r none
```

The tool user will still be able to toggle the tool to show its code cells. (To suppress that behavior, add a -t to the start_jupyter call)

For the older Jupyter Notebooks tool, read on. Note that we encourage you to switch to use jupyter6deb10 if you are undertaking any substantial changes to your tool.
Using Jupyter (debian7 version)

For the older Jupyter Notebooks tool (alias "jupyter"), read on. Note that we encourage you to switch to use jupyter6deb10 if you are undertaking any substantial changes to your tool or if you encounter problems with changing the notebook style.

Dashboard ("Tool" or AppMode) style

To deploy a notebook as "tool" or dashboard style, the code will be hidden, but the widgets, plots, and markdown will be displayed.

Two things must be done to deploy your tool in this style. From the Workspace tool's command line, do the following:

First, mark the tool mode on your notebook:

```bash
jupyter_tool.py -t notebook-name.ipynb
```

Then, set your notebook as trusted:

```bash
jupyter trust notebook-name.ipynb
```

Finally, check the code in to subversion (svn).

Note that you cannot make changes to a notebook once it's marked as a trusted tool. If you must make changes to a notebook that's deployed as a Dashboard, first follow the instructions in Notebook style below, then make your changes, and finally re-mark the notebook to use tool mode and trusted mode as above. Then you can check in the Dashboard-style tool's changes to svn!

Notebook style

To deploy a notebook as notebook style, all code in the notebook will be displayed. This is the default for Jupyter notebook based tools, so if it's a new tool, there's nothing special you must do. However, if you're switching a notebook from tool style to notebook style, you must do these things:

First, mark the tool mode to notebook:

```bash
jupyter_tool.py -n notebook-name.ipynb
```
Next, unsign your notebook:

```
jupyter_unsign.py notebook-name.ipynb
```

Open your notebook and make a change (if no code change is needed, just add a line of whitespace). Save that.

Lastly, check the source in to subversion (svn) and continue with the tool deploy.

Note that just changing the tool mode or signing/unsigning is insufficient. *You must change the notebook itself* then check it in in order to change its behavior.

**Autorun style**

For autorun, run with `-a` to autorun all cells on load of the notebook.