HTCondor Python Bindings Tutorial

Brian Bockelman

European HTCondor Week 2017

HTCondor Clients in 2012

Command Line Clients

Fully Featured!

Requires fork/exec and process handling

Outputs in multiple formats

Something Missing In The Middle

SOAP Clients

Features! (Some)

Language agnostic (everyone hates XML equally?)

Caveats with respect to scalability, security.

Python Bindings

- In late 2012, I wanted better integration between HTCondor and python.
    - Existing attempts made for awful libraries: as a rule, Unix libraries should not fork/exec.
    - Nothing handled ClassAds correctly.
    - Dan Bradley had previously mentioned that boost.python appeared to be a good way to integrate Python with C++ code.
- So, on December 26, I set off to see what could be done…

Design Philosophy

- **ClassAds**: Everything based on ClassAds; make these the “core” of the bindings.
- **pythonic**: Semantics and APIs should feel natural to a python programmer.
    - Use iterators, exceptions, guards. ClassAds behave as much like a dict as reasonable.
- **Backward compatible**: APIs are here to stay for as long as possible.
    - When we absolutely must, use standard python `DeprecationWarning` techniques.
- **Native code**: Call same HTCondor library code as CLI; identical in performance.
- **Complete**: If you can do it with the command line tools, you should be able to do it with python.

Not Our First Rodeo

- We, uh, might have done a python bindings tutorial before:
  - http://research.cs.wisc.edu/htcondor/HTCondorWeek2013/presentations/Bockelman_Python.pdf
  - http://research.cs.wisc.edu/htcondor/HTCondorWeek2014/presentations/TheisenT-Python.pdf
  - https://research.cs.wisc.edu/htcondor/HTCondorWeek2016/presentations/Bockelman_Python-tutorial.pdf
- Goal for this year: **Overhaul the experience!**

Pythonic!

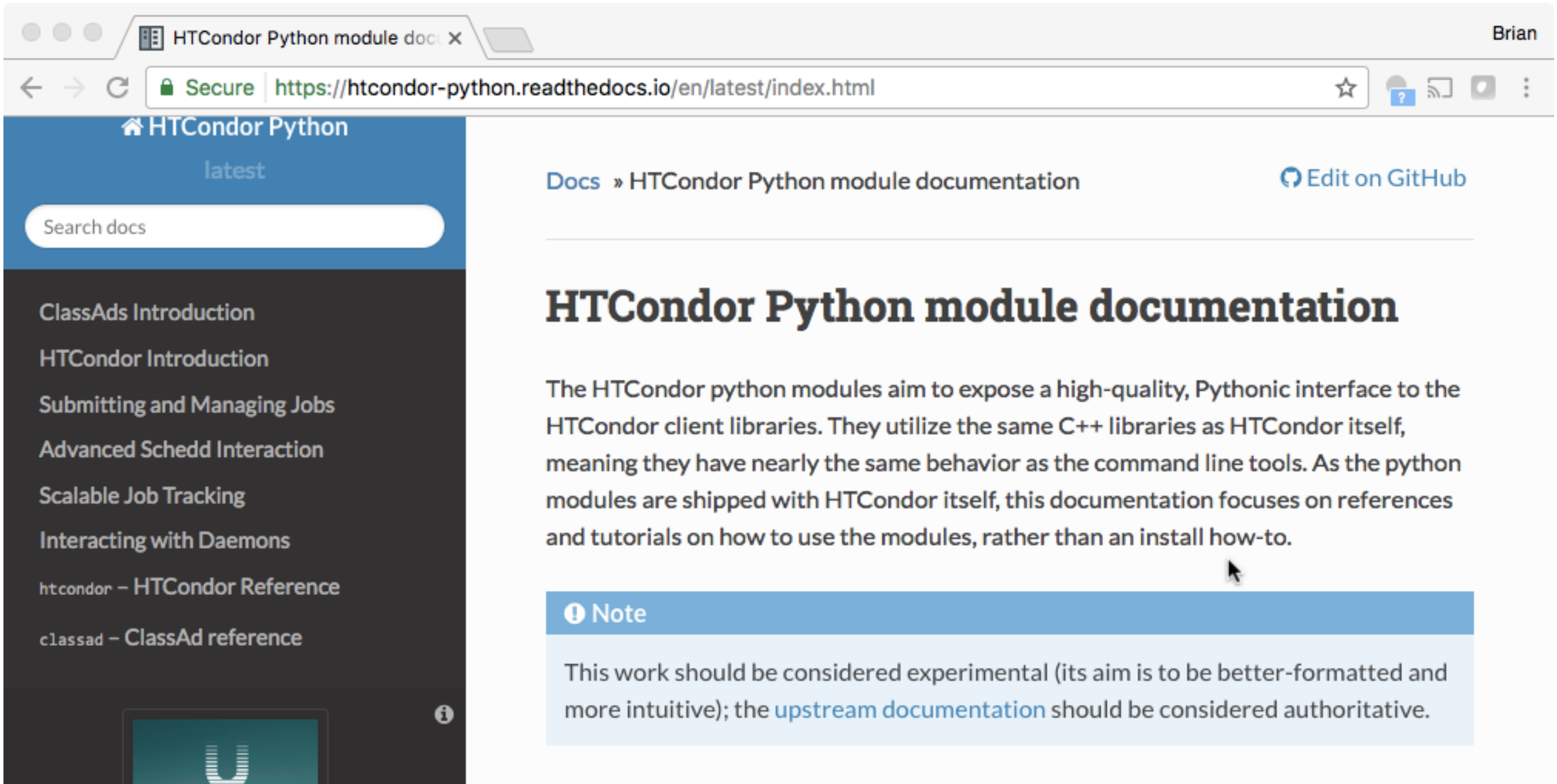
- Since *pythonic* is in our design philosophy, I decided the education should use the tools favored by the python community:

- Sphinx-based documentation. Hosted on ReadTheDocs; looks / feels / smells like python documentation. Updated, more complete, and contains *tutorials*: not just a reference!

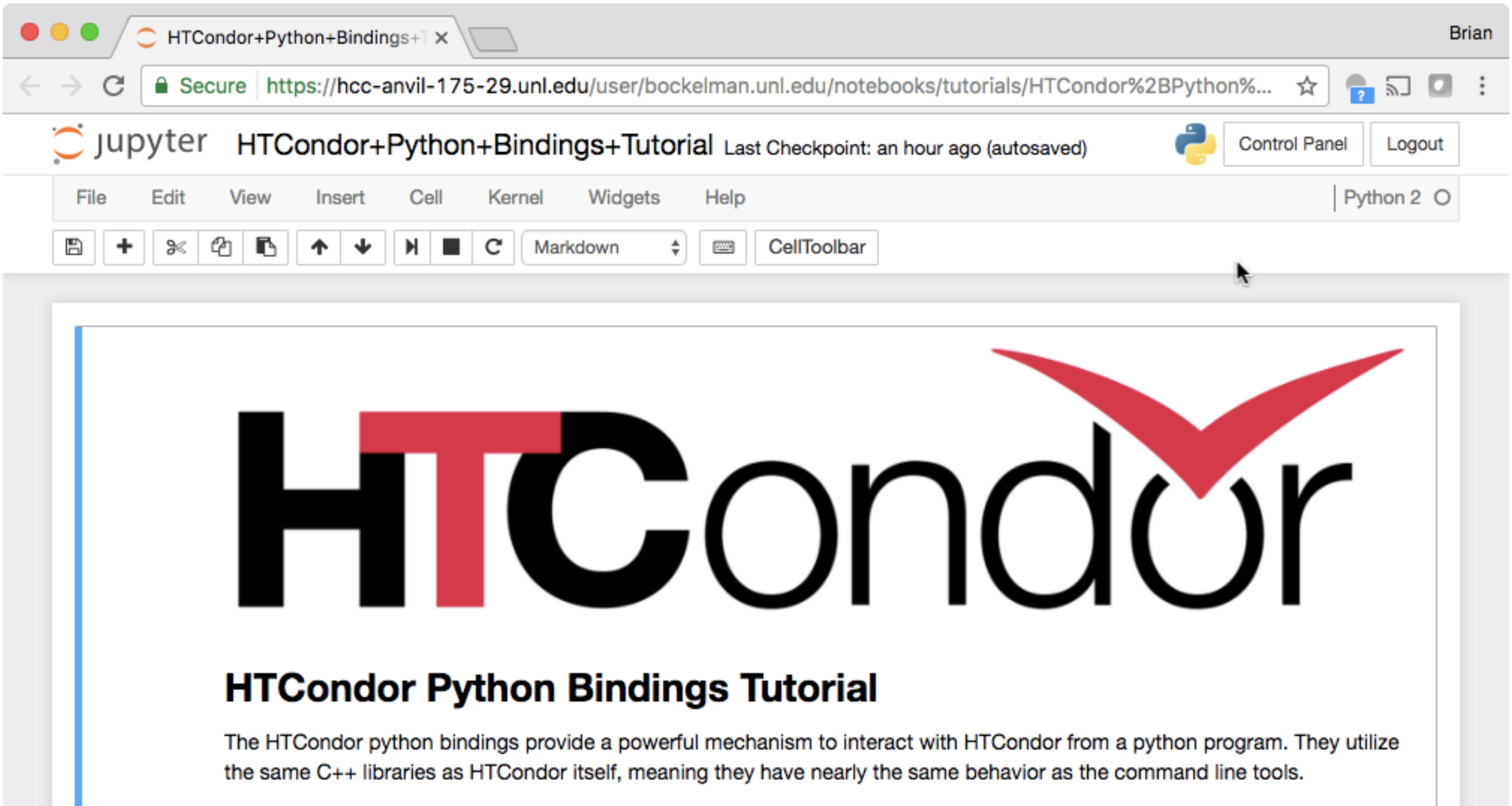
- JupyterHub-based tutorials. Login with a university credential; spawns a Docker container with a private HTCondor instance. Interact via your browser.

Sphinx Docs

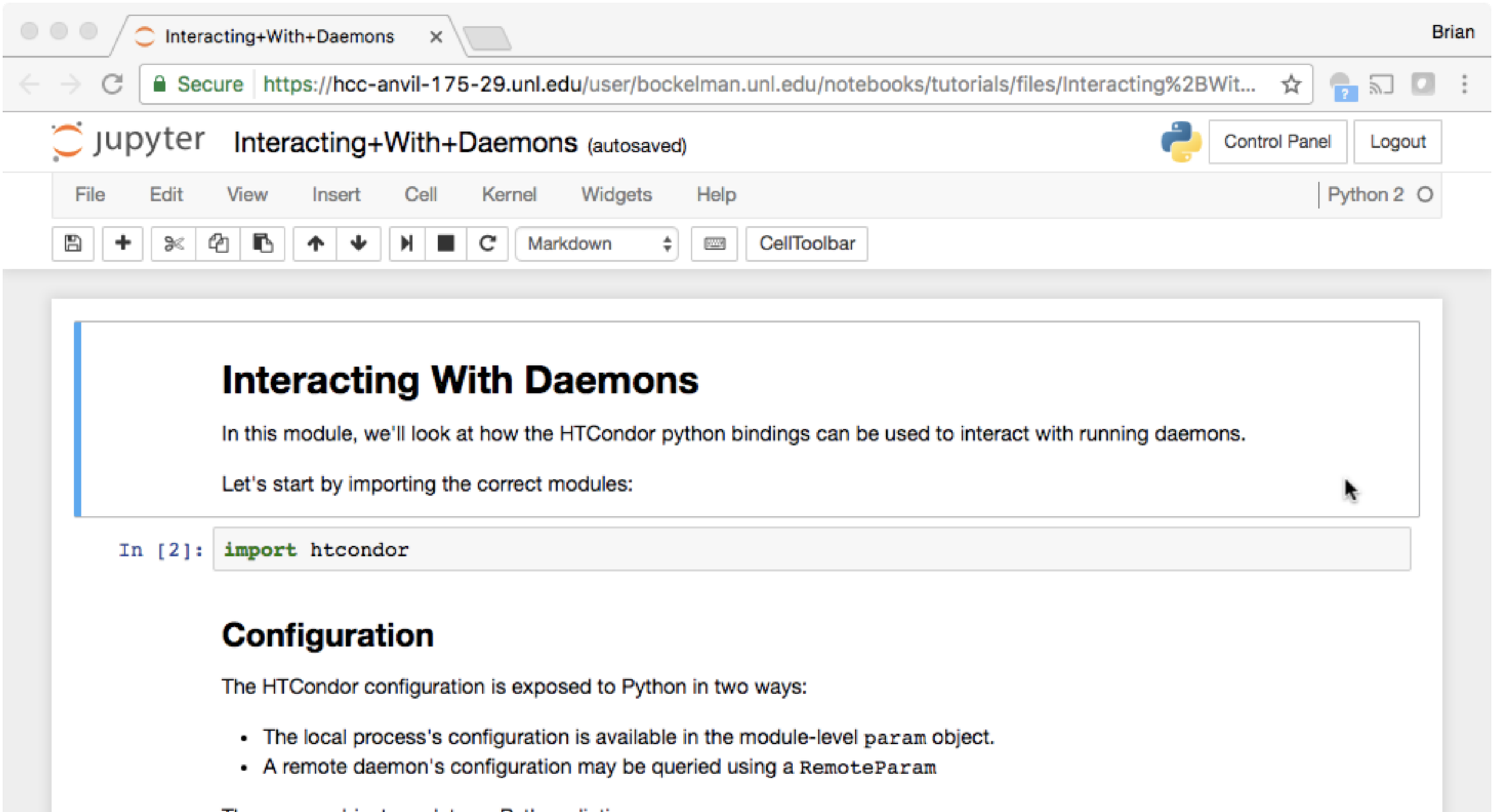
https://htcondor-python.readthedocs.io



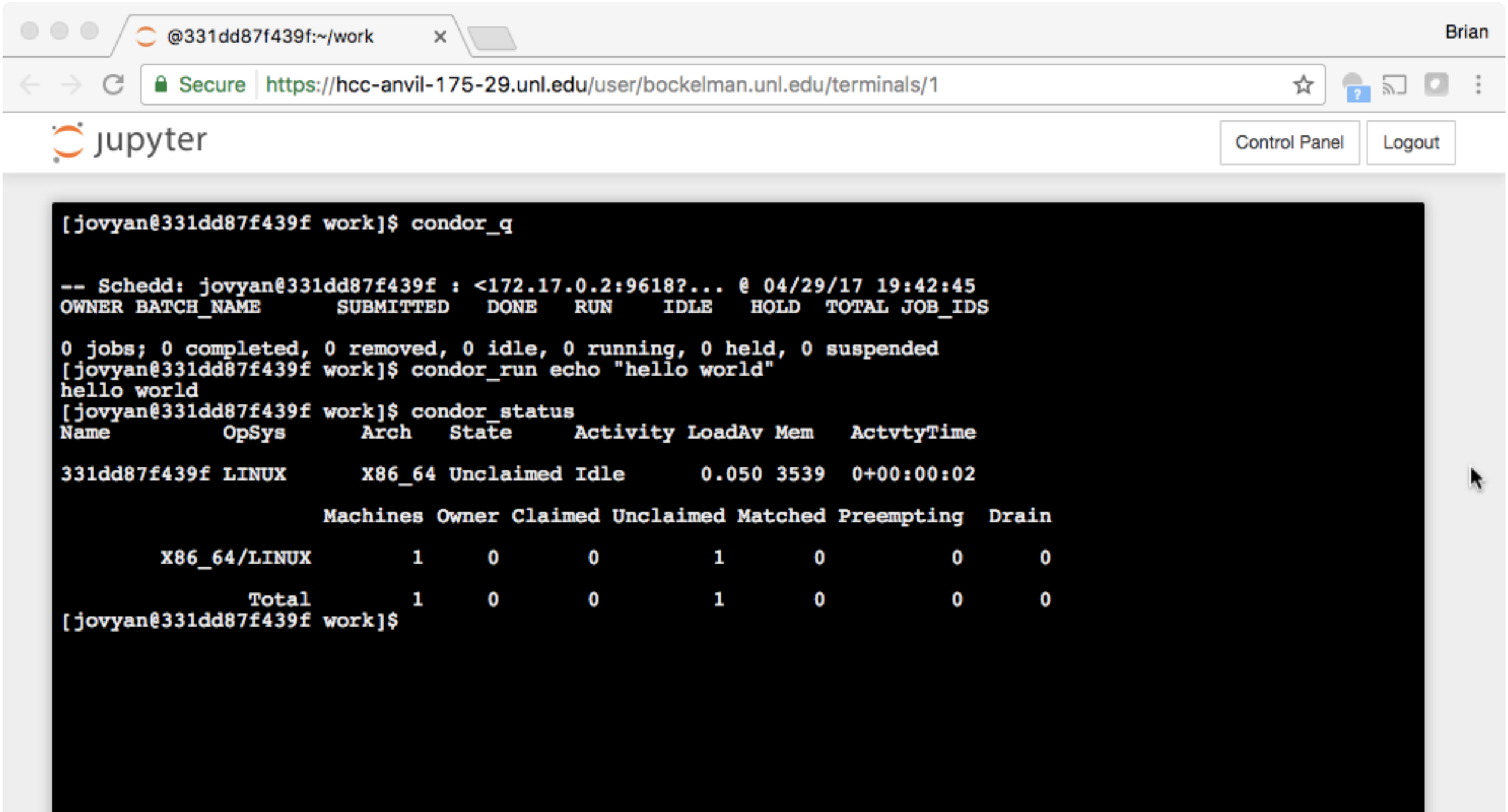
JupyterHub Tutorials



Notebook View



Terminal View



You can help!

- The contents of the tutorials and documentation are kept on GitHub:
  - https://github.com/bbockelm/htcondor-python/
    - I am interested in partnering with someone to help make the Jupyter service Docker-ized also!
- Find a bug? Spot some missing content?
  - Simply send a pull request; Travis-CI will test and update the static content once merged.

Let’s Proceed!

https://hcc-htcondor-python.unl.edu