

Jupyter, дополнительные главы

**Дмитрий Колодезев, ООО Промсофт
@d_key dk.promsoft@mail.ru
28.06.2018**

План сражения

- Где у него отладчик
- Сценарии использования
- Тигр в разрезе
- Все под контролем
- Как делиться
- Всеобщая история
- Маленькая книжка о большой памяти
- Вопросы

Panel Ipython

Ju(lia) + Py(thon) + (e)R

**Jupyter is a tool for exploring,
sharing, and discussing.**

Зачем

- Для экспериментов
- Для прототипов
- Для слайдов за столом
- Для слайдов у экрана
- Для учебников и статей в блоге
- Для неформальных отчетов
- Для встраивания в свои приложения

А где у него отладчик?



А.П. Кулик отлаживает надувной парусник

Для программиста удобнее думать о Jupyter как об отладчике, в котором можно писать документацию.

Отладка идет по ячейкам, не по строчкам.

Не делайте длинных ячеек.

Кратко про PDB:

**Здесь я запускаю
debug.irunb
и делю на ноль**

Эксперименты

- Грабь, бухай, отдыхай: заметки на салфетках.
- Импортируйте по ходу мысли
- Пишите
Код
Построчно
- Главная задача ноутбука — быстро пробовать, фиксировать что происходит.
- Делайте заметки.
- Потом делайте чистовую версию.

Эксперименты:

**Здесь я запускаю
hask.irunb
и учу плохому**

ПРОТОТИПЫ

- Ноутбуки для обсуждений и демонстраций
- часть кода не требуют обсуждения (тривиальное скачивание данных etc)
- Пароли импортируем
- Устоявшийся код перемещаем
- В ноутбуке — картинки, комментарии, интересные моменты
- «Сюда не смотри» - всё такое выкинуть

Прототипы:

**Здесь я запускаю
proto.ipynb
и делаю reload**

Обсуждение

- Самые важные презентации — за столом
- Ноутбук должен читаться
- Фокусник, достает зайцев из коробки по одному.
- WOW-эффект: картинки появляются прямо по ходу рассказа.
- Ближайший аналог - книжка-раскладушка

Презентация за столом:

**Здесь я запускаю
desk2.ipynb**

и достаю кроликов из шляпы

Слайды у экрана

- <https://damianavila.github.io/RISE/>
Презентации на основе Reveal.js
- Иногда удобно просто открыть ноутбук и показывать на экране
(проблемы в большом зале, текст мелкий)
- Ноутбуки хорошо дополняют слайды

Учебники — как надо

- ODS туториалы
https://github.com/Yorko/mlcourse_open/tree/master/jupyter_russian/tutorials
- Вероятностное программирование для хакеров
<https://github.com/CamDavidsonPilon/Probabilistic-Programming-and-Bayesian-Methods-for-Hackers>
- И вообще
<https://github.com/jupyter/jupyter/wiki/A-gallery-of-interesting-Jupyter-Notebooks>

Отчеты

- Если вы не доросли до корпоративного BI-дашборда, часто требующиеся цифры-картинки можно генерировать ноутбуками.
- Можно генерировать в кроне:

```
jupyter nbconvert --execute --to html  
notebook.ipynb
```
- Присылать все равно нужно html или pdf, т. к. их будут смотреть в почте.

Платформа

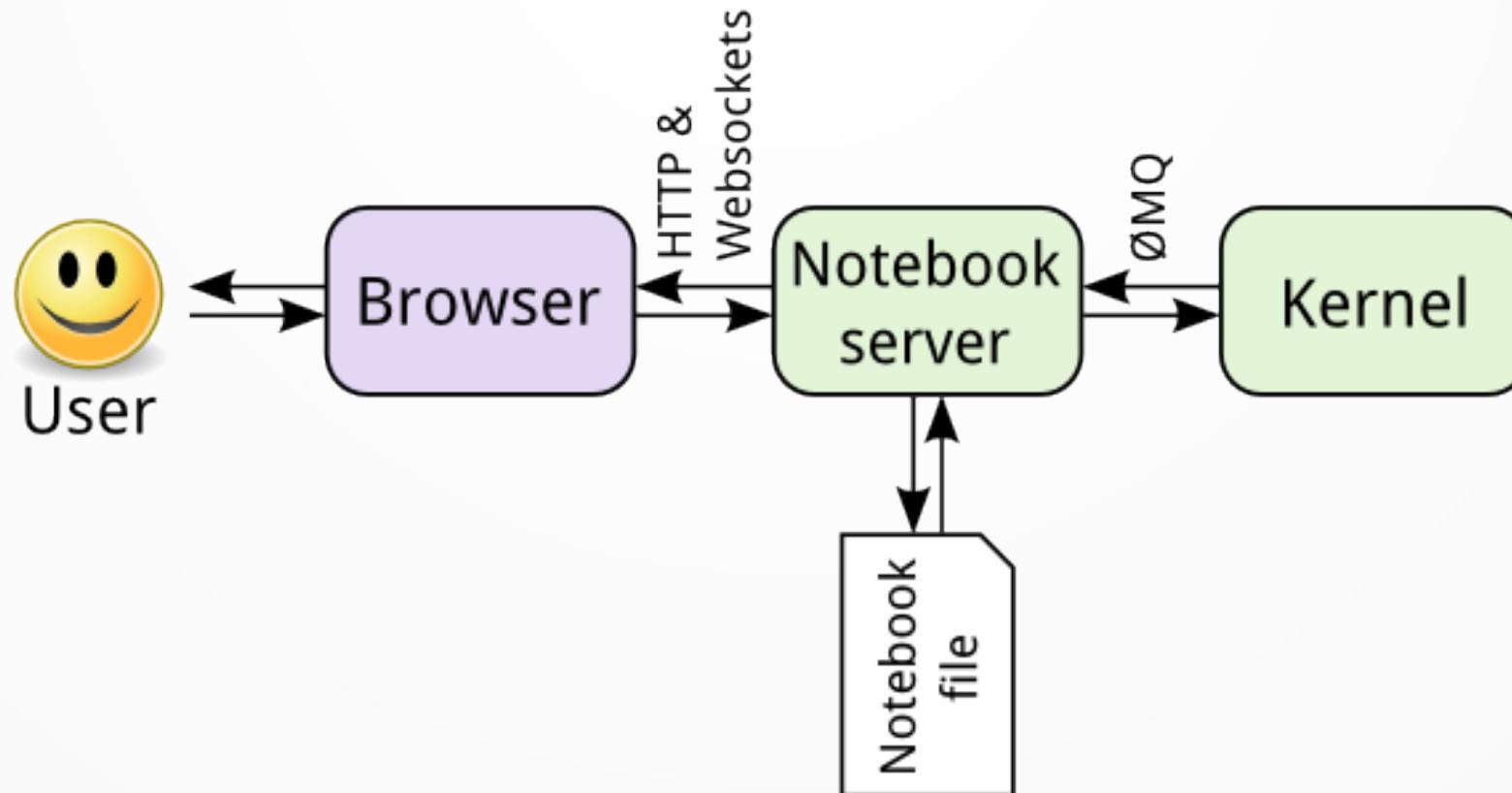
- Ipython можно вызывать из скриптов на python
гуглить embedding-ipython

```
import IPython
```

```
IPython.embed()
```

- Клиентское приложение — сервер — ядро
Любой компонент можно заменить на свой.

Честное слово, платформа



Любой из 5 элементов может быть заменен

Клиент:

- * JS-приложение notebook**
- * JS-приложение lab**
- * Консольный клиент nbconvert**
- * nteract <https://nteract.io/>**
- * qtconsole**
- * можно написать свой**

Сервер:

- * jupyter**
- * jupyterhub**
- * свои сервера у OReilly, IBM**
- * Coursera и проч.**
- * можно написать свой**

Ядро:

*** python23 R Julia Scala Go Perl PHP\
<https://github.com/jupyter/jupyter/wiki/Jupyter-kernels>**

*** Apache Toree для Spark**

*** МОЖНО написать свое
https://github.com/dsblank/simple_kernel**

КОНТРОЛЬ ВЕРСИЙ

- <https://nbdime.readthedocs.io/>
- `pip install --upgrade nbdime`
- `nbdime config-git --enable --global`
- `Nbshow`
- `Nbdiff`
- `Nbdiff-web`

Контроль версий

Здесь я открываю консоль
и делаю `git diff`

Как делиться

- GitHub и GitLab умеют рендерить ipynb
- Binder делает докер-образ из репозитория <https://mybinder.org/>
- Gist + кнопка.
Кстати, обязательно поставьте https://github.com/ipython-contrib/jupyter_contrib_nbextensions
- Собирать докер-образ jupyter+notebooks+data <https://github.com/jupyter/docker-stacks>

История

- Всё, что вы вводите в ячейки, попадает в историю
- Всё, что выводится из ячеек, попадает в историю
- In, Out
- `_1, _i1` и др.

История

**Здесь я открываю `history.ipnb`
и вижу, что все ходы записаны**

Память

- Раньше память текла сама по себе (jsonschema == 2.4.0)
- Ссылки из истории держат память
- Очевидными способами и ручным запуском gc память не очищается
- На помощь приходит магия!

Память

**Здесь я открываю memory.irpb
и мучительно пытаюсь забыть**

Не наговорился ещё

- Выучите горячие клавиши A B L O
- Поставьте nbextension
- Tqdm и interactions – ваши друзья
- Можно написать свою магию
<http://ipython.readthedocs.io/en/stable/config/custommagics.html>
- Можно написать свое расширение
<https://carreau.gitbooks.io/jupyter-book/content/notebook-extensions.html>
- См misc.ipynb

Вопросы?

@d_key

dk.promsoft@mail.ru

<https://telegram.me/Promsoft>