

# Адаптация Project Jupyter для анализа геоданных

НУГ Геоинформатики

<http://geolab.gis.land/>

**Выполнил:**

студент 3 курса группы БПИ161

**Терлыч Н.А.**

**Руководитель:**

Кандидат технических наук

доцент факультета компьютерных наук

**Родригес Залепинос Рамон Антонио**

# План презентации

1. Jupyter Notebook
2. Kernels
3. JupyterLab
4. Extensions
5. JupyterHub
6. Authenticators
7. Spawners
8. Overview

# Jupyter Notebook

интерактивная вычислительная  
среда для создания тетрадей



<https://jupyter.org/>

Уравнения

Живой Код

Графики

Видео

Текст

Виджеты

Изображения

Files



Running

Clusters

To import a notebook, drag the file onto the listing below or **click here**.

New ▾



<input type="checkbox"/> ▾	 / <b>examples</b>
<input type="checkbox"/>	 ..
<input type="checkbox"/>	 <b>Builtin Extensions</b>
<input type="checkbox"/>	 <b>Customization</b>
<input type="checkbox"/>	 <b>Embedding</b>
<input type="checkbox"/>	 <b>IPython Kernel</b>
<input type="checkbox"/>	 <b>Interactive Widgets</b>
<input type="checkbox"/>	 <b>Notebook</b>
<input type="checkbox"/>	 <b>Parallel Computing</b>
<input type="checkbox"/>	 <b>images</b>
<input type="checkbox"/>	 <b>utils</b>

jupyter Welcome to P

File Edit View Insert Cell

+

jupyter

## Welcome to the

This Notebook Server was

**WARNING**  
Don't rely on this server

Your server is hosted that

### Run some Python code

To run the code below:

1. Click on the cell to select it
2. Press **SHIFT+ENTER**

A full tutorial for using the

```
In [ ]: %matplotlib inline
import pandas as pd
import numpy as np
import matplotlib
```

jupyter Lorenz Differential Equations (autosaved)

File Edit View Insert Cell Kernel Help Python 3

Code Cell Toolbar: None

## Exploring the Lorenz System

In this Notebook we explore the [Lorenz system](#) of differential equations:

$$\begin{aligned}\dot{x} &= \sigma(y - x) \\ \dot{y} &= \rho x - y - xz \\ \dot{z} &= -\beta z + xy\end{aligned}$$

This is one of the classic systems in non-linear differential equations. It exhibits a range of complex behaviors as the parameters  $(\sigma, \beta, \rho)$  are varied, including what are known as *chaotic solutions*. The system was originally developed as a simplified mathematical model for atmospheric convection in 1963.

```
In [7]: interact(Lorenz, N=fix(10), angle=(0.,360.),
                sigma=(0.0,50.0), beta=(0.,5), rho=(0.0,50.0))
```

x

angle 308.2

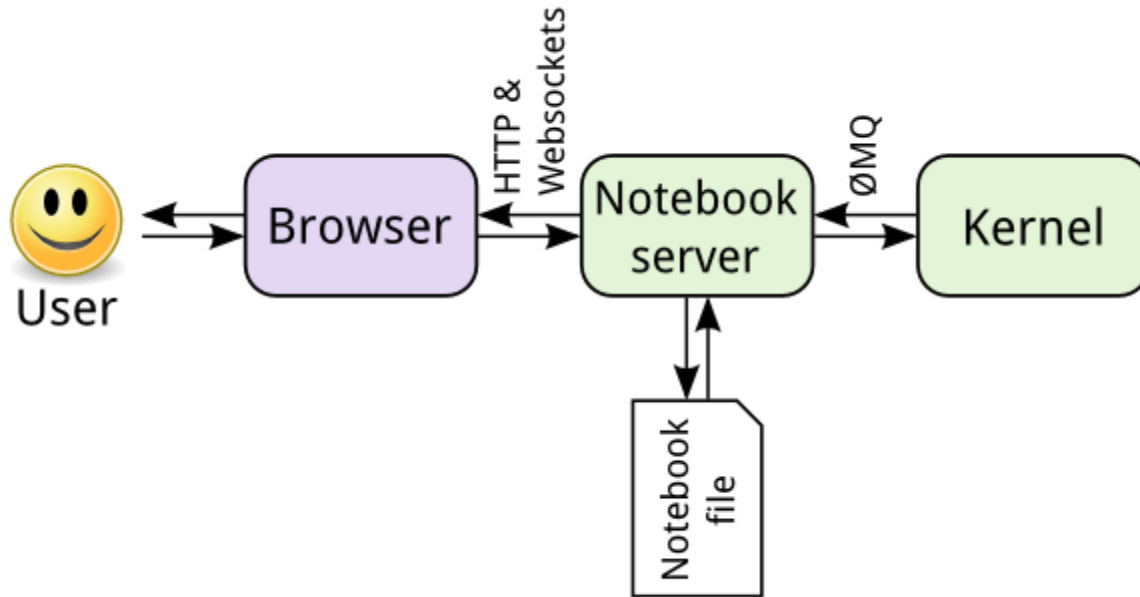
max\_time 12

$\sigma$  10

$\beta$  2.6

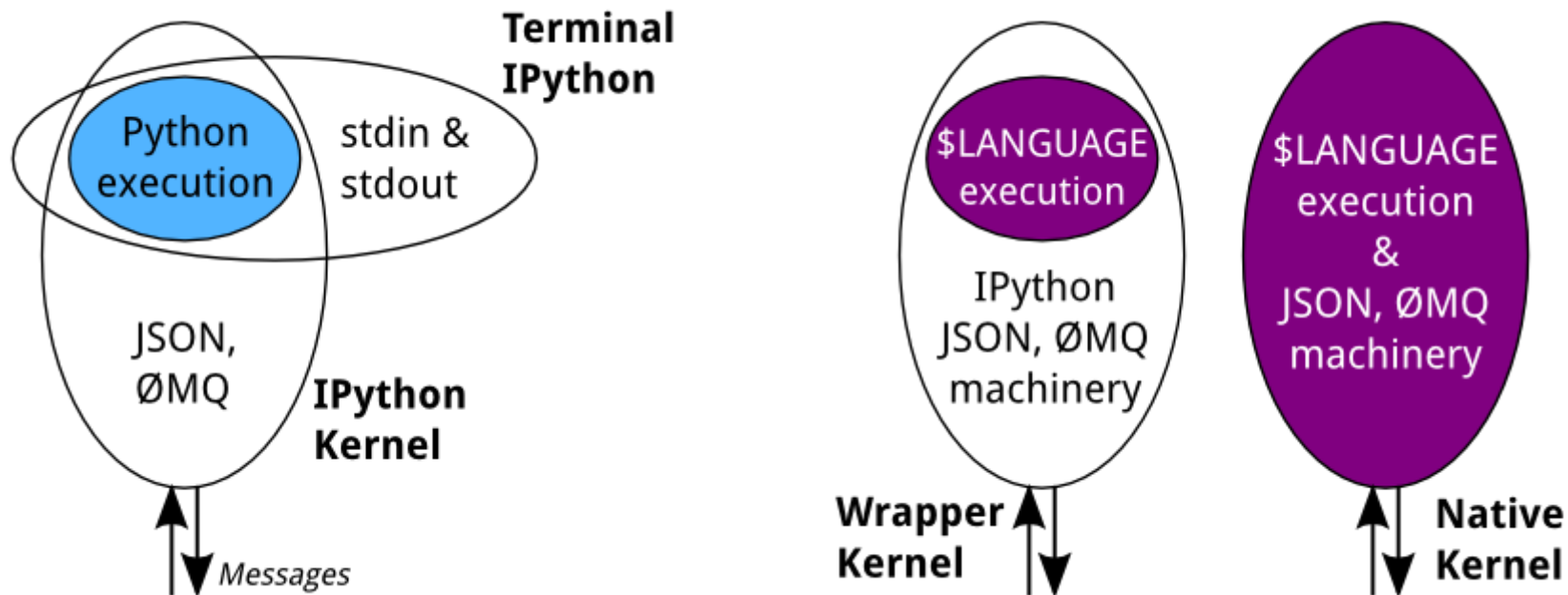
$\rho$  28

# Схема Jupyter Notebook



[https://jupyter.readthedocs.io/en/latest/architecture/how\\_jupyter\\_ipython\\_work.html](https://jupyter.readthedocs.io/en/latest/architecture/how_jupyter_ipython_work.html)

# IPython Kernel



[https://jupyter.readthedocs.io/en/latest/architecture/how\\_jupyter\\_ipython\\_work.html](https://jupyter.readthedocs.io/en/latest/architecture/how_jupyter_ipython_work.html)

# Наследование от IPython Kernel

```
from ipykernel.kernelbase import Kernel
```

```
class MyKernel(Kernel):  
    def do_execute(self, code, ...):  
        . . .  
        return content
```

```
if __name__ == '__main__':  
    from ipykernel.kernelapp import IPKernelApp  
    IPKernelApp.launch_instance(kernel_class = GeoKernel)
```



# + файл конфигурации

- **argv**: аргументы командной строки для запуска ядра.
- **display\_name**: название ядра для UI.
- **language**: название языка для ядра.
- **env** (optional): словарь переменных окружения. Будут добавлены к среде перед запуском ядра.
- **metadata** (optional): дополнительные данные, используемые клиентами тетрадей при выборе между ядрами.

```
{  
  "argv": ["python", "-m",  
           "mykernel", "-f",  
           "{connection_file}"],  
  "display_name": "MyKernel",  
  "language": "MyLanguage"  
}
```

# Выполнение кода

## `do_execute`

### Parameters:

- **code** (*str*) – Код для выполнения.
- **silent** (*bool*) – Показывать ли output.
- **store\_history** (*bool*) – Сохранять ли этот код.
- **user\_expressions** (*dict*) – Дополнительные пользовательские параметры.
- **allow\_stdin** (*bool*) – Поддерживается ли ввод во время выполнения.

```
content = {  
    # 'ok' или 'error' или 'abort'  
    'status' : str,  
    # Счётчик запусков  
    'execution_count' : self.execution_count  
}
```

# Вывод результата в ячейку

## HTML

```
self.send_response(  
    self.iopub_socket,  
    'display_data', {  
        'data': {  
            'text/html' : data  
        },  
        'metadata': {}  
    })
```

## Текст

```
self.send_response(  
    self.iopub_socket,  
    'stream', {  
        'name': 'stdout',  
        'text': data,  
        'data': {}  
    })
```

# Подсказки сигнатуры методов `Shift + Tab`

```
do_inspect(code, cursor_pos, detail_level=0)
```

```
content = {  
    # 'ok' или 'error'  
    'status' : 'ok',  
    'found' : bool,  
  
    # инспекция кода  
    'data' : dict,  
    'metadata' : dict  
}
```

```
[ ]:
```

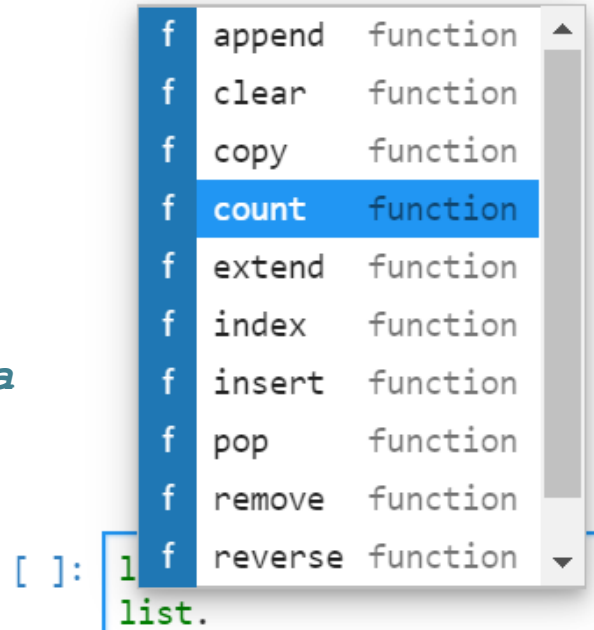
```
list  
list.append
```

```
Docstring: L.append(object) -> None -- a  
Type:      method_descriptor
```

# Автодополнение кода `Tab`

```
do_complete(code, cursor_pos)
```

```
content = {  
    # Возможные дополнения  
    'matches' : list,  
  
    # Область заменяемого кода  
    'cursor_start' : int,  
    'cursor_end' : int,  
  
    'status' : 'ok'  
}
```



# Другие методы

`history(...)` - сохранение истории запросов

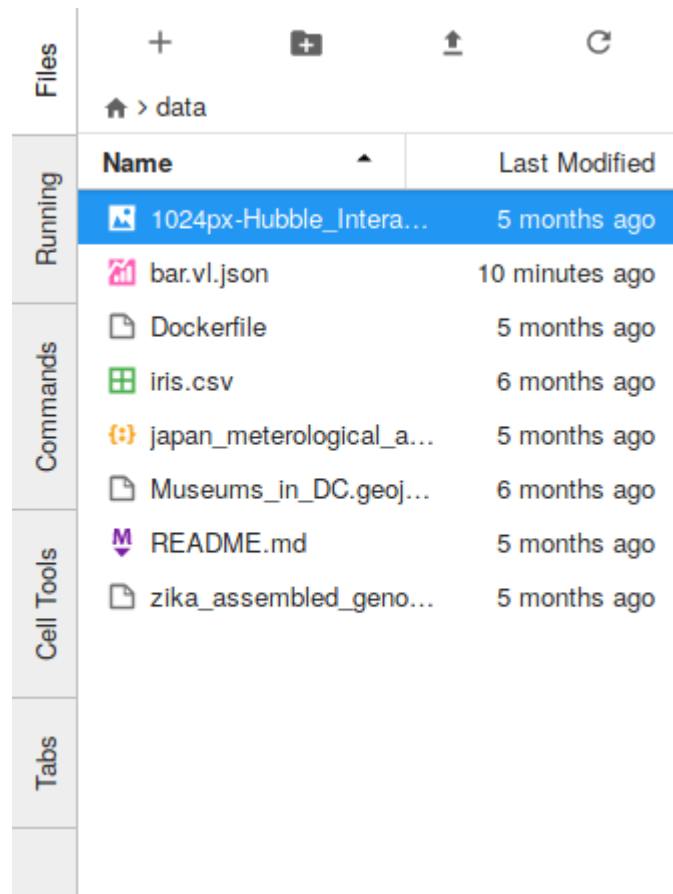
`is_complete(...)` - выполнять ли команду или ожидать дополнительный ввод (в консоли)

`connect(...)` - подключение к сокету запросов

`shutdown(...)` - выключение ядра

# Jupyter Lab

следующее поколение  
пользовательского интерфейса  
для Project Jupyter



The screenshot shows the Jupyter Lab file browser interface. The left sidebar has tabs for 'Files', 'Running', 'Commands', 'Cell Tools', and 'Tabs'. The 'Files' tab is active, showing a directory structure. The current path is 'data'. The file list includes:

Name	Last Modified
1024px-Hubble_Intera...	5 months ago
bar.vl.json	10 minutes ago
Dockerfile	5 months ago
iris.csv	6 months ago
japan_meterological_a...	5 months ago
Museums_in_DC.geoj...	6 months ago
README.md	5 months ago
zika_assembled_geno...	5 months ago

Files

+ + + ↻

🏠 > notebooks

Name	Last Modified
Data.ipynb	an hour ago
Fasta.ipynb	a day ago
Julia.ipynb	a day ago
<b>Lorenz.ipynb</b>	<b>seconds ago</b>
R.ipynb	a day ago
iris.csv	a day ago
lightning.json	9 days ago
lorenz.py	3 minutes ago

Running

Commands

Cell Tools

Tabs

Lorenz.ipynb x Terminal 1 x Console 1 x Data.ipynb x README.md x

Code Python 3

In this Notebook we explore the Lorenz system of differential equations:

$$\begin{aligned}\dot{x} &= \sigma(y - x) \\ \dot{y} &= \rho x - y - xz \\ \dot{z} &= -\beta z + xy\end{aligned}$$

Let's call the function once to view the solutions. For this set of parameters, we see the trajectories swirling around two points, called attractors.

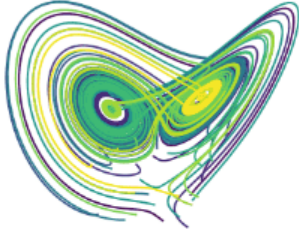
In [4]: `from lorenz import solve_lorenz`  
`t, x_t = solve_lorenz(N=10)`

Output View x

sigma  10.00

beta  2.67

rho  28.00



lorenz.py x

```

9 def solve_lorenz(N=10, max_time=4.0, sigma=10.0, beta=8./3, rho=28.0):
10     """Plot a solution to the Lorenz differential equations."""
11     fig = plt.figure()
12     ax = fig.add_axes([0, 0, 1, 1], projection='3d')
13     ax.axis('off')
14
15     # prepare the axes limits
16     ax.set_xlim((-25, 25))
17     ax.set_ylim((-35, 35))
18     ax.set_zlim((5, 55))
19
20     def lorenz_deriv(x_y_z, t0, sigma=sigma, beta=beta, rho=rho):
21         """Compute the time-derivative of a Lorenz system."""
22         x, y, z = x_y_z
23         return [sigma * (y - x), x * (rho - z) - y, x * y - beta * z]
24
25     # Choose random starting points, uniformly distributed from -15 to 15
26     np.random.seed(1)
27     x0 = -15 + 30 * np.random.random((N, 3))
28

```



# Расширения

## Добавляют функционал:

- элементы меню
- палитра команд
- горячие клавиши
- настройки среды
- API для других расширений

- **application plugins:**

расширяют функционал самого JupyterLab

- **mime renderer extensions:**

отвечают за рендеринг различных типов данных

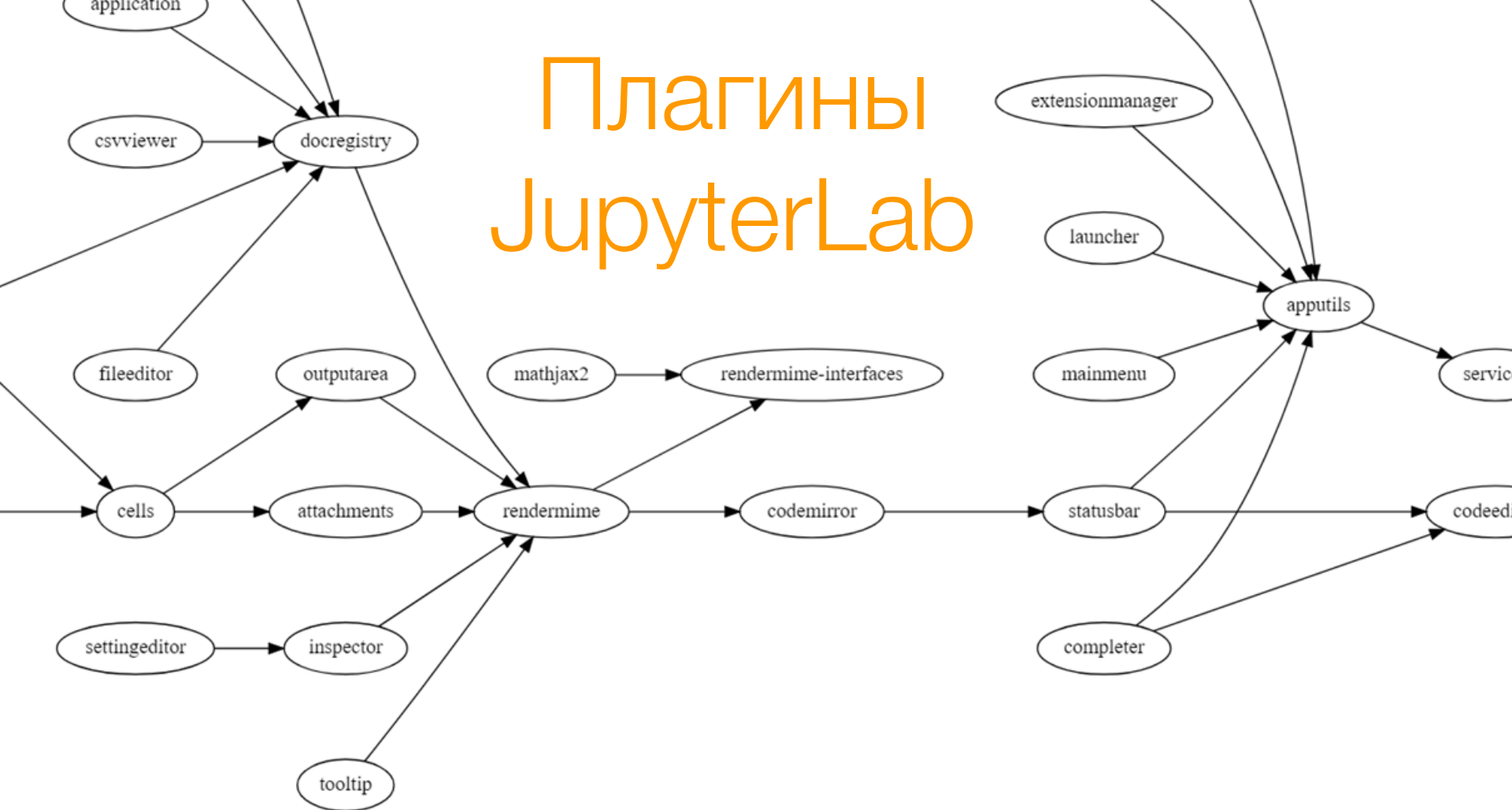
- **theme extensions:**

настраивают внешний вид JupyterLab

- **document widget extensions:**

описывают логику обработки документов

# Плагины JupyterLab



# Подсветка синтаксиса



This is CodeMirror

```
1 <!-- Create a simple CodeMirror instance -->
2 <link rel="stylesheet" href="lib/codemirror.css">
3 <script src="lib/codemirror.js"></script>
4 <script>
5     var editor = CodeMirror.fromTextArea(myTextarea, {
6         lineNumbers: true
7     });
8 </script>
```

<https://codemirror.net/>

# SimpleMode

**Codemirror** позволяет определять моды для языков

**SimpleMode** - аддон, позволяющий создать упрощённый CodeMirror мод для языка, описывая регулярными выражениями:

- комментарии {regex: `/\s*\s*\s*/?.*$/`, token: `'comment'`}
- ключевые слова {regex: `/\sfor/`, token: `'keyword'`}
- операторы {regex: `/-|>|&|!=/`, token: `'operator'`}
- типы данных {regex: `/"/`, token: `'string'`}
- ...

# Виджет для подсветки

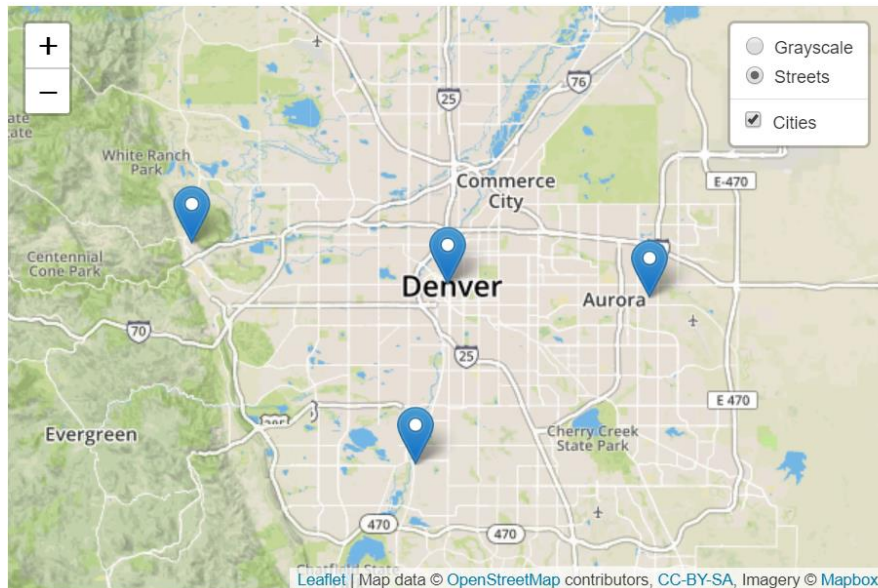
```
const extension: JupyterLabPlugin<void> = {  
  id: 'geo-extension',  
  autoStart: true,  
  requires: [ILayoutRestorer],  
  activate: () => {  
    defineGeoMode();  
  }  
};  
export default extension;
```

# Добавление карты



<https://leafletjs.com/>

Leaflet - открытая  
JavaScript библиотека для  
интерактивных карт



```
export class RenderedGeoJSON extends Widget implements
IRenderMime.IRenderer {
    renderModel(model: IRenderMime.IMimeModel): Promise<void> {
        return new Promise<void>((resolve, reject) => {
            // Добавление слоя в leaflet
            leaflet.tileLayer(url, options).addTo(this._map);
            // Добавление слоя GeoJSON по данным на карту
            this._geoJSONLayer = leaflet.geoJSON(model.data)
                .addTo(this._map);
            resolve();
        });
    }
}
```

# Фабрика отрисовки GeoJson

```
export const MIME_TYPE = 'application/geo+json';

export const rendererFactory: IRenderMime.IRendererFactory = {
  safe: true,
  mimeTypes: [MIME_TYPE],
  createRenderer: options => new RenderedGeoJSON(options)
};
```

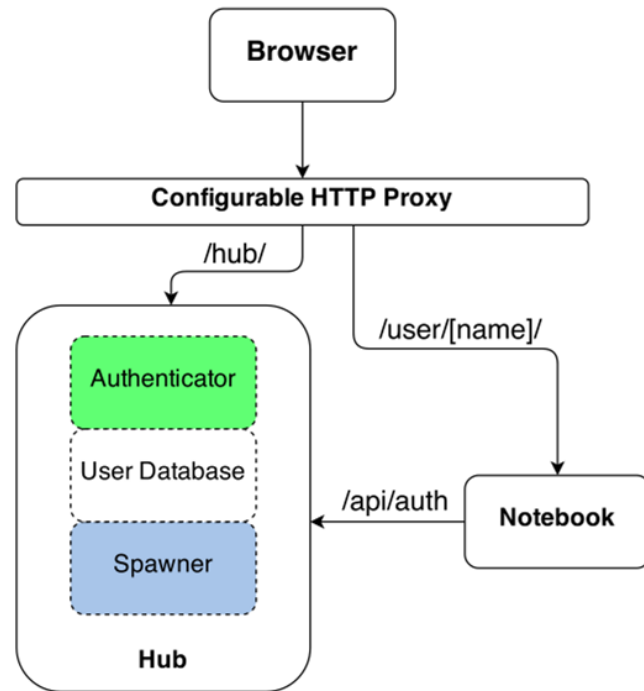


```
const extensions: IRenderMime.IExtension[] = [{
  id: 'geo-map-extension',
  rendererFactory,
  rank: 0,
  dataType: 'json',
  fileTypes: [...],
  documentWidgetFactoryOptions: {
    name: 'GeoJSON',
    primaryFileType: 'geojson',
    fileTypes: ['geojson', 'json'],
    defaultFor: ['geojson']
  }
}];
```





**JupyterHub** позволяет  
использовать всю мощь Jupyter  
Notebook для групп  
пользователей



<https://jupyterhub.readthedocs.io/en/stable/>

# Аутентификаторы

Аутентификаторы  
реализуют  
авторизацию  
пользователей в Hub  
и запуск  
пользовательских  
notebook-серверов

- Auth0
- Bitbucket
- GitHub
- GitLab
- Globus
- Google
- OpenShift

# Native Authenticator

## Sign Up

Username:

Password:

Create User

Something went wrong. Be sure your password has at least 10 characters and is not too common.

<https://native-authenticator.readthedocs.io/en/latest/options.html>

### Sign in

**Username:**

**Password:**

**Sign In**

# Spawners

Spawner запускает notebook-сервер для каждого пользователя.

- запускает процесс
- запрашивает статус процесса
- останавливает процесс

# Spawners

- [DockerSpawner](#) запускает сервера в Docker контейнерах
- [SudoSpawner](#) позволяет JupyterHub запускаться не из под root'а, запуская процессы через sudo
- [BatchSpawner](#) запускает удалённые сервера используя batch-системы
- [RemoteSpawner](#) запускает удалённые notebook-сервера через SSH



# Настройка конфигурации

```
# Говорит запускать JupyterLab по умолчанию
c.Spawner.default_url = '/lab'
c.Spawner.cmd = ['jupyter-labhub']
# Назначает аутентификатор
c.JupyterHub.authenticator_class = '...'
c.NativeAuthenticator.open_signup = True
# Назначает spawner
c.JupyterHub.spawner_class = '...'
# Добавляет имена администраторов
c.Authenticator.admin_users = {'adrax'}
```

# Сборка своего Jupyter

## JupyterHub

- Выбрать **Authenticator**
- Выбрать **Spawner**
- Настроить конфигурацию
- Поставить расширение для **JupyterLab**

## JupyterLab

- Написать **Kernel** для языка
- Добавить расширение **CodeMirror** для подсветки синтаксиса
- Добавить расширение **IRenderMime** для своего типа данных

Спасибо за внимание!