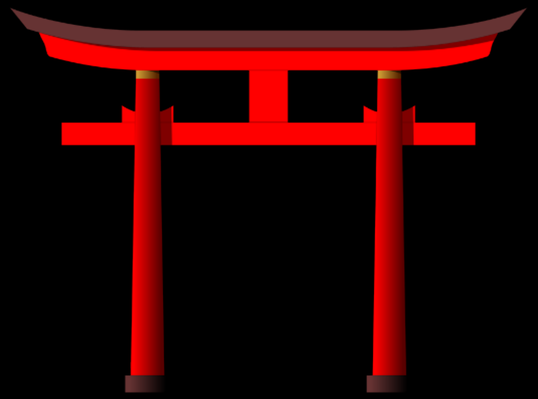


Apache Toree

A Jupyter Kernel for Scala and Apache Spark

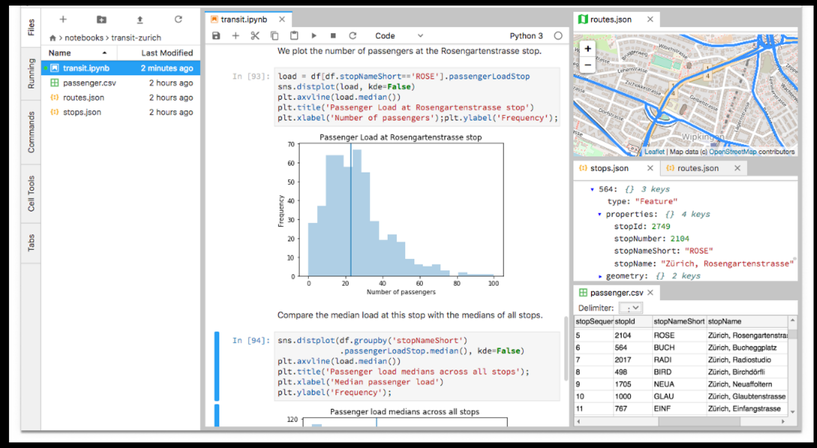
Luciano Resende
Apache Torree committer and PPMC member | Apple
ApacheCon North America 2022

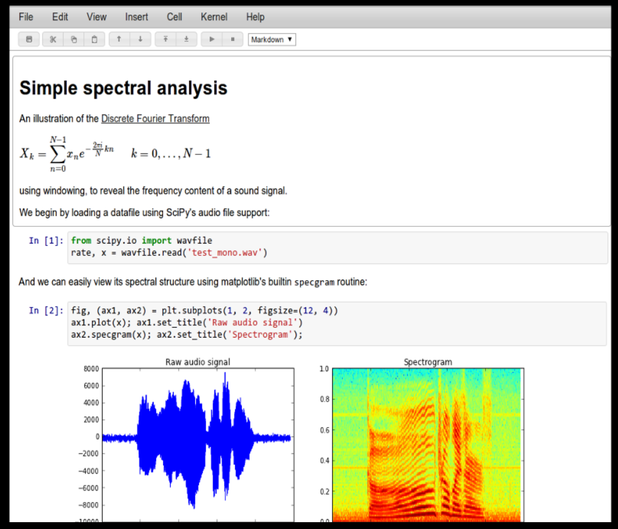
THIS IS NOT A CONTRIBUTION

Jupyter Notebooks

Jupyter Notebooks

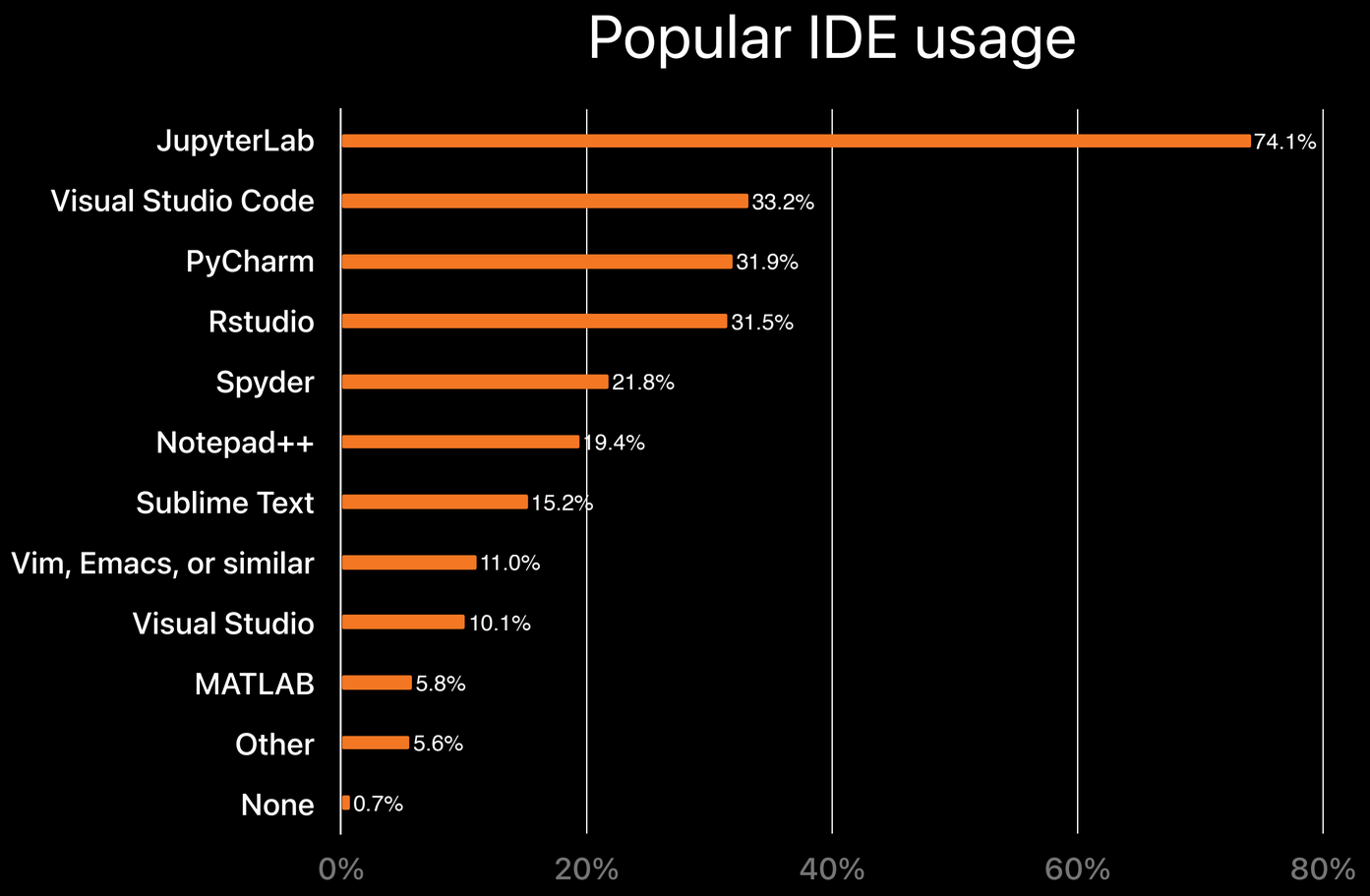
Notebooks are interactive computational environments, in which you can combine code execution, rich text, mathematics, plots and rich media





Jupyter Notebooks

Jupyter Ecosystem is
the de-facto standard tool in data science and AI community



Jupyter Notebooks

Simple, but Powerful

As simple as opening a web page, with the capabilities of a powerful, multilingual, development environment.

Interactive widgets

Code can produce rich outputs such as images, videos, markdown, LaTeX and JavaScript. Interactive widgets can be used to manipulate and visualize data in real-time.

Language of choice

Jupyter Notebooks have support for over 50 programming languages, including those popular in Data Science, Data Engineer, and AI such as Python, R, Julia and Scala.

Big Data Integration

Leverage Big Data platforms such as Apache Spark from Python, R and Scala. Explore the same data with pandas, scikit-learn, ggplot2, dplyr, etc.

Share Notebooks

Notebooks can be shared with others using e-mail, Dropbox, Google Drive, GitHub, etc

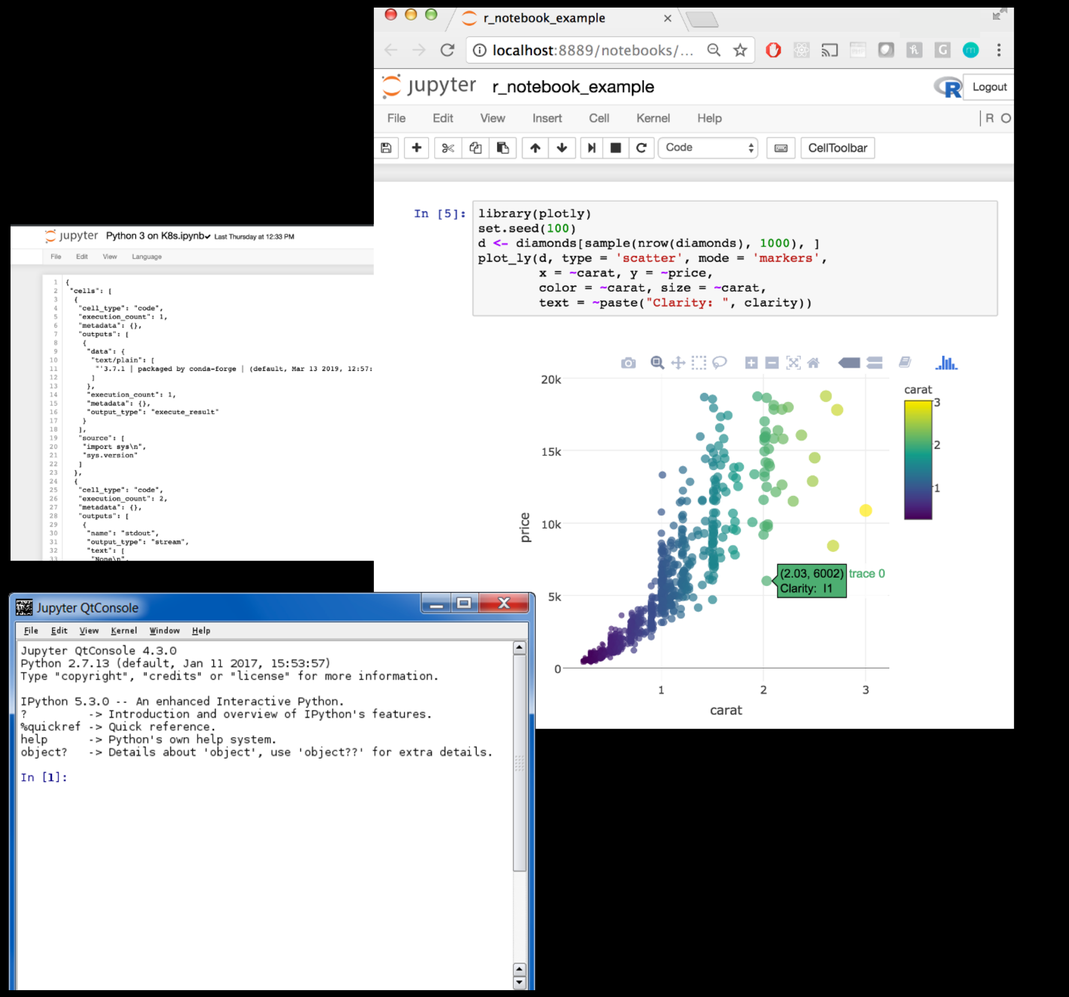
Jupyter Notebooks

Single page web interface

- File Browser
- Code Console (QT Console)
- Text Editor

Current Release

- Jupyter Notebook 6.4.12
- Available in Anaconda
- pip install --upgrade notebook



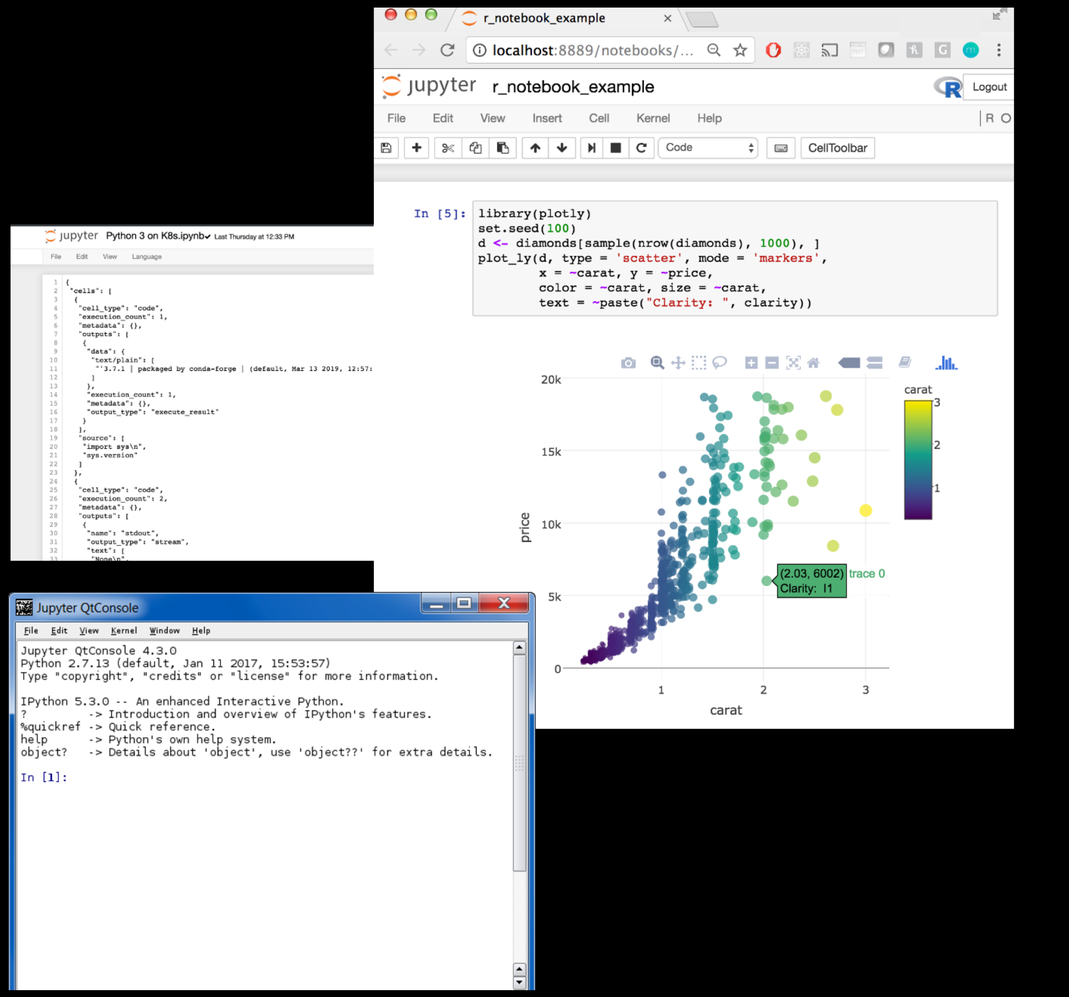
Jupyter Notebooks

The Classic Notebook is starting to move towards maintenance mode

- Community efforts being concentrated in the new JupyterLab UI.
- Community continue to deliver bug-fixes and security updates frequently

Jupyter 7.0 (based on JupyterLab) discussion:

- https://jupyter.org/enhancement-proposals/79-notebook-v7/notebook-v7.html



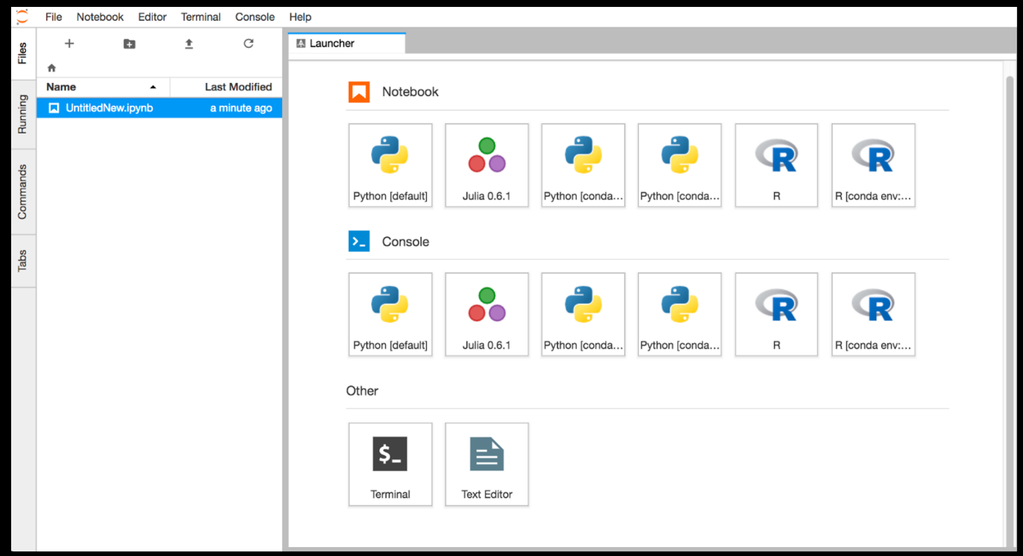
JupyterLab

JupyterLab is the next generation UI for the Jupyter Ecosystem.

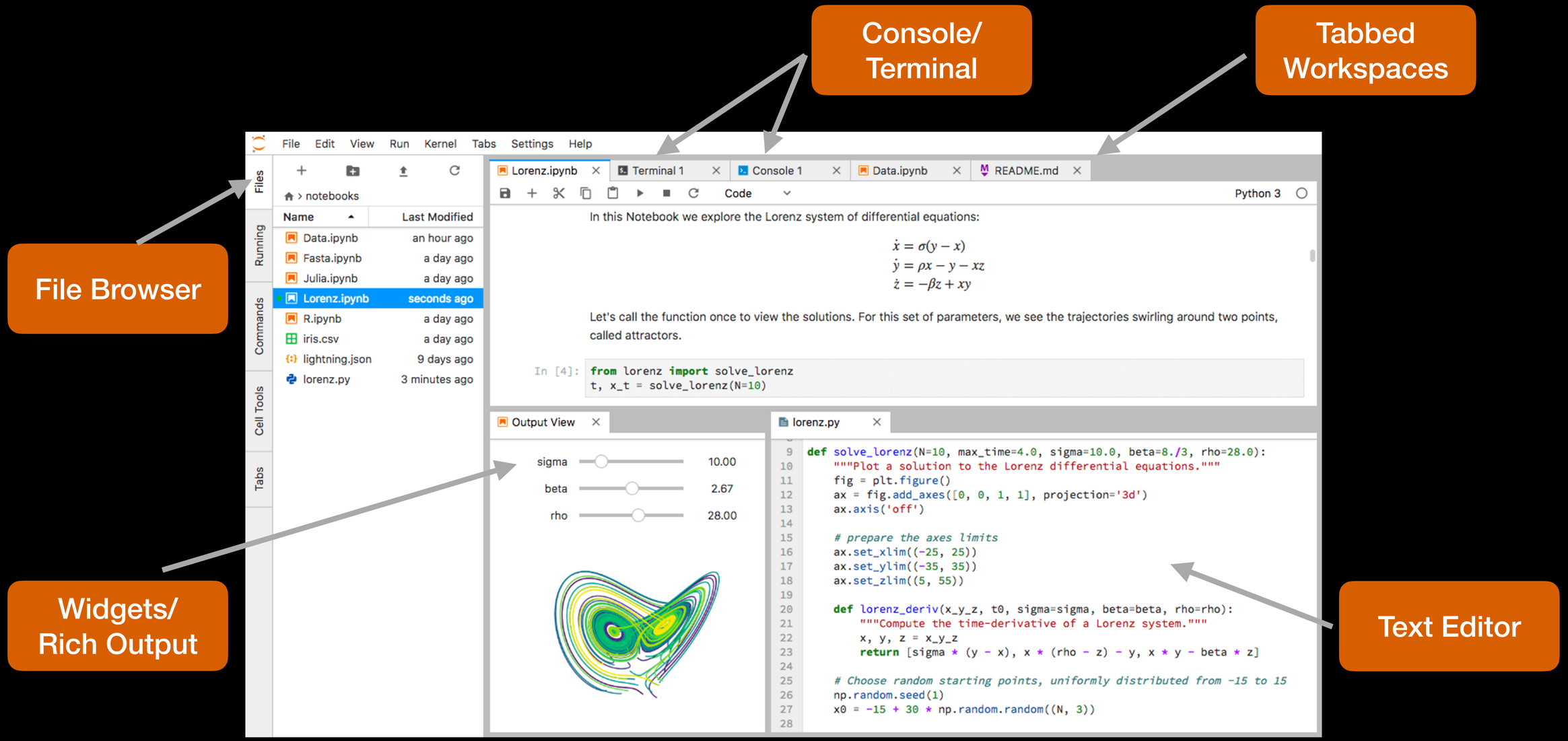
Brings all the previous improvements into a single unified platform plus more!

Provides a modular, extensible architecture

Retains backward compatibility with the old notebook we know and love



JupyterLab


Console/
Terminal
Tabbed
Workspaces
File Browser
Widgets/
Rich Output
Text Editor
File Edit View Run Kernel Tabs Settings Help
Files
Running
Commands
Cell Tools
Tabs
notebooks
Name
Last Modified
Data.ipynb an hour ago
Fasta.ipynb a day ago
Julia.ipynb a day ago
Lorenz.ipynb seconds ago
R.ipynb a day ago
iris.csv a day ago
lightning.json 9 days ago
lorenz.py 3 minutes ago
Lorenz.ipynb
Terminal 1
Console 1
Data.ipynb
README.md
Code
Python 3
In this Notebook we explore the Lorenz system of differential equations:
$\dot{x} = \sigma(y - x)$
$\dot{y} = \rho x - y - xz$
$\dot{z} = -\beta z + xy$
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
sigma 10.00
beta 2.67
rho 28.00
lorenz.py
def solve_lorenz(N=10, max_time=4.0, sigma=10.0, beta=8./3, rho=28.0):
"""Plot a solution to the Lorenz differential equations."""
fig = plt.figure()
ax = fig.add_axes([0, 0, 1, 1], projection='3d')
ax.axis('off')
# prepare the axes limits
ax.set_xlim((-25, 25))
ax.set_ylim((-35, 35))
ax.set_zlim((5, 55))
def lorenz_deriv(x_y_z, t0, sigma=sigma, beta=beta, rho=rho):
"""Compute the time-derivative of a Lorenz system."""
x, y, z = x_y_z
return [sigma * (y - x), x * (rho - z) - y, x * y - beta * z]
# Choose random starting points, uniformly distributed from -15 to 15
np.random.seed(1)
x0 = -15 + 30 * np.random.random((N, 3))

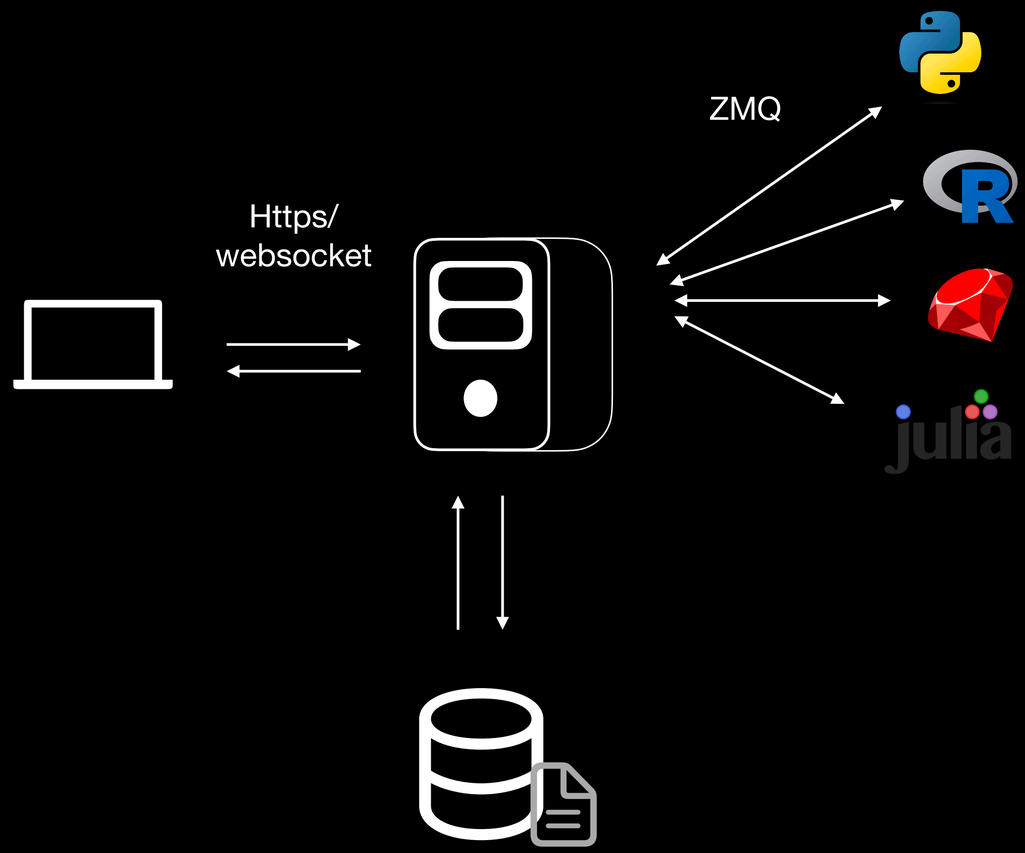
Jupyter Notebooks

Notebook UI runs on the browser

The Notebook Server serves the ‘Notebooks’

Kernels interpret/execute cell contents

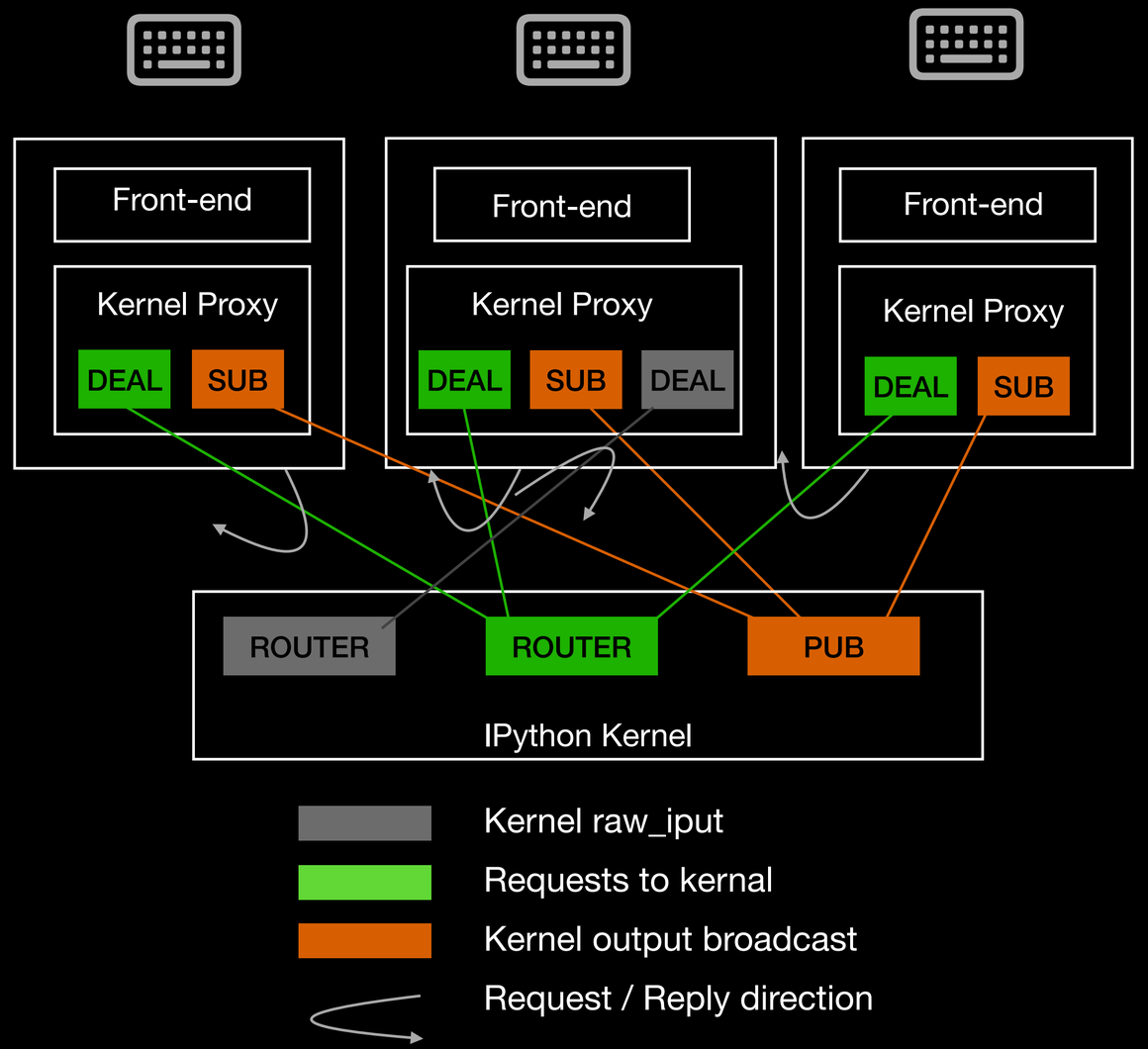
- Are responsible for code execution
- Abstracts different languages
- 1:1 relationship with Notebook
- Runs and consume resources as long as notebook is running



Jupyter Notebooks

Available Sockets:

- Shell (requests, history, info)
- IOPub (status, display, results)
- Stdin (input requests from kernel)
- Control (shutdown, interrupt)
- Heartbeat (poll)



Jupyter Notebooks

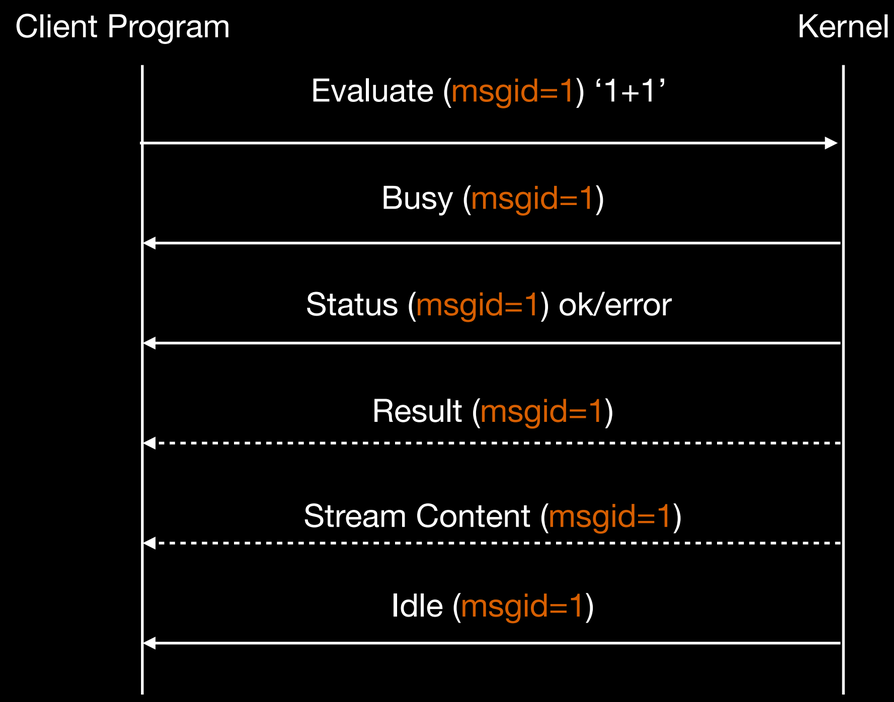
Two types of responses

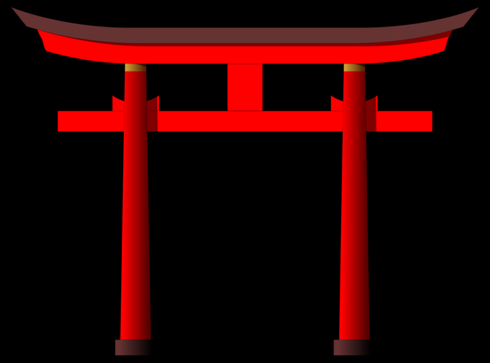
Results

- Computations that return a result
- 1+1
- val a = 2 + 5

Stream Content

- Values that are written to output stream
- df.show(10)





Apache Toree

Apache Toree

A Scala based Jupyter Kernel that enables Jupyter Notebooks to execute Scala code and connect to Apache Spark to build interactive applications.

Apache Toree History

- December 2014 - Open Sourced Spark Kernel to GitHub
- July 2015 - Joined developerWorks Open
- https://developer.ibm.com/open/spark-kernel/
- December 2015 - Accepted as an Apache Incubator Project
- https://toree.apache.org

Apache Toree Releases

Release	Scala Version	Spark Version
Toree 0.1.x	Scala 2.11	Spark 1.6
Toree 0.2.x - 0.4.x	Scala 2.11	Spark 2.x
Toree 0.5.x	Scala 2.12	Spark 3.x

Apache Toree

Installing the Toree Kernel

- pip install –upgrade toree

Configuring the Toree Kernel

- jupyter toree install --spark_home=/usr/local/bin/apache-spark/

Apache Toree

```
{
  "argv": [
    "/usr/local/share/jupyter/kernels/apache_toree_scala/bin/run.sh",
    "--profile",
    "{connection_file}"
  ],
  "env": {
    "DEFAULT_INTERPRETER": "Scala",
    "__TOREE_SPARK_OPTS__": "",
    "__TOREE_OPTS__": "",
    "SPARK_HOME": "/Users/lresende/opt/spark-3.2.2-bin-hadoop2.7/",
    "PYTHONPATH": “/Users/lresende/opt/spark-3.2.2-bin-hadoop2.7//python:/Users/lresende/opt/spark-3.2.2-bin-hadoop2.7/python/lib/py4j-0.10.9.5-src.zip”,
    "PYTHON_EXEC": "python"
  },
  "display_name": "Apache Toree - Scala",
  "language": "scala",
  "interrupt_mode": "signal",
  "metadata": {}
}
```

Apache Toree

```
{
  "argv": [
    “/usr/local/share/jupyter/kernels/apache_toree_scala/bin/run.sh",
    "--spark-context-initialization-mode",
    "none",
    "--profile",
    "{connection_file}"
  ],
  "env": {
    "DEFAULT_INTERPRETER": "Scala",
    "__TOREE_SPARK_OPTS__": "",
    "__TOREE_OPTS__": "",
    "SPARK_HOME": "/Users/lresende/opt/spark-3.2.2-bin-hadoop2.7/",
    "PYTHONPATH": “/Users/lresende/opt/spark-3.2.2-bin-hadoop2.7//python:/Users/lresende/opt/spark-3.2.2-bin-hadoop2.7/
python/lib/py4j-0.10.9.5-src.zip”,
    "PYTHON_EXEC": "python"
  },
  "display_name": "Apache Toree - Scala",
  "language": "scala",
  "interrupt_mode": "signal",
  "metadata": {}
}
```

Apache Toree

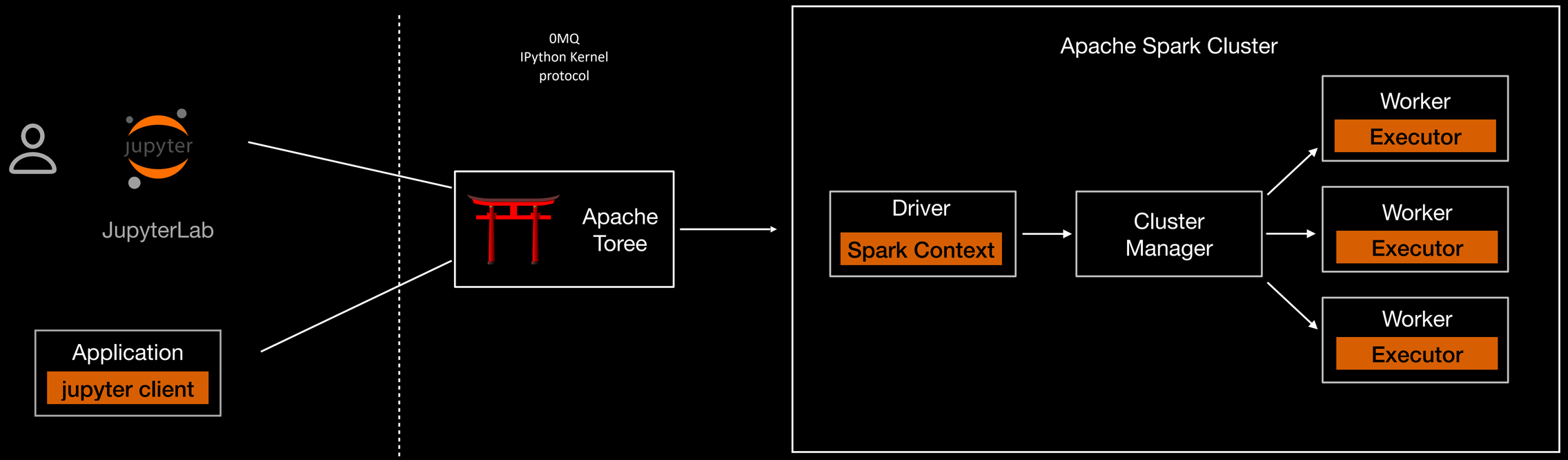
```
{
  "language": "scala",
  "display_name": "Spark - Scala (YARN Cluster Mode)",
  "metadata": {  "process_proxy": {  "class_name":
"enterprise_gateway.services.processproxies.yarn.YarnClusterProcessProxy”  }  },
  "env": {
    "SPARK_HOME": "/usr/hdp/current/spark2-client",
    "__TOREE_SPARK_OPTS__": "--master yarn --deploy-mode cluster --name ${KERNEL_ID:-ERROR__NO__KERNEL_ID} --conf
spark.yarn.submit.waitAppCompletion=false --conf spark.yarn.am.waitTime=1d ${KERNEL_EXTRA_SPARK_OPTS}",
    "__TOREE_OPTS__": "--alternate-sigint USR2",
    "DEFAULT_INTERPRETER": "Scala"
  },
  "argv": [
    "/usr/local/share/jupyter/kernels/spark_scala_yarn_cluster/bin/run.sh",
    "--RemoteProcessProxy.kernel-id",
    "{kernel_id}",
    "--RemoteProcessProxy.response-address",
    "{response_address}",
   "--RemoteProcessProxy.spark-context-initialization-mode",
    "lazy"
  ]
}
```

Apache Toree

```
PROG_HOME="$(cd "`dirname "$0"`"/..; pwd)"
if [ -z "$SPARK_HOME" ]; then
  echo "SPARK_HOME must be set to the location of a Spark distribution!"
  exit 1
fi
echo "Starting Spark Kernel with SPARK_HOME=$SPARK_HOME"
KERNEL_ASSEMBLY=`(cd ${PROG_HOME}/lib; ls -1 toree-assembly-*.jar;)`
# disable randomized hash for string in Python 3.3+
TOREE_ASSEMBLY=${PROG_HOME}/lib/${KERNEL_ASSEMBLY}
eval exec \
     "${SPARK_HOME}/bin/spark-submit" \
     --name "'Apache Toree'" \
     "${SPARK_OPTS}" \
     --class org.apache.toree.Main \
     "${TOREE_ASSEMBLY}" \
     "${TOREE_OPTS}" \
     "$@"
```

Apache Toree Architectural Diagram

Apache Toree running as an Apache Spark application in client mode



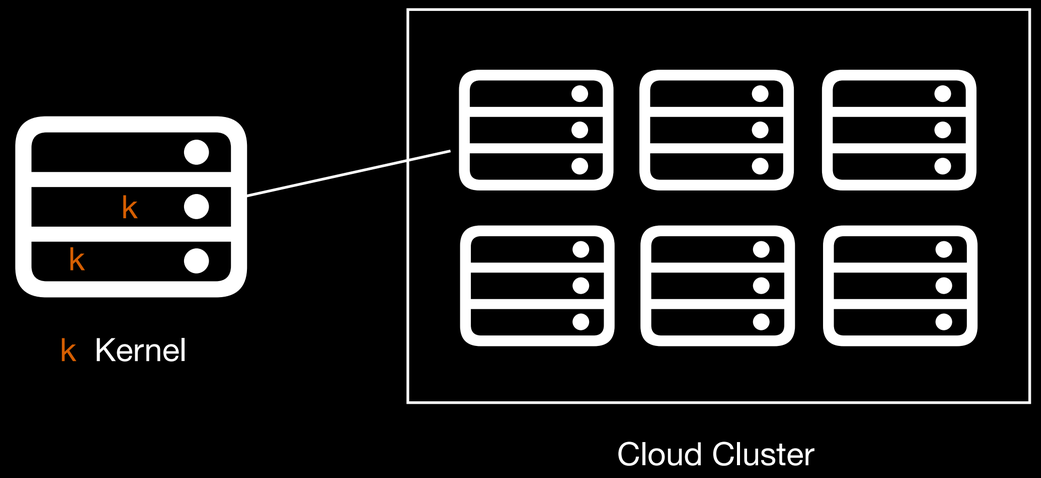
Jupyter Notebooks
Stack Limitation

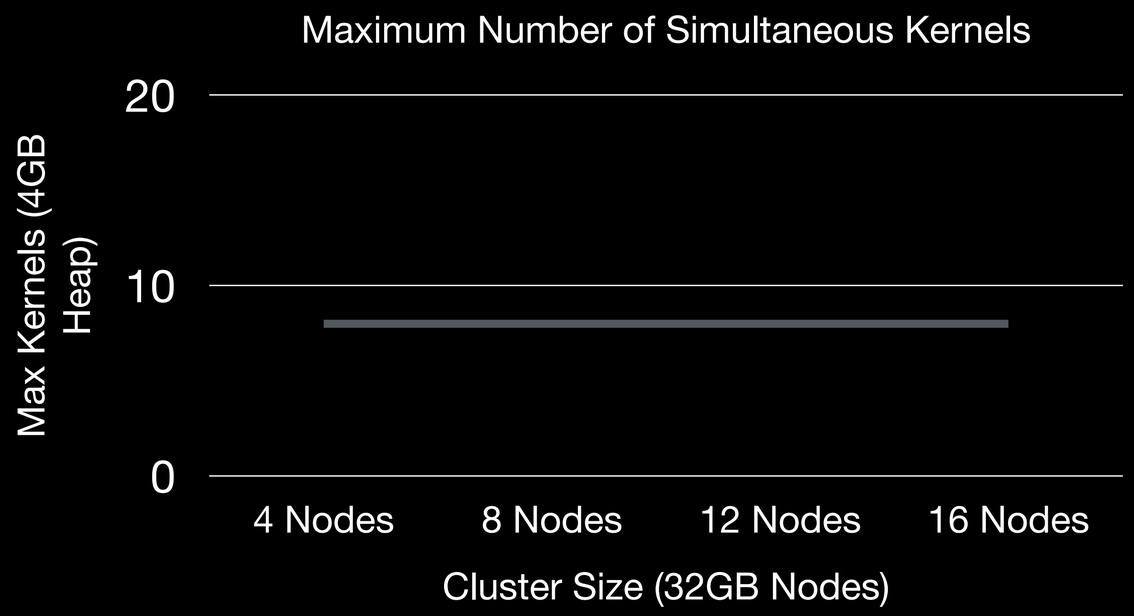
Scalability

- Jupyter Kernels running as local process
- Resources are limited by what is available on the one single node that runs all Kernels and associated Spark drivers

Security

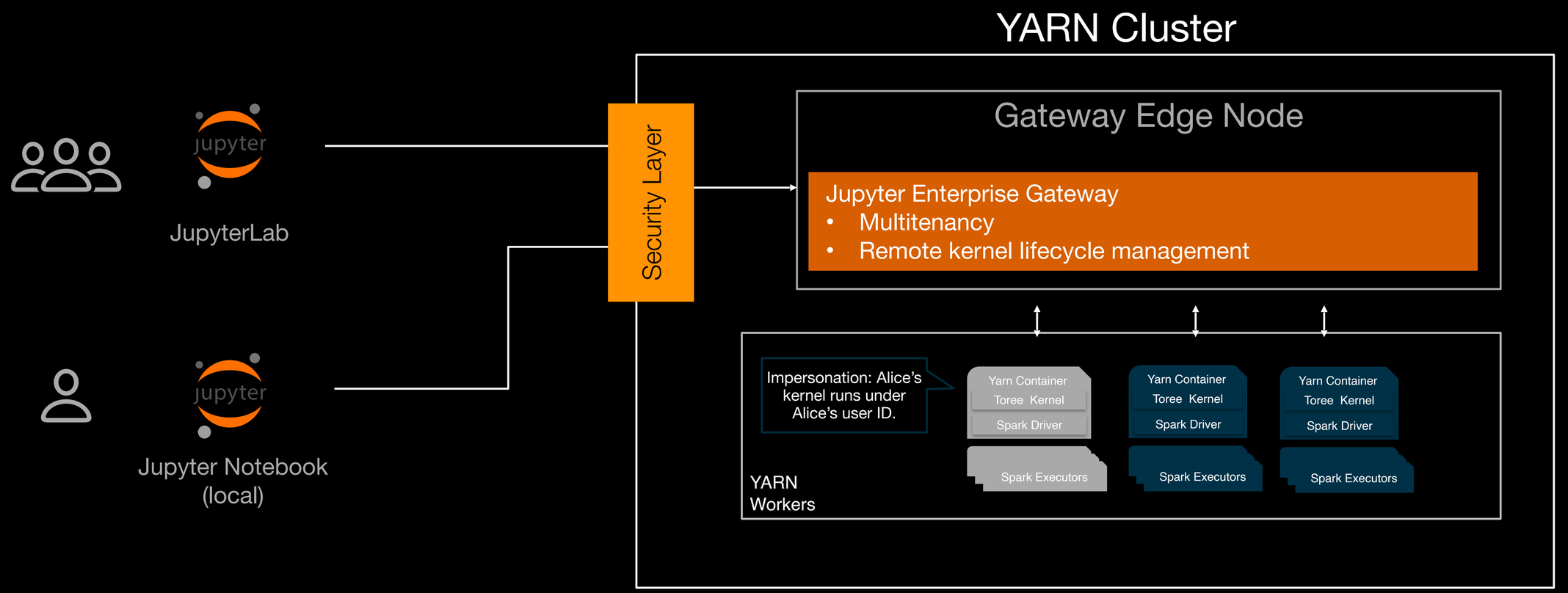
- Single user sharing the same privileges
- Users can see and control each other process using Jupyter administrative utilities

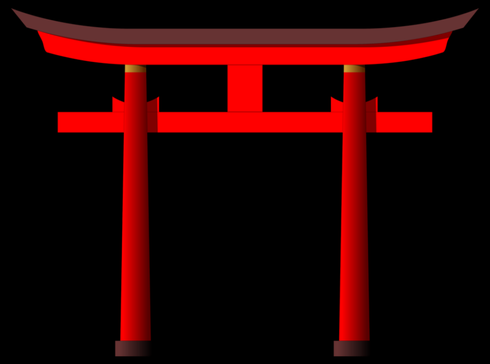




Apache Toree Architectural Diagram

Apache Toree running as an Apache Spark application in cluster mode





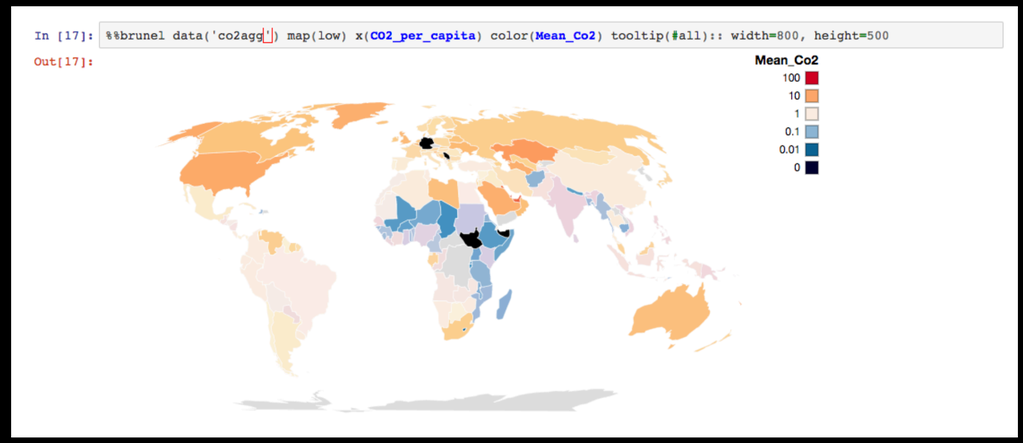
Apache Toree Add-ons

Apache Toree Visualizations

Apache Toree, via extensions like Brunel for Apache Toree or Plotly for Scala, supports rich visualizations that integrates directly with Spark Data Frame APIs

Notes:
Brunel seems to be broken with Spark 3.2.2 (from previous 2.x/3.0.x)

Plotly seems to have a dependency issue (https://github.com/alexarchambault/plotly-scala/issues/14)



Apache Toree Visualizations

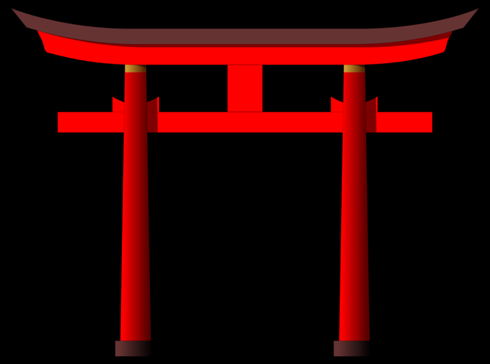
Apache Toree provides a set of magics that enhances the user experience manipulating data coming from Spark tables or data

```
In [6]: %%sql
        select * from customers

Out[6]: +---+-------------+-------+---------+-------+
        |age|          job|marital|education|balance|
        +---+-------------+-------+---------+-------+
        | 30|   unemployed|married|  primary|   1787|
        | 33|     services|married|secondary|   4789|
        | 35|   management| single| tertiary|   1350|
        | 30|   management|married| tertiary|   1476|
        | 59|  blue-collar|married|secondary|      0|
        | 35|   management| single| tertiary|    747|
        | 36|self-employed|married| tertiary|    307|
        | 39|   technician|married|secondary|    147|
        | 41| entrepreneur|married| tertiary|    221|
        | 43|     services|married|  primary|    -88|
        +---+-------------+-------+---------+-------+
        only showing top 10 rows
```

```
In [10]: %%dataframe
         customers
```

Out[10]:

age	job	marital	education	balance
30	unemployed	married	primary	1787
33	services	married	secondary	4789
35	management	single	tertiary	1350
30	management	married	tertiary	1476
59	blue-collar	married	secondary	0
35	management	single	tertiary	747
36	self-employed	married	tertiary	307
39	technician	married	secondary	147
41	entrepreneur	married	tertiary	221
43	services	married	primary	-88



Apache Toree APIs

Apache Toree

Accessing Toree programmatically using Scala

Torre Client APIs

https://github.com/lresende/toree-gateway/blob/v1.0/src/main/scala/org/apache/toree/gateway/ToreeGateway.scala

```
object ToreeGatewayClient extends App {
   // Parse our configuration and create a client connecting
to our kernel
  val configFileContent =
scala.io.Source.fromFile("config.json").mkString
  val config: Config =
ConfigFactory.parseString(configFileContent)

  val client = (new ClientBootstrap(config)
    with StandardSystemInitialization
    with StandardHandlerInitialization).createClient()
…
    val promise = Promise[String]
    try {
      val exRes: DeferredExecution = client.execute("1+1")
      .onResult(executeResult => {
        handleResult(promise, executeResult)
      }).onError(executeReplyError =>{
        handleError(promise, executeReplyError)
      }).onSuccess(executeReplyOk => {
        handleSuccess(promise, executeReplyOk)
      }).onStream(streamResult => {
        handleStream(promise, streamResult)
      })

    } catch {
      case t : Throwable => {
        log.info("Error submitting request: " + t.getMessage,
t)
        promise.success("Error submitting request: " +
t.getMessage)
      }
    }
    Await.result(promise.future, Duration.Inf)
}
```

Apache Toree

Accessing Toree programmatically using Python

Jupyter Client package

https://github.com/lresende/toree-gateway/blob/master/python/toree_client.py

```
self.client = BlockingKernelClient(connection_file=connectionFileLocation)
self.client.load_connection_file(connection_file=connectionFileLocation)
…
msg_id = self.client.execute(code='1+1', allow_stdin=False)

reply = self.client.get_shell_msg(block=True, timeout=timeout)

results = []
while True:
       try:
            msg = self.client.get_iopub_msg(timeout=timeout)
        except:
            raise Exception("Error: Timeout executing request")

       # Stream results are being returned, accumulate them to results
       if msg['msg_type'] == 'stream':
             type = 'stream'
             results.append(msg['content']['text'])
             continue
       elif msg['msg_type'] == 'execute_result':
          if 'text/plain' in msg['content']['data']:
                type = 'text'
                results.append(msg['content']['data']['text/plain'])
                elif 'text/html' in msg['content']['data']:
       # When idle, responses have all been processed/returned
         elif msg['msg_type'] == 'status':
              if msg['content']['execution_state'] == 'idle':
                   break
if reply['content']['status'] == 'ok':
    return ''.join(results)
```

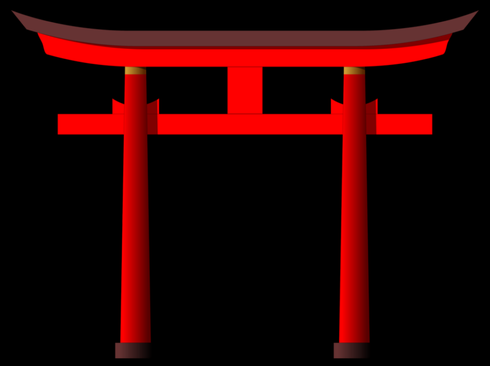
Apache Toree

Accessing Toree programmatically using Jupyter Enterprise Gateway

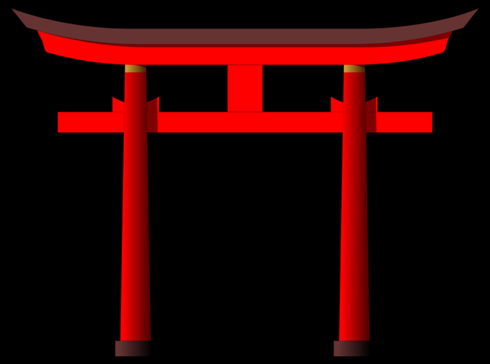
GatewayClient

https://github.com/jupyter-server/enterprise_gateway/tree/master/enterprise_gateway/client

```
 # initialize environment
gatewayClient = GatewayClient()
kernel = gatewayClient.start_kernel('toree-kernelspec')

…

result = self.kernel.execute("1+1")

…

# shutdown environment
gatewayClient.shutdown_kernel(cls.kernel)
```



Demo



Apache Toree: Join the Community

Contribute to Apache Toree

Roadmap suggestions for contributors

- Decouple plain Scala kernel from Spark
- Enhance startup performance
- Evaluate/Implement better async/parallelism framework
- Progress bar for Spark Jobs
- Spark 3.x and Scala 2.13
- Help with documentation and website enhancements

Apache Toree Resources

Apache Toree

https://toree.apache.org

Apache Toree Mailing List

dev@toree.incubator.apache.org

Apache Toree source code at GitHub

https://github.com/apache/incubator-toree

☆ Star and fork the project on Github

Apache Toree
0.5.0 - incubating

pip install -upgrade toree

Questions

lresende@apache.org

@lresende1975

https://www.linkedin.com/in/lresende

https://github.com/lresende