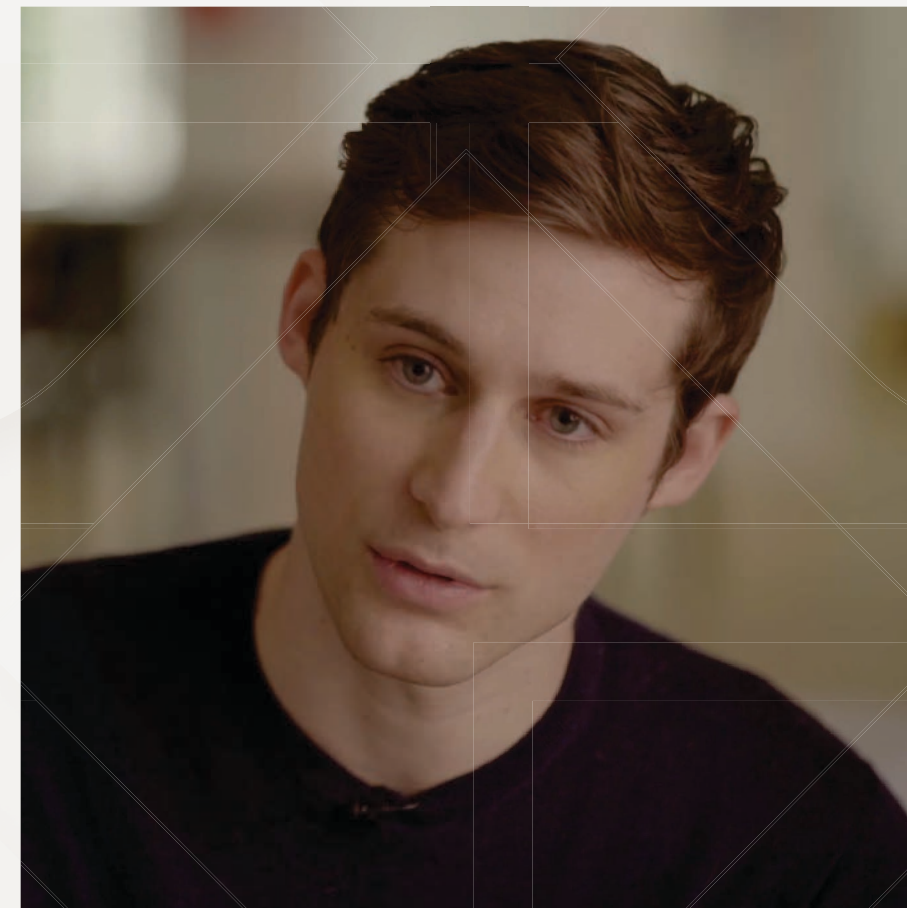


Embracing Digital Sovereignty

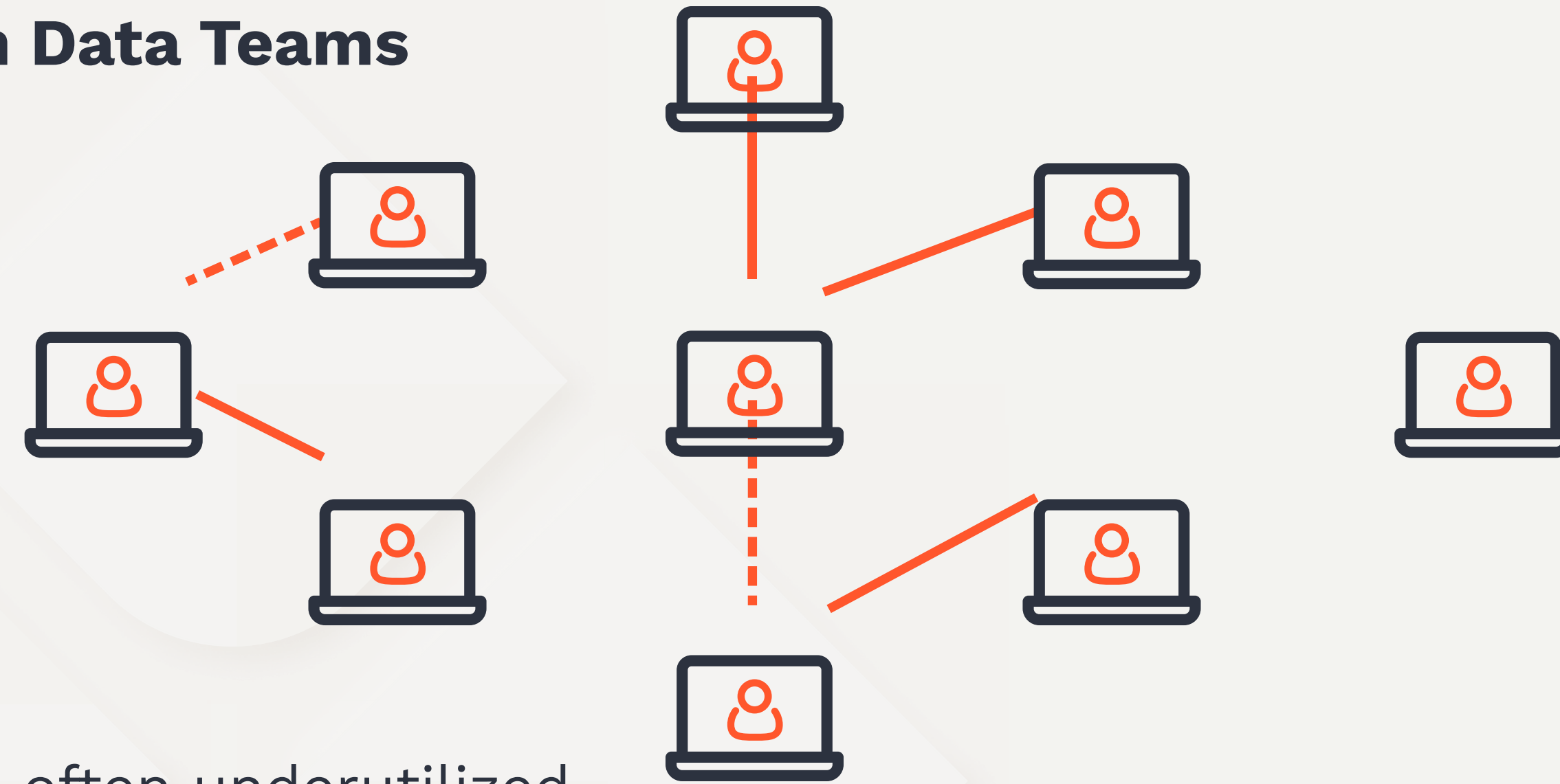
An Open Source Initiative for Autonomous Data Science Environments by the French Public Service



Joseph Garrone

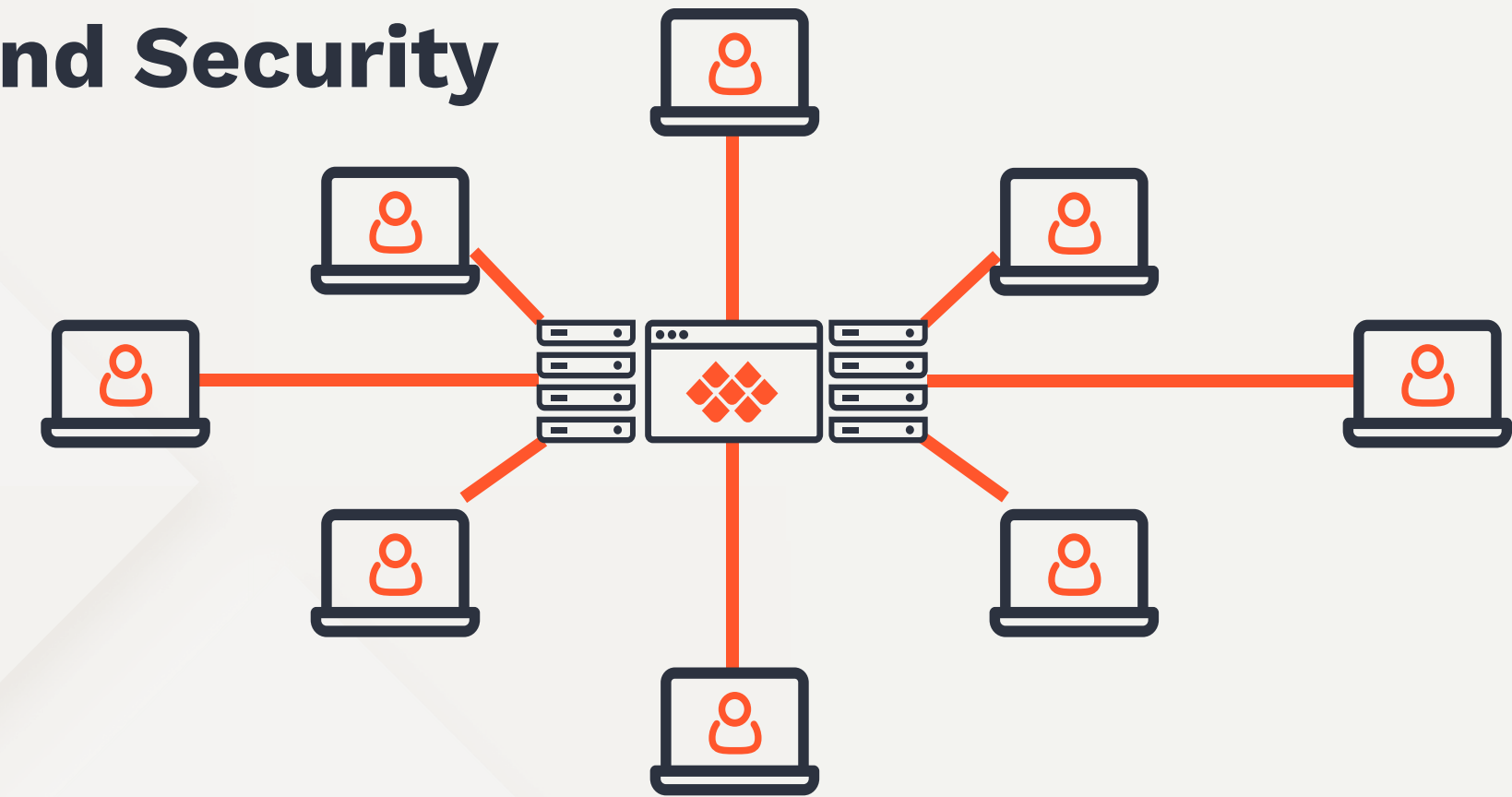
DINUM - Lead Dev of Onyxia

Overcoming the Constraints of Individual Computing in Data Teams



1. **Cost Efficiency:** Individual workstations are expensive and often underutilized.
2. **Computational Capacity:** Physical limitations on RAM/GPU in a single machine.
3. **Scalability:** Traditional computers lack the ability to scale on demand.
4. **Reproducibility:** Challenges in transitioning from experimentation to production due to difficulty in replicating unique software environments for accurate results.
5. **Administrative Restrictions:** Security measures restrict software installations, causing delays and additional administrative tasks.
6. **Data Security:** Downloading sensitive data on internet-exposed machines can spread and risk the data.

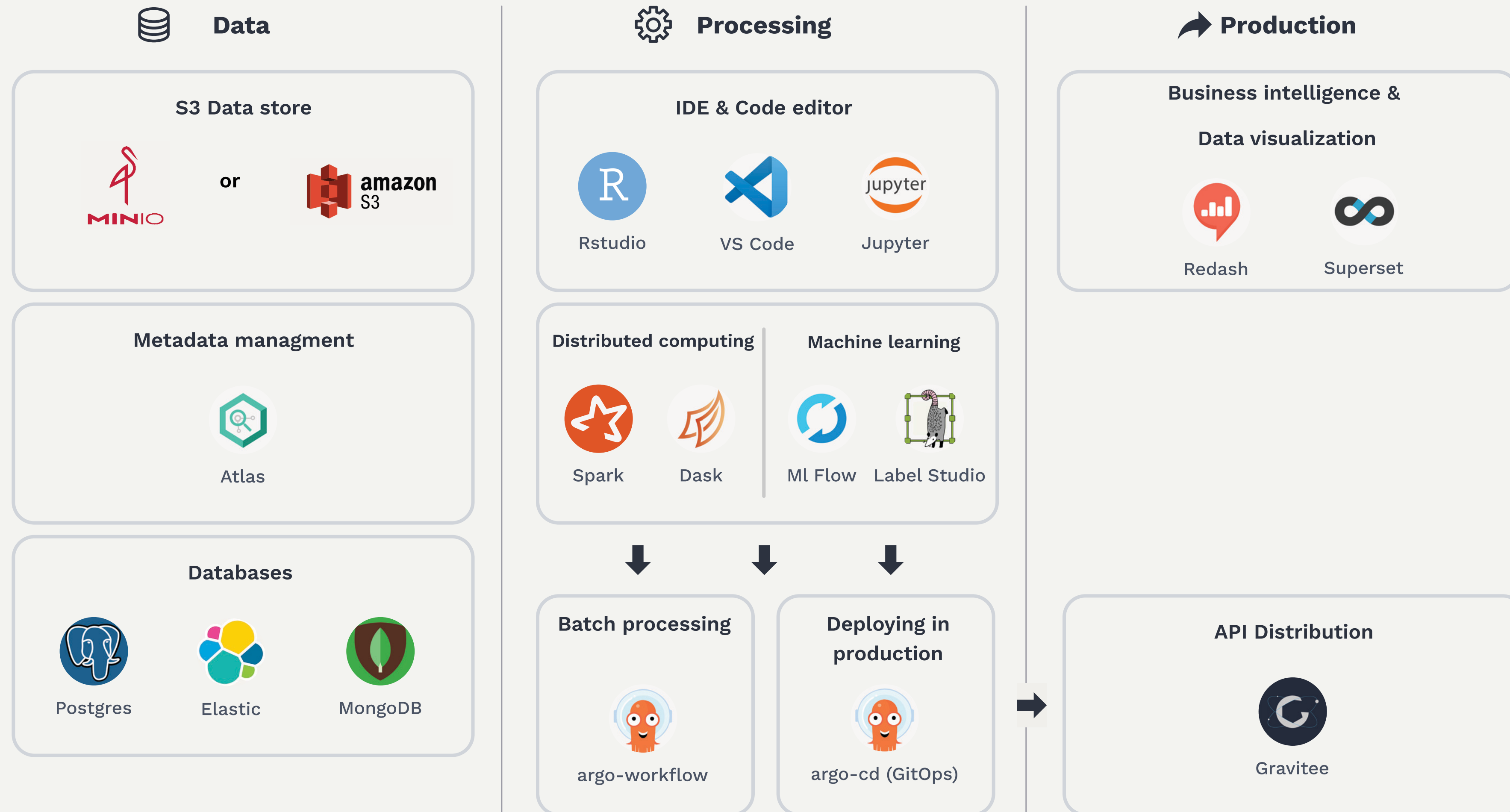
The Power of Cloud-Based Data Platforms: Unmatched Flexibility and Security



1. **Pooling Resources:** Cloud-based data platforms allow data scientists to access and release vast resources on-demand.
2. **Cost and Capacity Efficiency:** Shared resources lead to cost savings and access to levels of power unthinkable with personal workstations.
3. **Horizontal Scalability:** Additional processes can be initiated on-demand, without halting current operations.
4. **Containerization with Kubernetes:** Docker-based environments allow for quick, easy, and replicable setup and teardown of working environments.
5. **Enhanced Data Security:** Sensitive data can be processed in a secure environment, isolated from direct internet exposure.

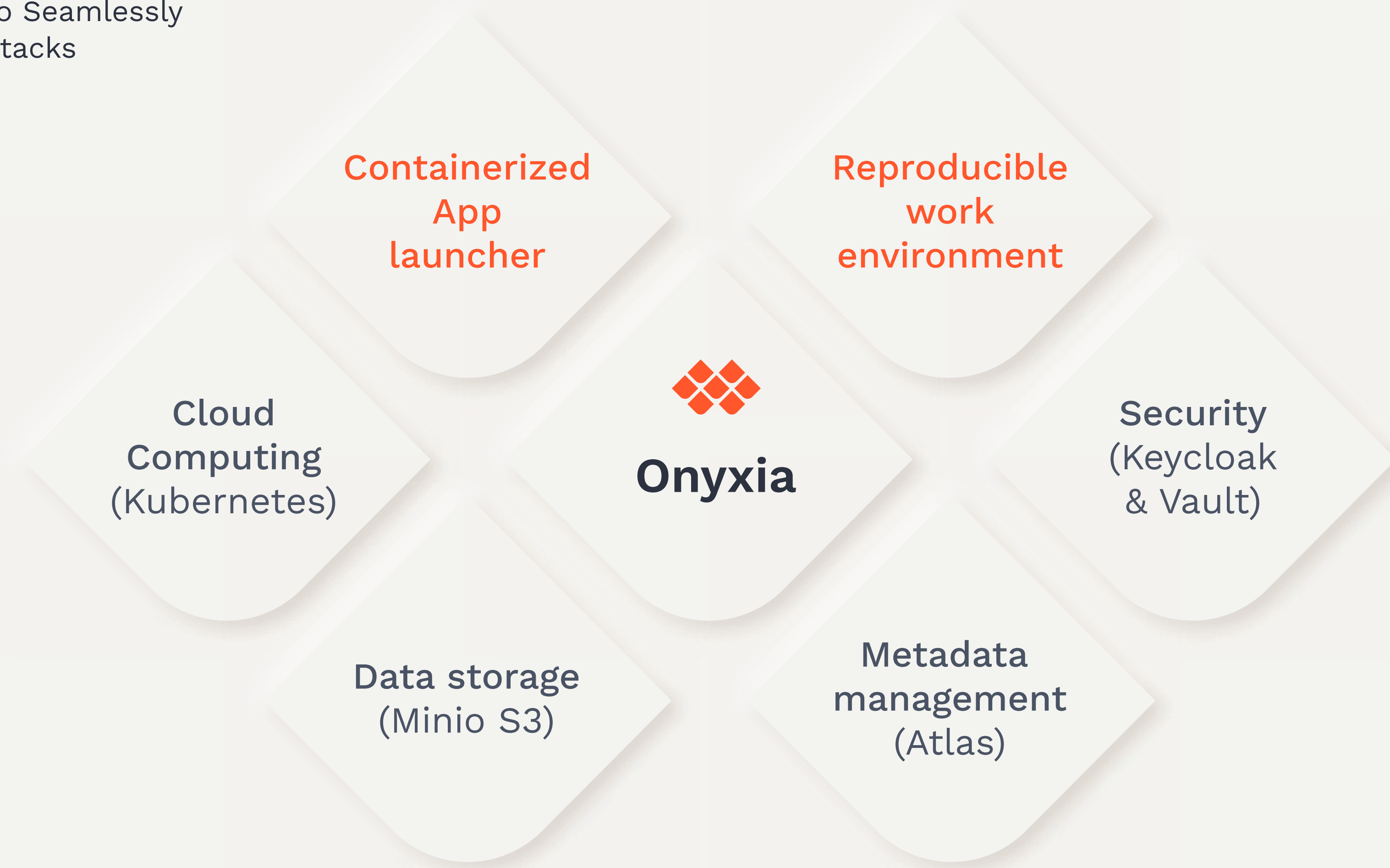
The state of the art in data science technologies

The ideal technology stack for a data scientist



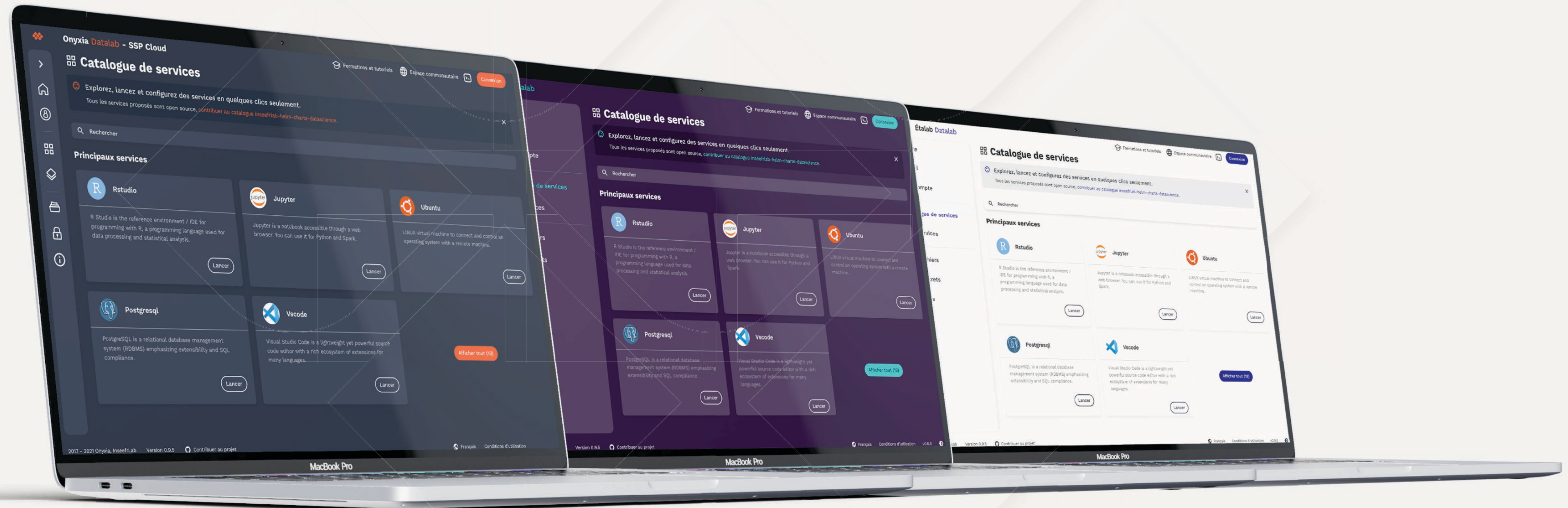
Bridging Technological Complexity: The Role of Onyxia

Democratizing Technology: How Onyxia
Enables Non-IT Experts to Seamlessly
Operate Advanced Tech Stacks



The Versatility of Onyxia: Beyond datalab.sspcloud.fr

Sculpt Your Ideal Data Platform: Deploy a Customized Onyxia Instance On-Premise or Through Cloud Providers



Onyxia - SSP Cloud Datalab

Trainers Documentation Login

Reduce Home My account Services catalog My Services My Secrets My Files

Welcome to the Onyxia datalab

Work with Python or R, enjoy all the computing power you need!

Login

An ergonomic environment and on-demand services
Analyze data, perform distributed computing and take advantage of a large catalog of services. Reserve the computing power you need.
Consult the catalog

An active and enthusiastic community at your service
Use and share the resources available to you: tutorials, training and exchange channels.
Join the community

Fast, flexible and online data storage
To easily access your data and those made available to you from your programs - S3 API implemented.
Consult the catalog

2017 - 2022 Onyxia, InseeFrLab Contribute

Embarquement immédiat

Code.gouv.fr Le SILL Login

Services catalog

Explore, launch and configure services with just a few clicks.
Un repo avec les logiciels du sill qu'il est possible de tester. Contribuez au Logiciels du SILL catalog

SEARCH

Main services

Flarum
Un simple logiciel de forum pour construire des communautés géniales.
Launch

Nocodb
NocoDB is an open source NoCode platform that turns any database into a smart spreadsheet.
Launch

Ubuntu
Ubuntu with a graphical user interface.
Launch

Jupyter
Jupyter est une application web utilisée pour programmer dans plus de 40 langages de programmation, dont Python, Julia, Ruby, R, ou encore Scala.
Launch

Elastic
Elasticsearch est un logiciel utilisant Lucene pour l'indexation et la recherche de données. Il fournit un moteur de recherche distribué et multi-entité à travers une interface REST.
Launch

Show more (8)

2017 - 2022 Onyxia, InseeFrLab Contribute

Onyxia, a graphical user interface for Helm

Helm is a package manager for Kubernetes.



```
apt-get install jupyter-notebook
```

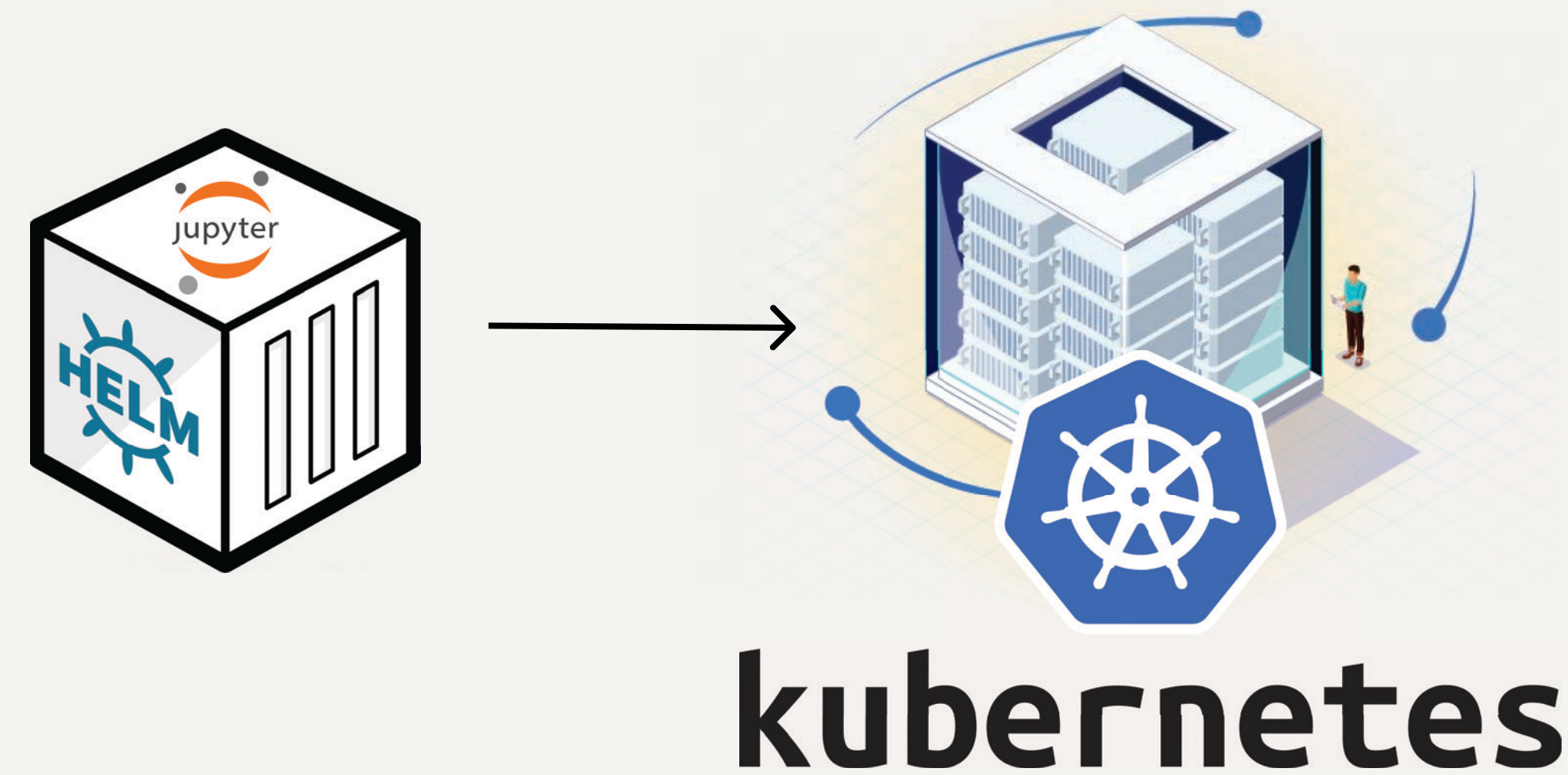
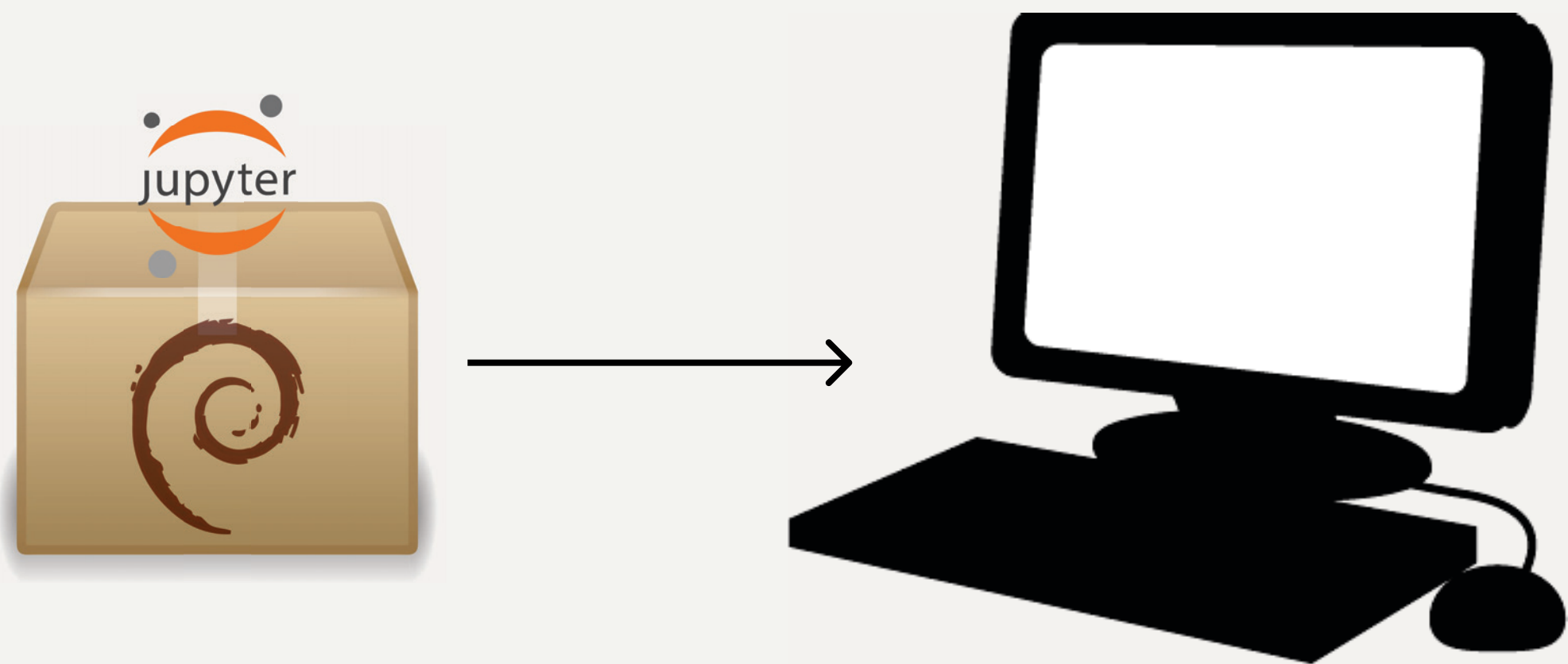


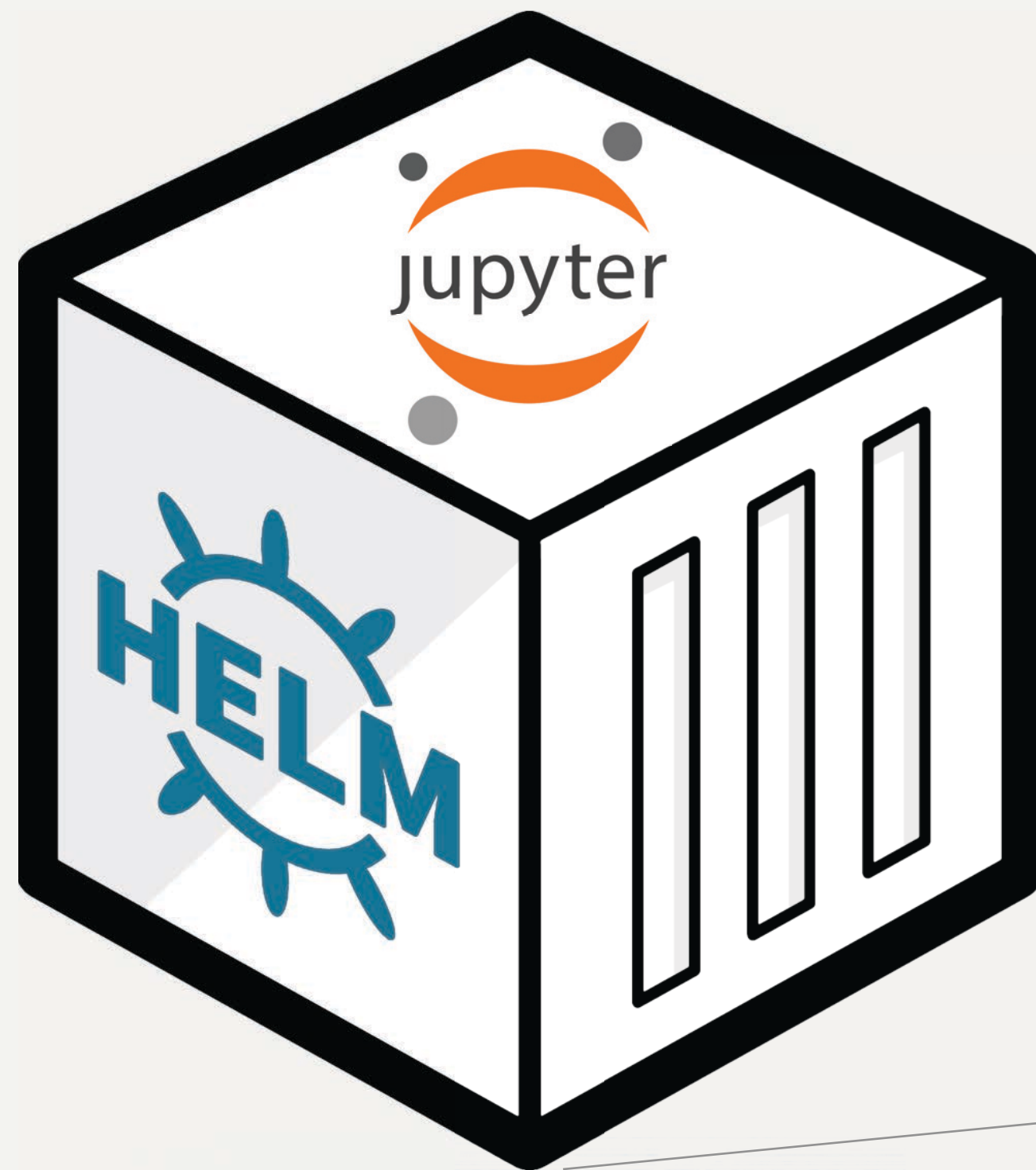
Onyxia, a graphical user interface for Helm

Helm is a package manager for Kubernetes.

```
apt-get install jupyter-notebook
```

```
helm install jupyter-notebook
```





```
{} values.schema.json ●
QW2GON
1  {}
2  "S3": {
3    "accessKeyId": {
4      "type": "string",
5      "x-onyxia": {
6        "overwriteDefaultWith": "s3.accessKeyId"
7      },
8      "description": "Defines the $AWS_ACCESS_KEY_ID ENV variable in your Notebook"
9    },
10   "secretAccessKey": {
11     "type": "string",
12     "render": "password",
13     "x-onyxia": {
14       "overwriteDefaultWith": "s3.secretAccessKey"
15     },
16     "description": "Defines the $AWS_SECRET_ACCESS_KEY ENV variable in your Notebook"
17   }
18 },
19 > "Kubernetes": {...
21 },
22 > "Init": {...
24 },
25 > "Resources": {...
27 },
28 > "Networking": {...
30 },
31 > "Security": {...
33 }
34 }
```

```
{} values.schema.json ●
1 QW2GON
2 "S3": {
3   "accessKeyId": {
4     "type": "string",
5     "x-onyxia": {
6       "overwriteDefaultWith": "s3.accessKeyId"
7     },
8     "description": "Defines the $AWS_ACCESS_KEY_ID ENV variable in your Notebook"
9   },
10  "secretAccessKey": {
11    "type": "string",
12    "render": "password",
13    "x-onyxia": {
14      "overwriteDefaultWith": "s3.secretAccessKey"
15    },
16    "description": "Defines the $AWS_SECRET_ACCESS_KEY ENV variable in your Notebook"
17  }
18 },
19 > "Kubernetes": { ...
21 },
22 > "Init": { ...
24 },
25 > "Resources": { ...
27 },
28 > "Networking": { ...
30 },
31 > "Security": { ...
33 }
34 }
```

Create your personal services

 **Jupyter-python**

Jupyter-python configurations

- S3**
- Kubernetes
- Init
- Resources
- Networking

AccessKeyId
VRPZMJ7QR9JSERWMDH1M
Defines the \$AWS_ACCESS_KEY_ID ENV variable in your Notebook

SecretAccessKey
.....
Defines the \$AWS_SECRET_ACCESS_KEY ENV variable in your Notebook

Cancel Launch

Create your personal services

 **Jupyter-python**

Jupyter-python configurations

- < S3
- Kubernetes**
- Init
- Resources
- Networking >

AccessKeyId
VRPZMJ7QR9JSERWMDH1M
Defines the \$AWS_ACCESS_KEY_ID ENV variable in your Notebook

SecretAccessKey
.....
Defines the \$AWS_SECRET_ACCESS_KEY ENV variable in your Notebook

```
{ } values.schema.json ●
1 QW2GON {
2   "S3": {
3     "accessKeyId": {
4       "type": "string",
5       "x-onyxia": {
6         "overwriteDefaultWith": "s3.accessKeyId"
7       },
8       "description": "Defines the $AWS_ACCESS_KEY_ID ENV variable in your Notebook"
9     },
10    "secretAccessKey": {
11      "type": "string",
12      "render": "password",
13      "x-onyxia": {
14        "overwriteDefaultWith": "s3.secretAccessKey"
15      },
16      "description": "Defines the $AWS_SECRET_ACCESS_KEY ENV variable in your Notebook"
17    }
18  },
19  > "Kubernetes": { ...
20  },
21  },
22  > "Init": { ...
23  },
24  },
25  > "Resources": { ...
26  },
27  },
28  > "Networking": { ...
29  },
30  },
31  > "Security": { ...
32  },
33  }
34 }
```

Cancel Launch

Create your personal services

 **Jupyter-python**

Jupyter-python configurations

< S3

Kubernetes

Init

Resources

Networking >

AccessKeyId

VRPZMJ7QR9JSERWMDH1M

Defines the \$AWS_ACCESS_KEY_ID ENV variable in your Notebook

SecretAccessKey

.....

Defines the \$AWS_SECRET_ACCESS_KEY ENV variable in your Notebook

{} values.schema.json ●

QW2GON

```
1  {}
2  "S3": {
3    "accessKeyId": {
4      "type": "string",
5      "x-onyxia": {
6        "overwriteDefaultWith": "s3.accessKeyId"
7      },
8      "description": "Defines the $AWS_ACCESS_KEY_ID ENV variable in your Notebook"
9    },
10   "secretAccessKey": {
11     "type": "string",
12     "render": "password",
13     "x-onyxia": {
14       "overwriteDefaultWith": "s3.secretAccessKey"
15     },
16     "description": "Defines the $AWS_SECRET_ACCESS_KEY ENV variable in your Notebook"
17   }
18 },
19 > "Kubernetes": { ...
21 },
22 > "Init": { ...
24 },
25 > "Resources": { ...
27 },
28 > "Networking": { ...
30 },
31 > "Security": { ...
33 }
34 }
```

Cancel

Launch



Create your personal services

 **Jupyter-python**

Jupyter-python configurations

- < S3
- Kubernetes
- Init
- Resources**
- Networking
- >

AccessKeyId
VRPZMJ7QR9JSERWMDH1M
Defines the \$AWS_ACCESS_KEY_ID ENV variable in your Notebook

SecretAccessKey
.....
Defines the \$AWS_SECRET_ACCESS_KEY ENV variable in your Notebook

```
{ } values.schema.json ●
1 QW2GON {
2   "S3": {
3     "accessKeyId": {
4       "type": "string",
5       "x-onyxia": {
6         "overwriteDefaultWith": "s3.accessKeyId"
7       },
8       "description": "Defines the $AWS_ACCESS_KEY_ID ENV variable in your Notebook"
9     },
10    "secretAccessKey": {
11      "type": "string",
12      "render": "password",
13      "x-onyxia": {
14        "overwriteDefaultWith": "s3.secretAccessKey"
15      },
16      "description": "Defines the $AWS_SECRET_ACCESS_KEY ENV variable in your Notebook"
17    }
18  },
19  > "Kubernetes": { ...
21  },
22  > "Init": { ...
24  },
25  > "Resources": { ...
27  },
28  > "Networking": { ...
30  },
31  > "Security": { ...
33  }
34 }
```

Cancel Launch

Create your personal services

 **Jupyter-python**

Jupyter-python configurations

- < S3
- Kubernetes
- Init
- Resources
- Networking**
- >

AccessKeyId
VRPZMJ7QR9JSERWMDH1M
Defines the \$AWS_ACCESS_KEY_ID ENV variable in your Notebook

SecretAccessKey
.....
Defines the \$AWS_SECRET_ACCESS_KEY ENV variable in your Notebook

```
{ } values.schema.json ●
1 QW2GON {
2   "S3": {
3     "accessKeyId": {
4       "type": "string",
5       "x-onyxia": {
6         "overwriteDefaultWith": "s3.accessKeyId"
7       },
8       "description": "Defines the $AWS_ACCESS_KEY_ID ENV variable in your Notebook"
9     },
10    "secretAccessKey": {
11      "type": "string",
12      "render": "password",
13      "x-onyxia": {
14        "overwriteDefaultWith": "s3.secretAccessKey"
15      },
16      "description": "Defines the $AWS_SECRET_ACCESS_KEY ENV variable in your Notebook"
17    }
18  },
19  > "Kubernetes": { ...
21  },
22  > "Init": { ...
24  },
25  > "Resources": { ...
27  },
28  > "Networking": { ...
30  },
31  > "Security": { ...
33  }
34 }
```

Cancel Launch

Create your personal services

 **Jupyter-python**

Jupyter-python configurations

- < **S3** Kubernetes Init Resources Networking >

AccessKeyId
VRPZMJ7QR9JSERWMDH1M
Defines the \$AWS_ACCESS_KEY_ID ENV variable in your Notebook

SecretAccessKey
.....
Defines the \$AWS_SECRET_ACCESS_KEY ENV variable in your Notebook

```
{ } values.schema.json ●
1 QW2GON {
2   "S3": {
3     "accessKeyId": {
4       "type": "string",
5       "x-onyxia": {
6         "overwriteDefaultWith": "s3.accessKeyId"
7       },
8       "description": "Defines the $AWS_ACCESS_KEY_ID ENV variable in your Notebook"
9     },
10    "secretAccessKey": {
11      "type": "string",
12      "render": "password",
13      "x-onyxia": {
14        "overwriteDefaultWith": "s3.secretAccessKey"
15      },
16      "description": "Defines the $AWS_SECRET_ACCESS_KEY ENV variable in your Notebook"
17    }
18  },
19  > "Kubernetes": { ...
21  },
22  > "Init": { ...
24  },
25  > "Resources": { ...
27  },
28  > "Networking": { ...
30  },
31  > "Security": { ...
33  }
34 }
```

Cancel Launch

Create your personal services

 **Jupyter-python**

Jupyter-python configurations

- < **S3** Kubernetes Init Resources Networking >

AccessKeyId
VRPZMJ7QR9JSERWMDH1M
Defines the \$AWS_ACCESS_KEY_ID ENV variable in your Notebook

SecretAccessKey
.....
Defines the \$AWS_SECRET_ACCESS_KEY ENV variable in your Notebook

```
{ } values.schema.json ●
1 QW2GON {
2   "S3": {
3     "accessKeyId": {
4       "type": "string",
5       "x-onyxia": {
6         "overwriteDefaultWith": "s3.accessKeyId"
7       },
8       "description": "Defines the $AWS_ACCESS_KEY_ID ENV variable in your Notebook"
9     },
10    "secretAccessKey": {
11      "type": "string",
12      "render": "password",
13      "x-onyxia": {
14        "overwriteDefaultWith": "s3.secretAccessKey"
15      },
16      "description": "Defines the $AWS_SECRET_ACCESS_KEY ENV variable in your Notebook"
17    }
18  },
19  > "Kubernetes": { ...
21  },
22  > "Init": { ...
24  },
25  > "Resources": { ...
27  },
28  > "Networking": { ...
30  },
31  > "Security": { ...
33  }
34 }
```

Cancel Launch

Create your personal services

 **Jupyter-python**

Jupyter-python configurations

- < **S3** Kubernetes Init Resources Networking >

AccessKeyId
VRPZMJ7QR9JSERWMDH1M
Defines the \$AWS_ACCESS_KEY_ID ENV variable in your Notebook

SecretAccessKey
.....
Defines the \$AWS_SECRET_ACCESS_KEY ENV variable in your Notebook

```
{ } values.schema.json ●
1 QW2GON {
2   "s3": {
3     "accessKeyId": {
4       "type": "string",
5       "x-onyxia": {
6         "overwriteDefaultWith": "s3.accessKeyId"
7       },
8       "description": "Defines the $AWS_ACCESS_KEY_ID ENV variable in your Notebook"
9     },
10    "secretAccessKey": {
11      "type": "string",
12      "render": "password",
13      "x-onyxia": {
14        "overwriteDefaultWith": "s3.secretAccessKey"
15      },
16      "description": "Defines the $AWS_SECRET_ACCESS_KEY ENV variable in your Notebook"
17    }
18  },
19  > "Kubernetes": { ...
21  },
22  > "Init": { ...
24  },
25  > "Resources": { ...
27  },
28  > "Networking": { ...
30  },
31  > "Security": { ...
33  }
34 }
```

Cancel Launch

Create your personal services

 **Jupyter-python**

Jupyter-python configurations

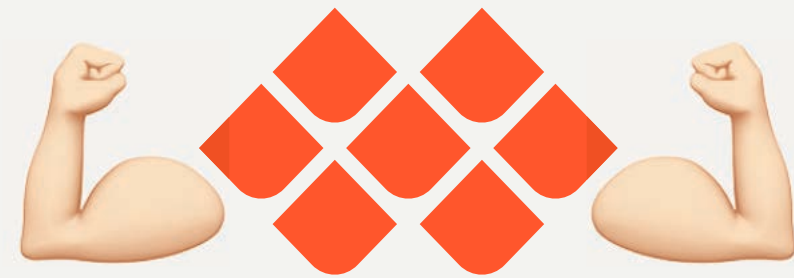
- < **S3** Kubernetes Init Resources Networking >

AccessKeyId
VRPZMJ7QR9JSERWMDH1M
Defines the \$AWS_ACCESS_KEY_ID ENV variable in your Notebook

SecretAccessKey
.....
Defines the \$AWS_SECRET_ACCESS_KEY ENV variable in your Notebook

```
{ } values.schema.json ●
1 QW2GON {
2   "S3": {
3     "accessKeyId": {
4       "type": "string",
5       "x-onyxia": {
6         "overwriteDefaultWith": "s3.accessKeyId"
7       },
8       "description": "Defines the $AWS_ACCESS_KEY_ID ENV variable in your Notebook"
9     },
10    "secretAccessKey": {
11      "type": "string",
12      "render": "password",
13      "x-onyxia": {
14        "overwriteDefaultWith": "s3.secretAccessKey"
15      },
16      "description": "Defines the $AWS_SECRET_ACCESS_KEY ENV variable in your Notebook"
17    }
18  },
19  > "Kubernetes": { ...
21  },
22  > "Init": { ...
24  },
25  > "Resources": { ...
27  },
28  > "Networking": { ...
30  },
31  > "Security": { ...
33  }
34 }
```

Cancel Launch



Create your personal services

 **Jupyter-python**

Jupyter-python configurations

- < **S3** Kubernetes Init Resources Networking >

AccessKeyId
VRPZMJ7QR9JSERWMDH1M
Defines the \$AWS_ACCESS_KEY_ID ENV variable in your Notebook

SecretAccessKey
..... 
Defines the \$AWS_SECRET_ACCESS_KEY ENV variable in your Notebook

```
{} values.schema.json ●
1 QW2GON
2 {
3   "S3": {
4     "accessKeyId": {
5       "type": "string",
6       "x-onyxia": {
7         "overwriteDefaultWith": "s3.accessKeyId"
8       },
9       "description": "Defines the $AWS_ACCESS_KEY_ID ENV variable in your Notebook"
10    },
11    "secretAccessKey": {
12      "type": "string",
13      "render": "password",
14      "x-onyxia": {
15        "overwriteDefaultWith": "s3.secretAccessKey"
16      },
17      "description": "Defines the $AWS_SECRET_ACCESS_KEY ENV variable in your Notebook"
18    }
19  },
20  "Kubernetes": { ...
21  },
22  "Init": { ...
23  },
24  "Resources": { ...
25  },
26  "Networking": { ...
27  },
28  "Security": { ...
29  },
30  }
31 }
32 }
33 }
34 }
```

Cancel Launch

Create your personal services



 **Jupyter-python**

Cancel

Launch

Jupyter-python configurations



- < **S3** Kubernetes Init Resources Networking >

AccessKeyId
VRPZMJ7QR9JSERWMDH1M
Defines the \$AWS_ACCESS_KEY_ID ENV variable in your Notebook

SecretAccessKey
..... 
Defines the \$AWS_SECRET_ACCESS_KEY ENV variable in your Notebook

```
helm install insee/jupyter-python
--set s3.accessKeyId=VRPZMJ7QR9JSERWMDH1M
--set s3.secretAccessKey=zrVMM34sqL0
--set ingress.host=jhon_01.lab.sspcloud.fr
--set ...
```

Create your personal services

 **Jupyter-python**

Cancel **Launch**

Jupyter-python configurations

- < **S3** Kubernetes Init Resources Networking >

AccessKeyId
VRPZMJ7QR9JSERWMDH1M
Defines the \$AWS_ACCESS_KEY_ID ENV variable in your Notebook

SecretAccessKey
..... 
Defines the \$AWS_SECRET_ACCESS_KEY ENV variable in your Notebook

```
helm install insee/jupyter-python
--set s3.accessKeyId=VRPZMJ7QR9JSERWMDH1M
--set s3.secretAccessKey=zrVMM34sqL0
--set ingress.host=jhon_01.lab.sspcloud.fr
--set ...
```

Create your personal services

 **Jupyter-python**

Cancel **Launch**

Jupyter-python configurations

- < **S3** Kubernetes Init Resources Networking >

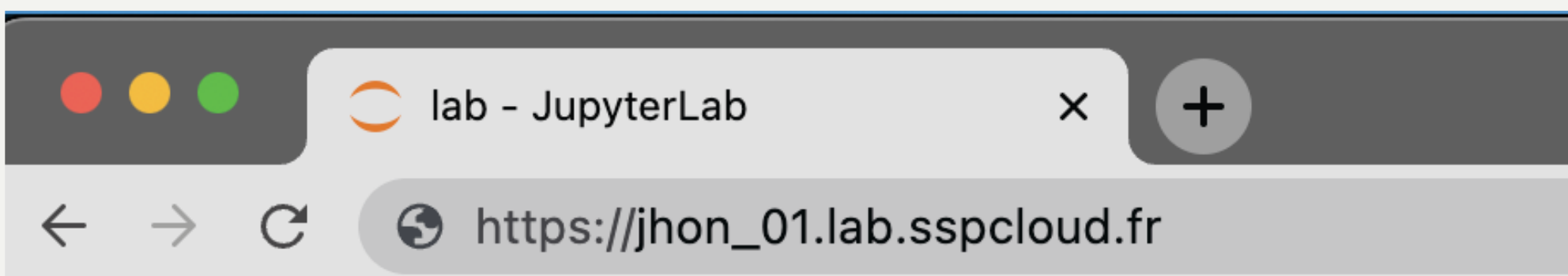
AccessKeyId
VRPZMJ7QR9JSERWMDH1M
Defines the \$AWS_ACCESS_KEY_ID ENV variable in your Notebook

SecretAccessKey
..... 
Defines the \$AWS_SECRET_ACCESS_KEY ENV variable in your Notebook


```
helm install insee/jupyter-python
--set s3.accessKeyId=VRPZMJ7QR9JSERWMDH1M
--set s3.secretAccessKey=ZrVMM34sqL0
--set ingress.host=jhon_01.lab.sspcloud.fr
--set ...
```

The screenshot shows a JupyterLab interface. At the top, a browser window displays the URL `https://jhon_01.lab.sspcloud.fr`. Below the browser is the JupyterLab menu bar with options: File, Edit, View, Run, Kernel, Git, Tabs, Settings, Help. A terminal window is open, showing the command `echo $AWS_ACCESS_KEY_ID` and its output `VRPZMJ7QR9JSERWMDH1M`.

```
helm install insee/jupyter-python
--set s3.accessKeyId=VRPZMJ7QR9JSERWMDH1M
--set s3.secretAccessKey=zrVMM34sqL0
--set ingress.host=jhon_01.lab.sspcloud.fr
--set ...
```



File Edit View Run Kernel Git Tabs Settings Help

```
onyxia@jupyter-python-407512-0:~$ echo $AWS_ACCESS_KEY_ID
VRPZMJ7QR9JSERWMDH1M
onyxia@jupyter-python-407512-0:~$
```


<https://onyxia.sh>

