

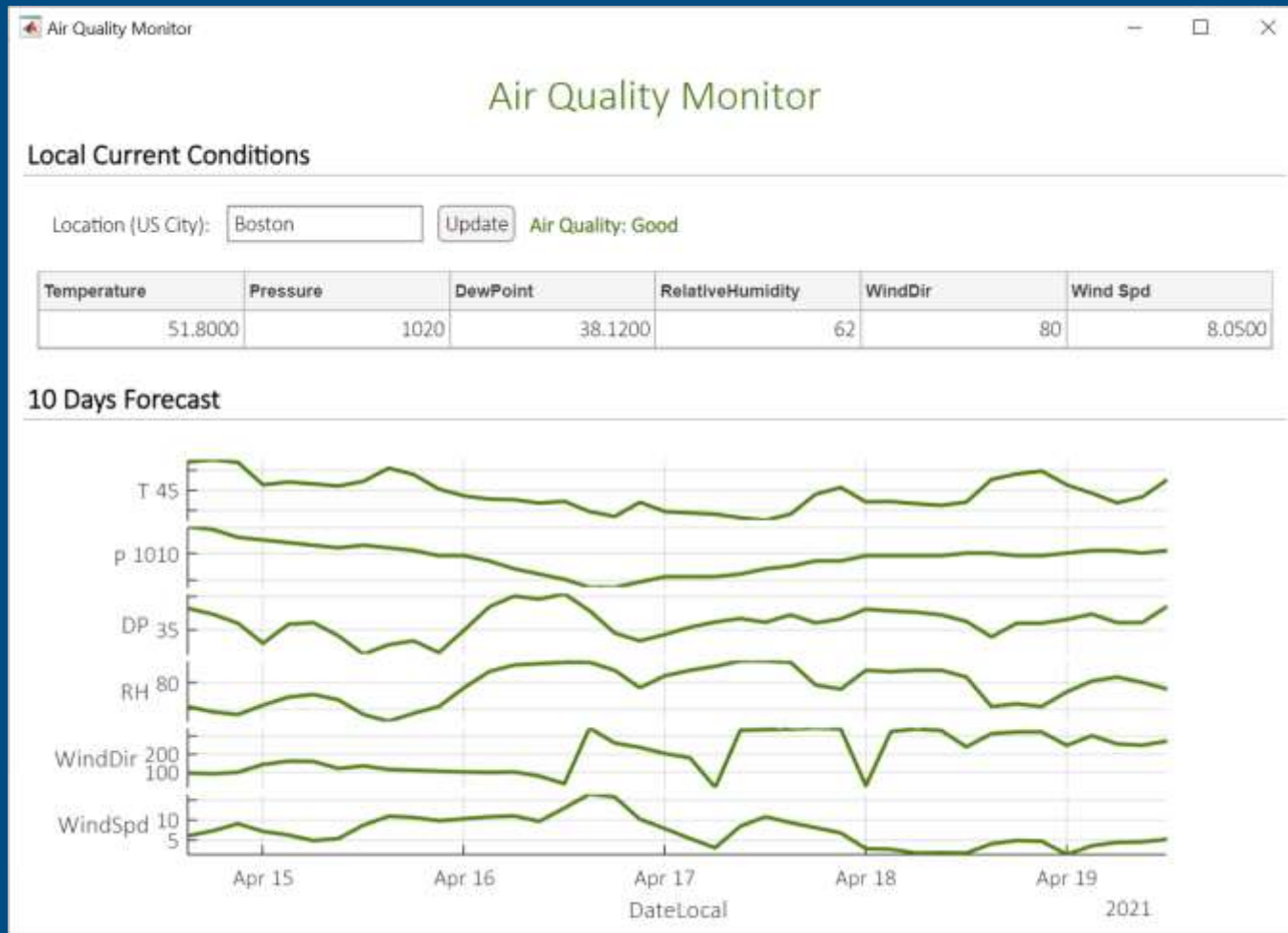
MATLAB EXPO 2021

MATLAB With Python *for engineers and data scientists*

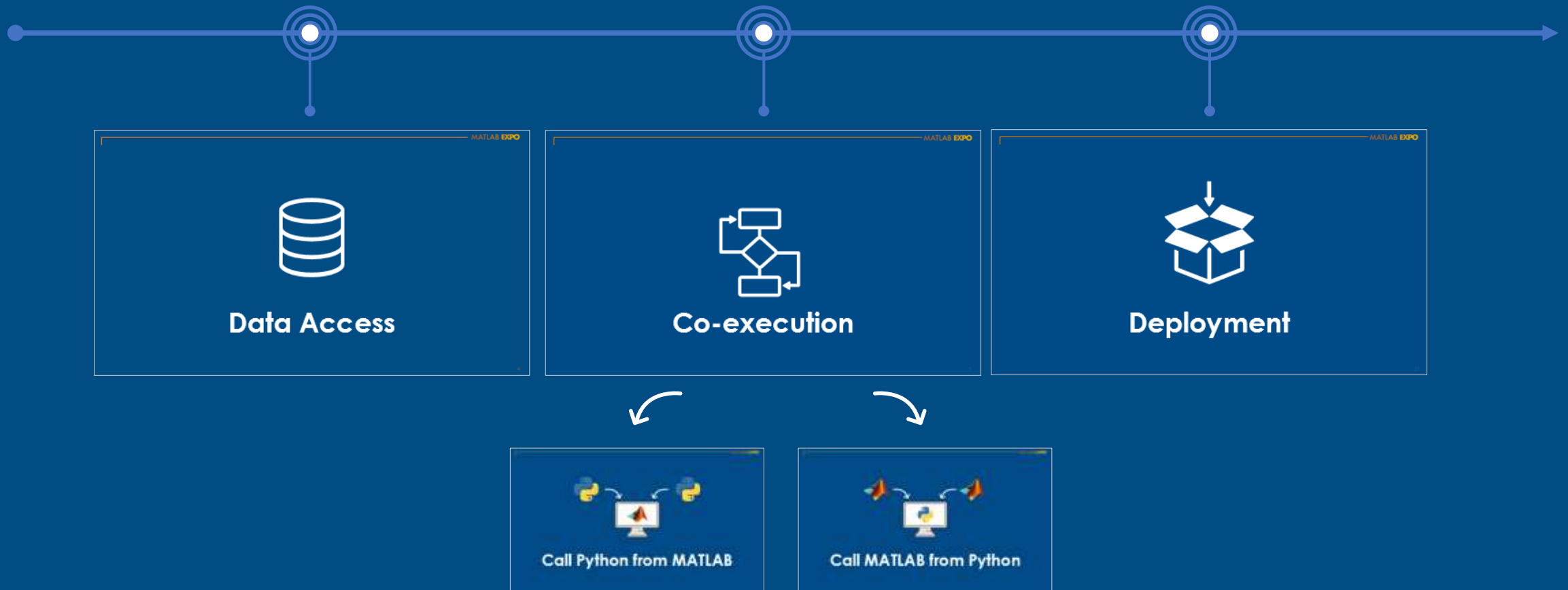
성 호 현 The MathWorks Korea



Example: Build Air Quality App using MATLAB and Python



Development plan





Data Access

Access Data from a Web Service

<https://openweathermap.org/>

Data Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment

OpenWeather global services

Weather forecasts, nowcasts and history in fast and elegant way

2 Billion Forecasts Per Day
2,500 new subscribers a day

2,600,000 customers
20+ weather APIs

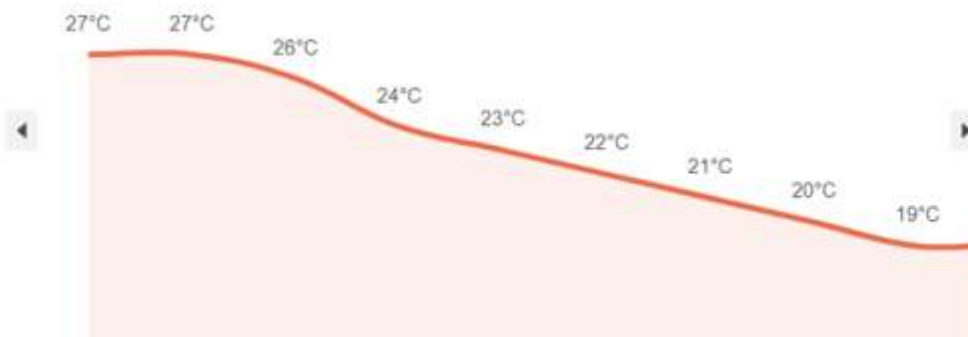


Hourly forecast

Temperature

Precipitation

now 6pm 7pm 8pm 9pm 10pm 11pm Sep 10 12am 1am



clear sky few clouds scattered clouds scattered clouds overcast clouds overcast clouds overcast clouds broken clouds broken clouds
3.3m/s 3.5m/s 3.8m/s 3.4m/s 2.8m/s 3.0m/s 3.5m/s 3.2m/s 2.4m/s

8-day forecast

Wed, Sep 09	27 / 19°C	clear sky
Thu, Sep 10	25 / 18°C	broken clouds
Fri, Sep 11	28 / 17°C	scattered clouds
Sat, Sep 12	25 / 16°C	clear sky
Sun, Sep 13	26 / 15°C	broken clouds
Mon, Sep 14	32 / 17°C	scattered clouds
Tue, Sep 15	27 / 20°C	moderate rain
Wed, Sep 16	21 / 17°C	light rain

What type of data?

Numerical, Textual, Geolocalized, Timeseries, ...

Data Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment



**current
weather**



**hourly
forecast**



**daily
forecast**



**climatic
forecast**



**historical
weather**

Called by:

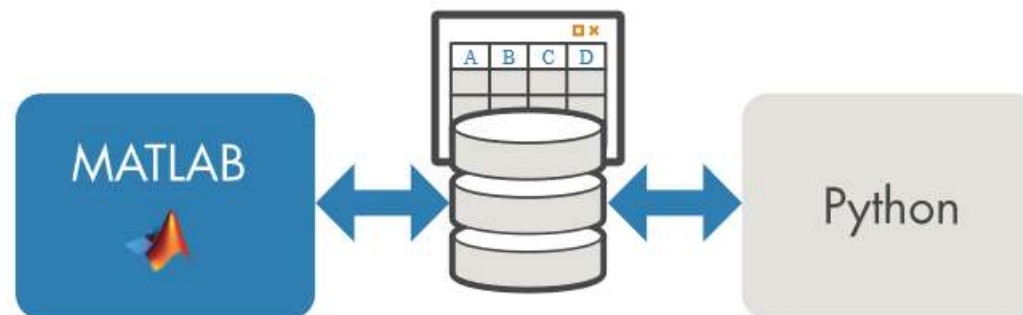
geographical coordinates, zip codes, city name, city ID, number of cities (only in current and forecasted APIs)

<https://openweathermap.org/>

Store & transfer tabular data between languages

Parquet files

- Columnar storage format available to any project in the Hadoop big data ecosystem, regardless of the choice of data processing framework, data model or programming language (More on [Parquet](#))



```
parquetwrite("temperatureFitting.parquet",T)
```

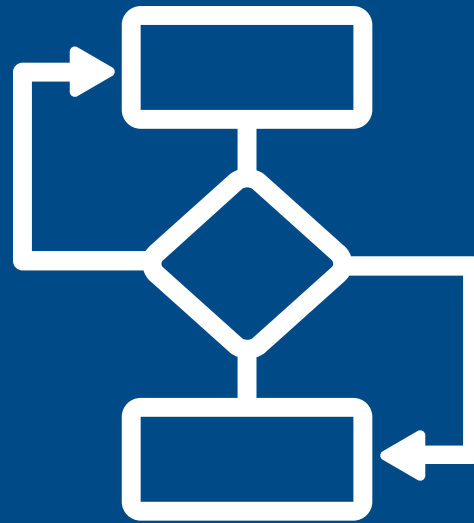
T = 40x3 table

	Time	Temperatures	SineFit
1	2020-07-16...	19.0200	23.8001
2	2020-07-16...	20.3300	24.6581
3	2020-07-16...	20.1300	23.0750
4	2020-07-16...	18.9300	19.9678
5	2020-07-17...	17.1500	17.1366
6	2020-07-17...	16.0200	16.2214
7	2020-07-17...	16.7900	17.7525
8	2020-07-17...	21.1200	23.2100

```
# Use parquet file as alternative to exchange tables with MATLAB
df = pd.read_parquet("temperatureFitting.parquet")
df.head()
```

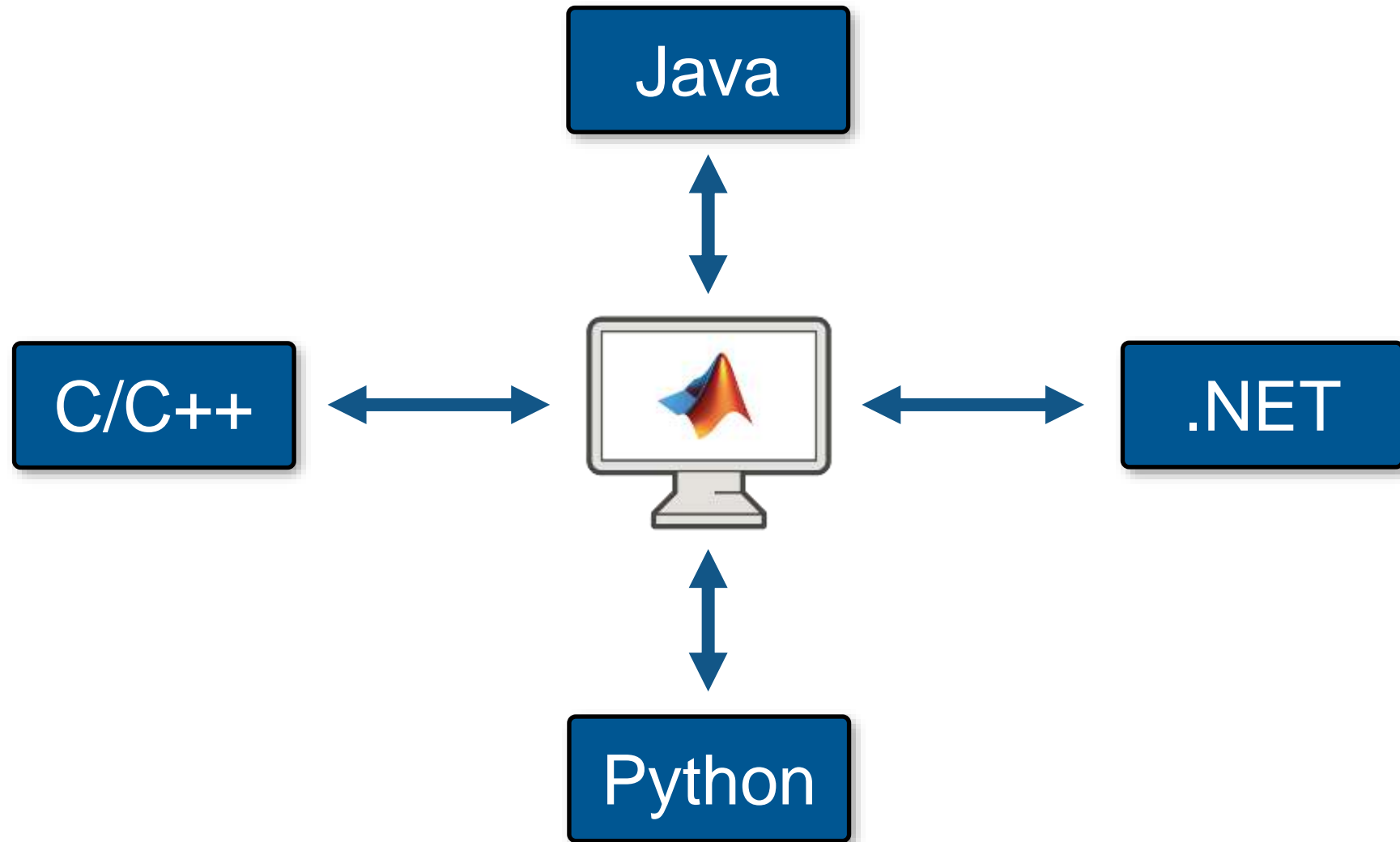
	Time	Temperatures	SineFit
0	2020-07-16 12:00:00	19.02	23.800150
1	2020-07-16 15:00:00	20.33	24.658138
2	2020-07-16 18:00:00	20.13	23.074984
3	2020-07-16 21:00:00	18.93	19.967804
4	2020-07-17 00:00:00	17.15	17.136577

- Call Python from MATLAB
- Call MATLAB from Python



Co-execution

MATLAB provides flexible integration with multiple languages



Data
Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment

Given: Existing Python Code accessing & preparing weather data

Data preparation

Modeling

Deployment

Weather Data

Data
Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment

Call Python from MATLAB

Data preparation

Modeling

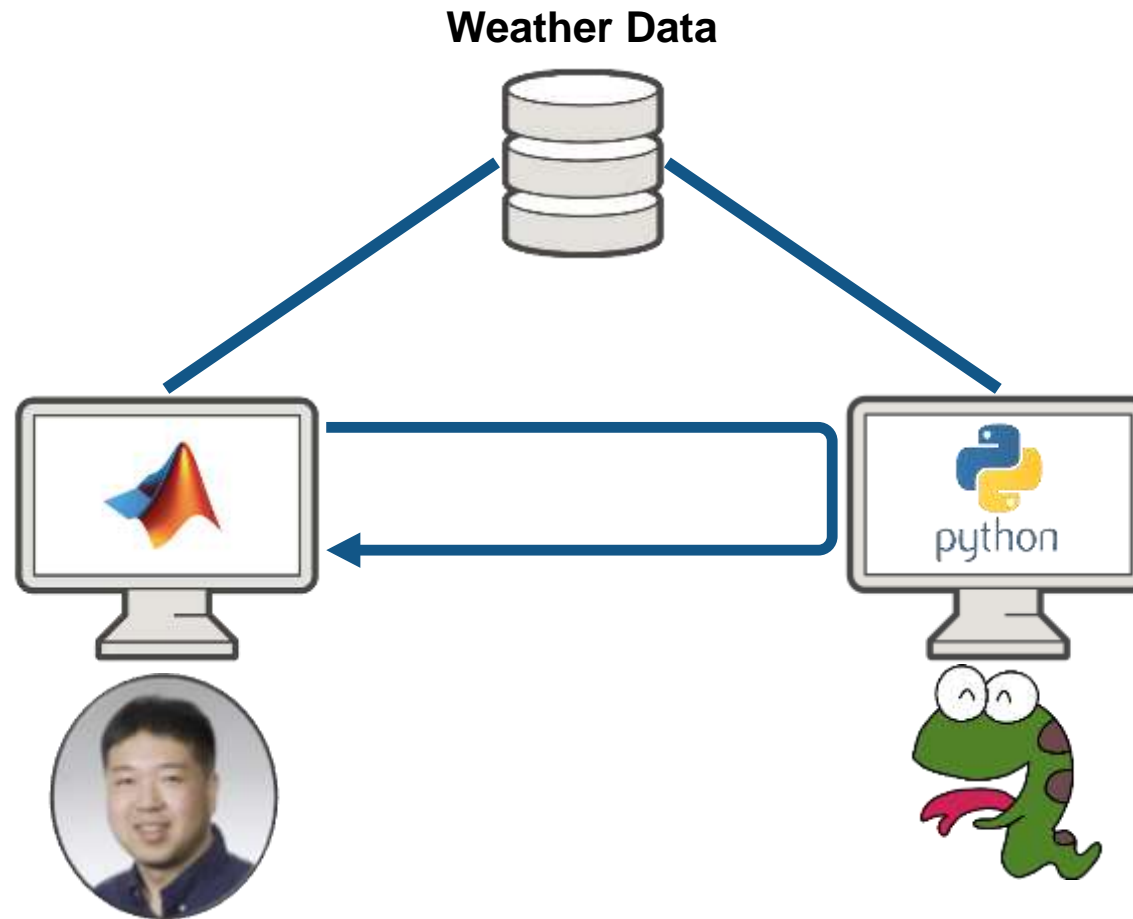
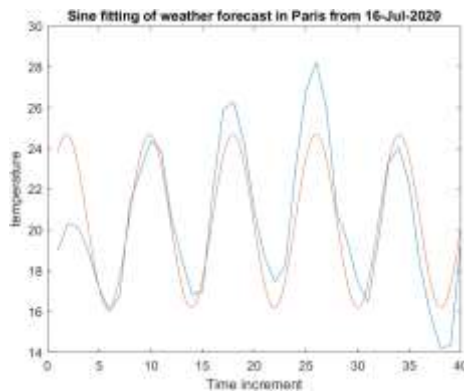
Deployment

Data
Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment



Call MATLAB from Python

Data preparation

Modeling

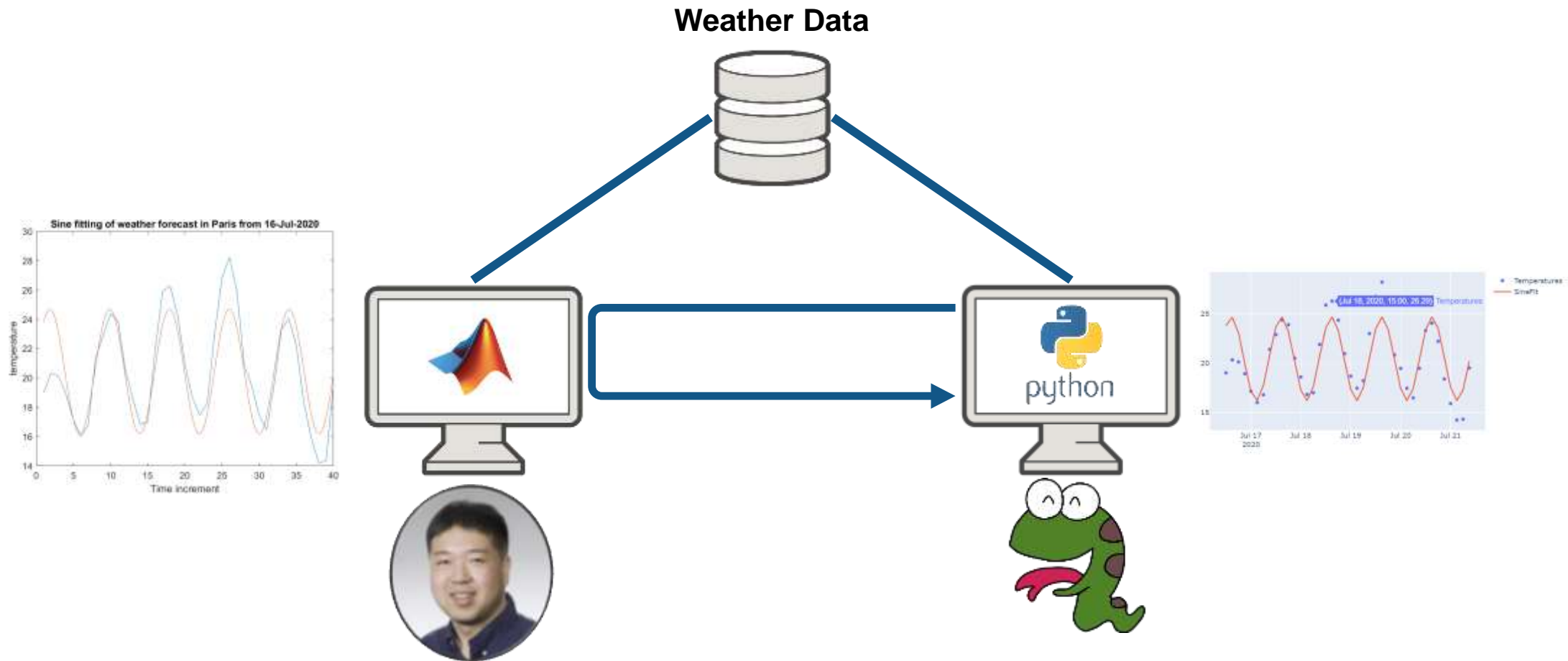
Deployment

Data
Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment



Deploy: MATLAB Analytics into Python

Data preparation

Modeling

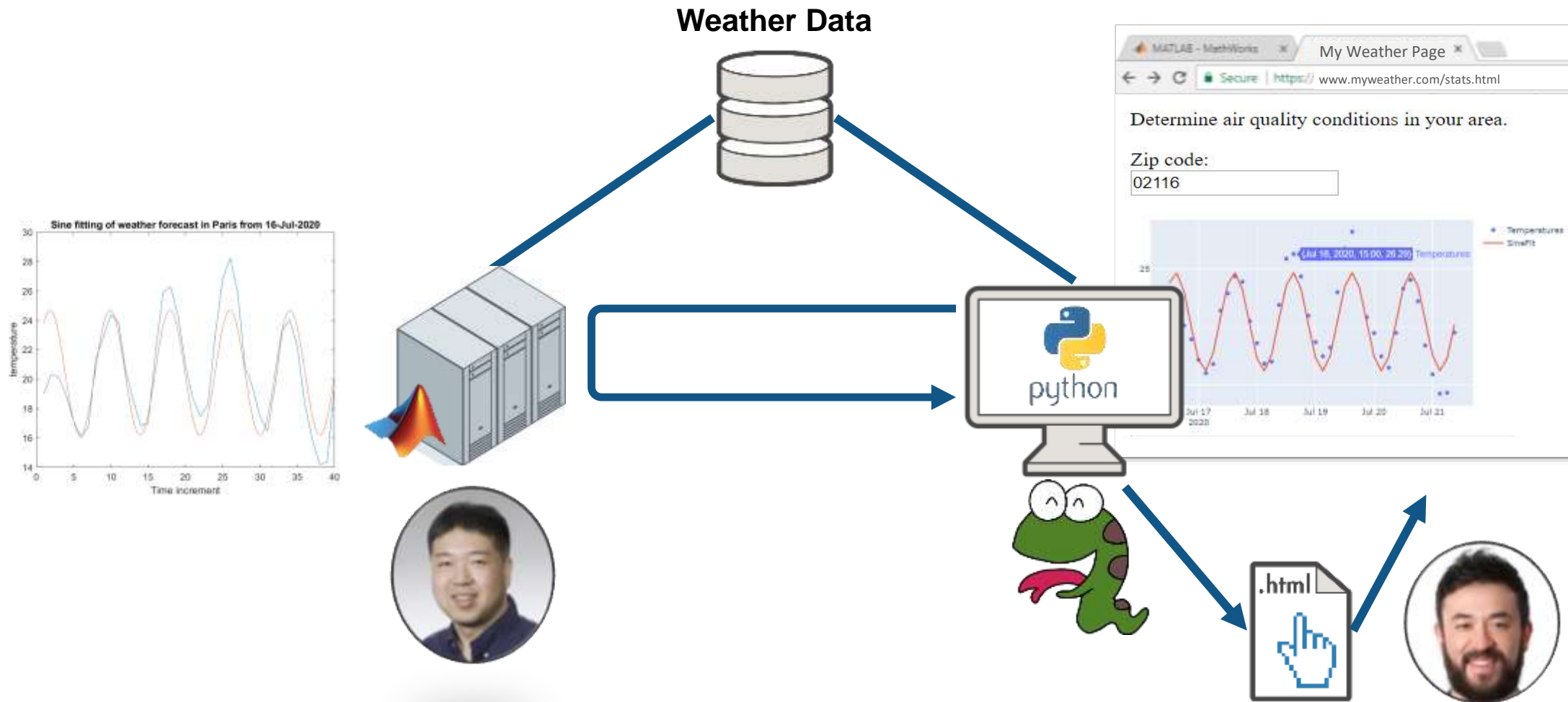
Deployment

Data
Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment



Deploy: MATLAB Analytics into Python

Data preparation

Modeling

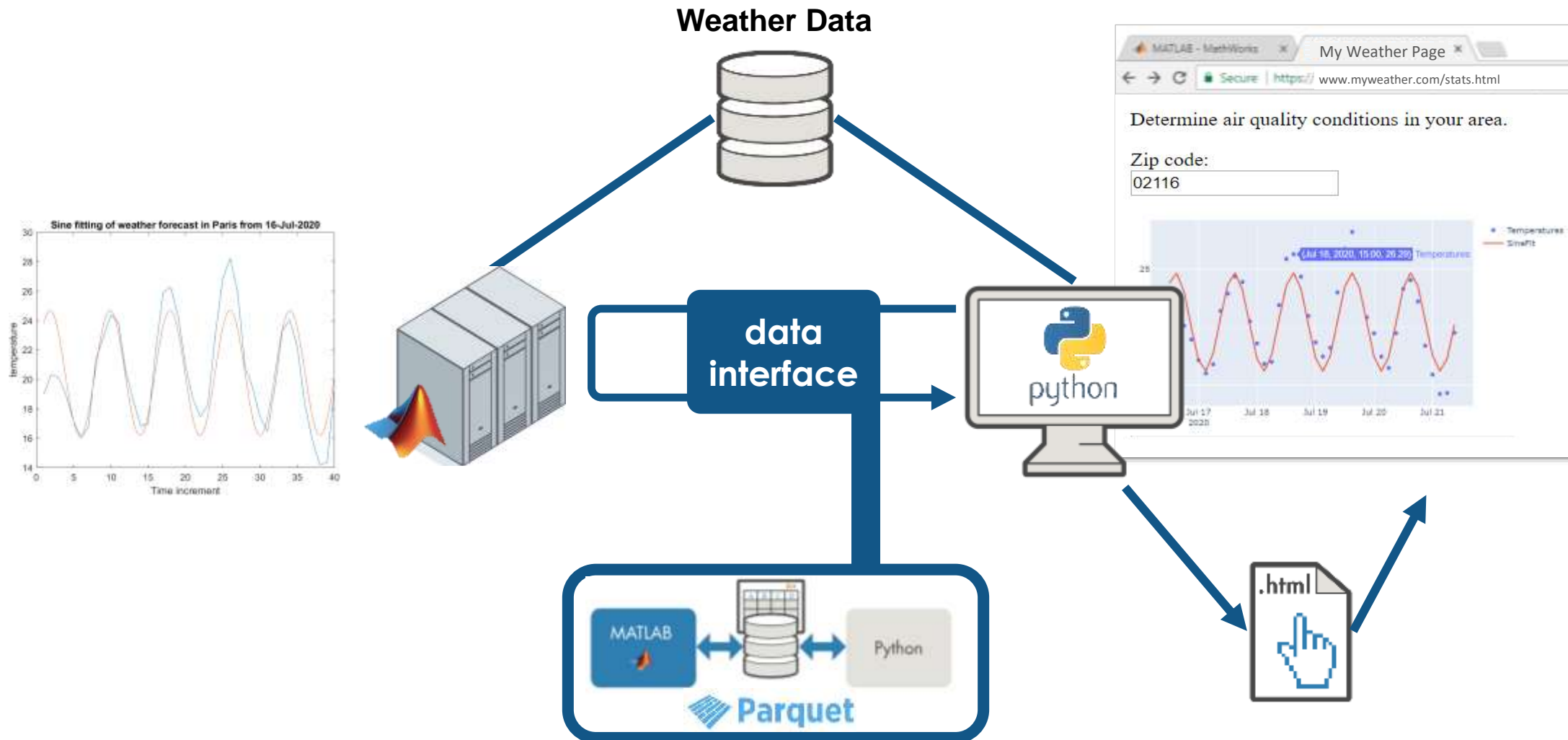
Deployment

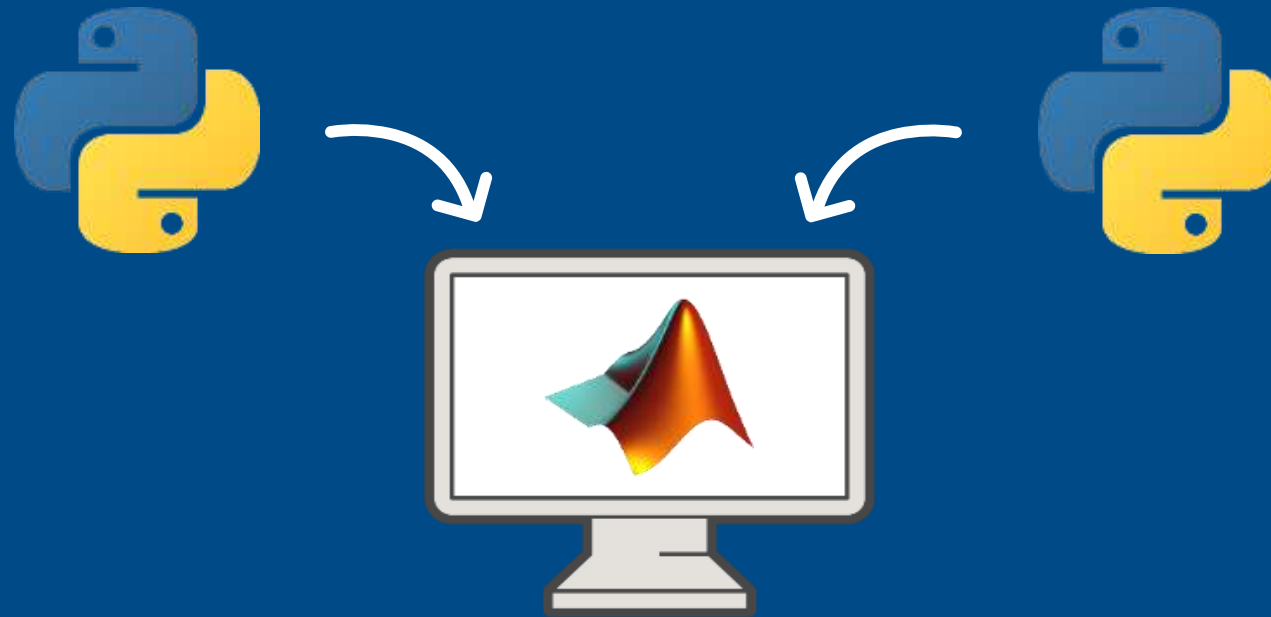
Data
Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment



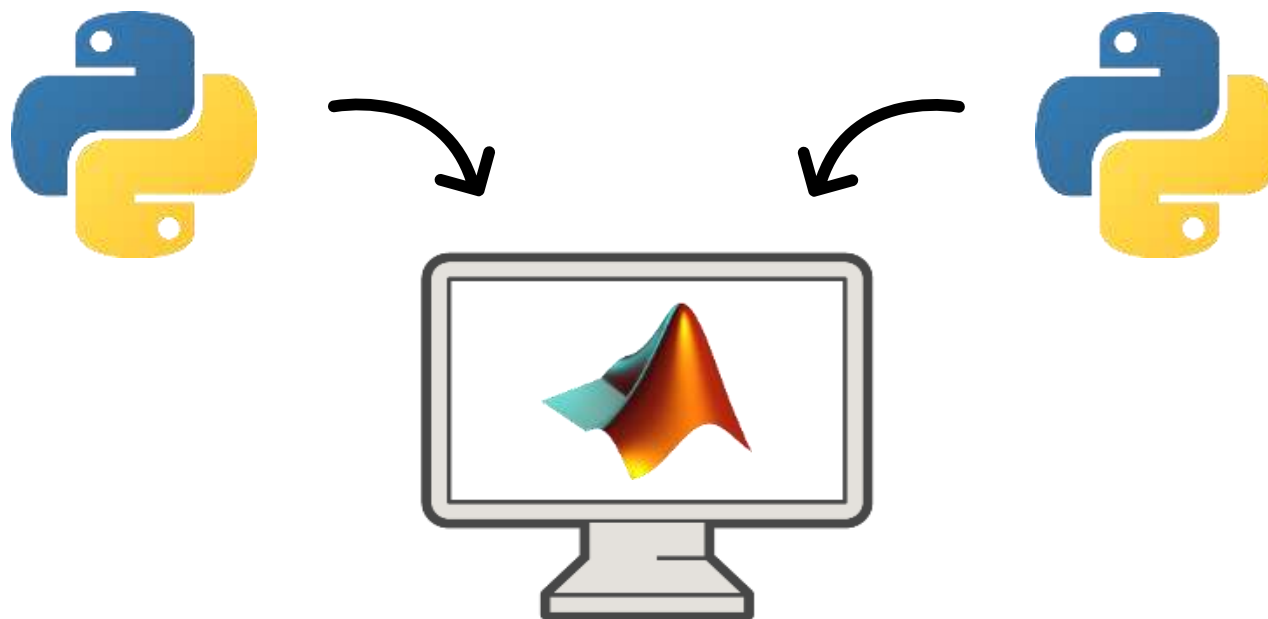


Call Python from MATLAB

Why Call Python from MATLAB?

Already working in MATLAB, and:

- Want to reuse existing Python code
- Need functionality that is only available in Python



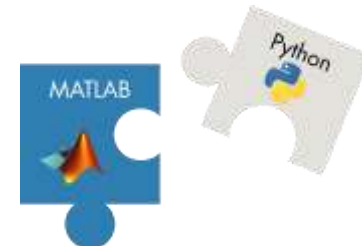
Data
Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment

Calling Python libraries from MATLAB



Use the weather.py module to get the air quality for Paris. This is a user-defined Python module which includes functions to read and parse the current and forecasted weather data by location.

```
jsonData = py.weather.get_current_weather("Paris", "France", apikey.Key)
```

jsonData =
Python **dict** with no properties.

```
{'coord': {'lon': 2.35, 'lat': 48.85}, 'weather': [{'id': 803, 'main': 'Cloudy'}
```

Parse the json data returned from the weather API.

The Python dictionary can be represented as a MATLAB struct.

```
weatherData = py.weather.parse_json(jsonData);  
struct(weatherData)
```

```
ans = struct with fields:  
    temp: 18.7100  
 feels_like: 17.3000  
    temp_min: 17.7800  
    temp_max: [1x1 py.int]
```

Use a function (prepData.m) to prepare data for machine learning (create a table with the expected variable names, preprocessing steps, etc).

```
currentData = prepData(weatherData)
```

currentData = 1x12 table

	DateLocal	city	StateName	T	P	DP	RH	WindDir	WindSpd		
1	01-Jul-2020 11:...	"Paris"	Ile de France	21.6200	20.2600	349.2200	1010	5.1000	73		

```
def get_current_weather(city, country, apikey):  
    # get current conditions in specified location  
    # get_current_weather('boston', 'us', key)  
    import urllib.request  
    import json  
    # read current conditions  
    try:  
        url = "https://api.openweathermap.org/data/2.5/weather?q="+city+"&"+country+"&appid="+apikey  
        response = urllib.request.urlopen(url)  
        html = response.read()  
        json_data = json.loads(html)  
  
    except urllib.error.URLError:  
        # if weather API doesn't work, read the file  
        json_data = read_backup(city)  
  
    return json_data
```

Data
Access

Co-Execution

- Call Python
from MATLAB

- Call MATLAB
from Python

Deployment

Calling Python libraries from MATLAB



Data
Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment

A screenshot of the Visual Studio Code editor interface. The title bar shows 'weather.py - Visual Studio ...'. The editor is open to a file named 'weather.py' located at 'C:\>_work > MATLABwithPython > weatherPrediction > 1_CallPythonFromMATLAB > weather.py'. The code in the editor is as follows:

```
9  
10  
11 # weather.py  
12 import csv  
13 import datetime  
14 import json  
15 import urllib.request  
16  
17 BASE_URL = 'https://api.openweathermap.org/data/2.5/{ }?q={ },{ }&uni  
18 FORECAST_KEYS = {'current_time': 'DateLocal', 'temp': 'T', 'deg': 'Wi  
19 | | | | 'speed': 'WindSpd', 'humidity': 'RH', 'pressure': 'P  
20  
21 def read_backup(city):  
22     '''Read example data from a backup file'''  
23  
24     with open('backupdata.csv', newline='') as csvfile:  
25         reader = csv.DictReader(csvfile)  
26         for s in [*reader]:
```

The status bar at the bottom shows 'Ln 11, Col 13 (10 selected) Spaces: 4 UTF-8 LF Python'.

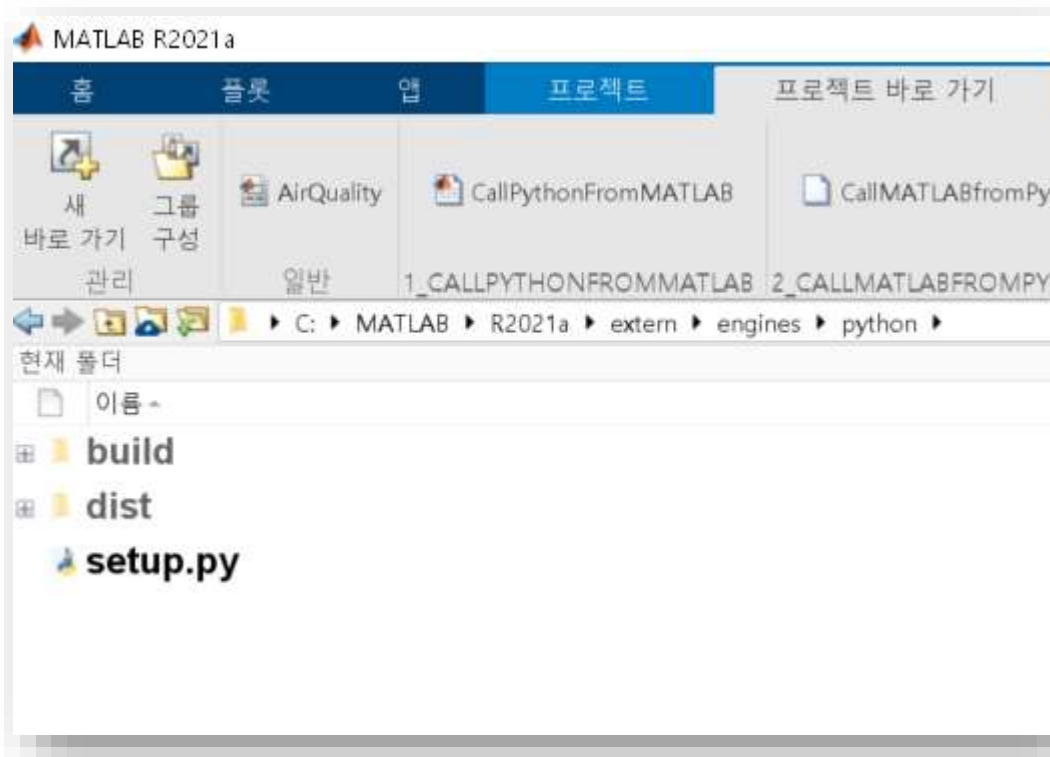
Connect to Python

Data Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment



명령 창

MATLAB을 처음 사용한다면 [시작하기](#)를 참조하십시오.

```
>> pyenv
```

```
ans =
```

PythonEnvironment - 속성 있음:

Version: "3.7"

Executable: "C:\Python\python.EXE"

Library: "C:\Python\python37.dll"

Home: "C:\Python"

Status: Loaded

ExecutionMode: InProcess

ProcessID: "50412"

ProcessName: "MATLAB"

fx >> |

The screenshot displays the MATLAB R2020b interface. The top menu bar includes HOME, PLOTS, APPS, LIVE EDITOR (active), INSERT, and VIEW. The LIVE EDITOR tab is selected, showing a toolbar with icons for Code, Control, Task, Section Break, Text, Table of Contents, Code Example, Image, Hyperlink, and Equation. Below the toolbar, the current folder is C:_work\MATLABwithPython\weatherPrediction\1_CallPythonFromMATLAB. The workspace shows files: __pycache__, accessKey.txt, airQualModel.mat, CallPythonFromMATLAB..., predictAirQual.m, prepData.m, and weather.py. The Live Editor window displays the following code:

```
Executable: "C:\Users\fperino\AppData\Local\Programs\Python\Python36\python.exe"  
Library: "C:\Users\fperino\AppData\Local\Programs\Python\Python36\python36.dll"  
Home: "C:\Users\fperino\AppData\Local\Programs\Python\Python36"  
Status: Loaded  
ExecutionMode: InProcess  
ProcessID: "256"  
ProcessName: "MATLAB"
```

Below the code, a test function is shown:

```
2 py.math.sqrt(12)  
ans = 3.4641
```

The Command Window at the bottom shows the output of the test function.

Data
Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment

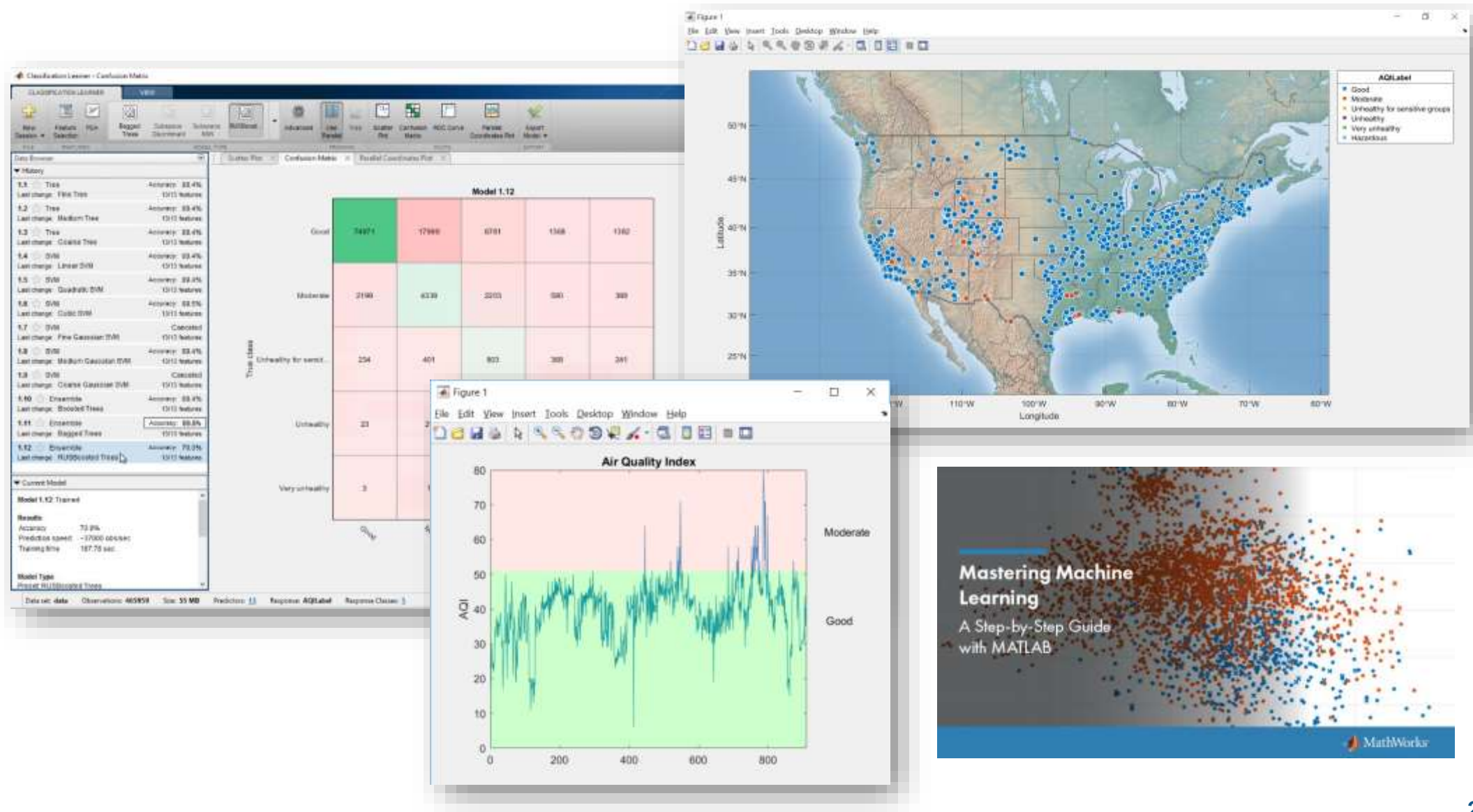
Air Quality Prediction Model

Data
Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment



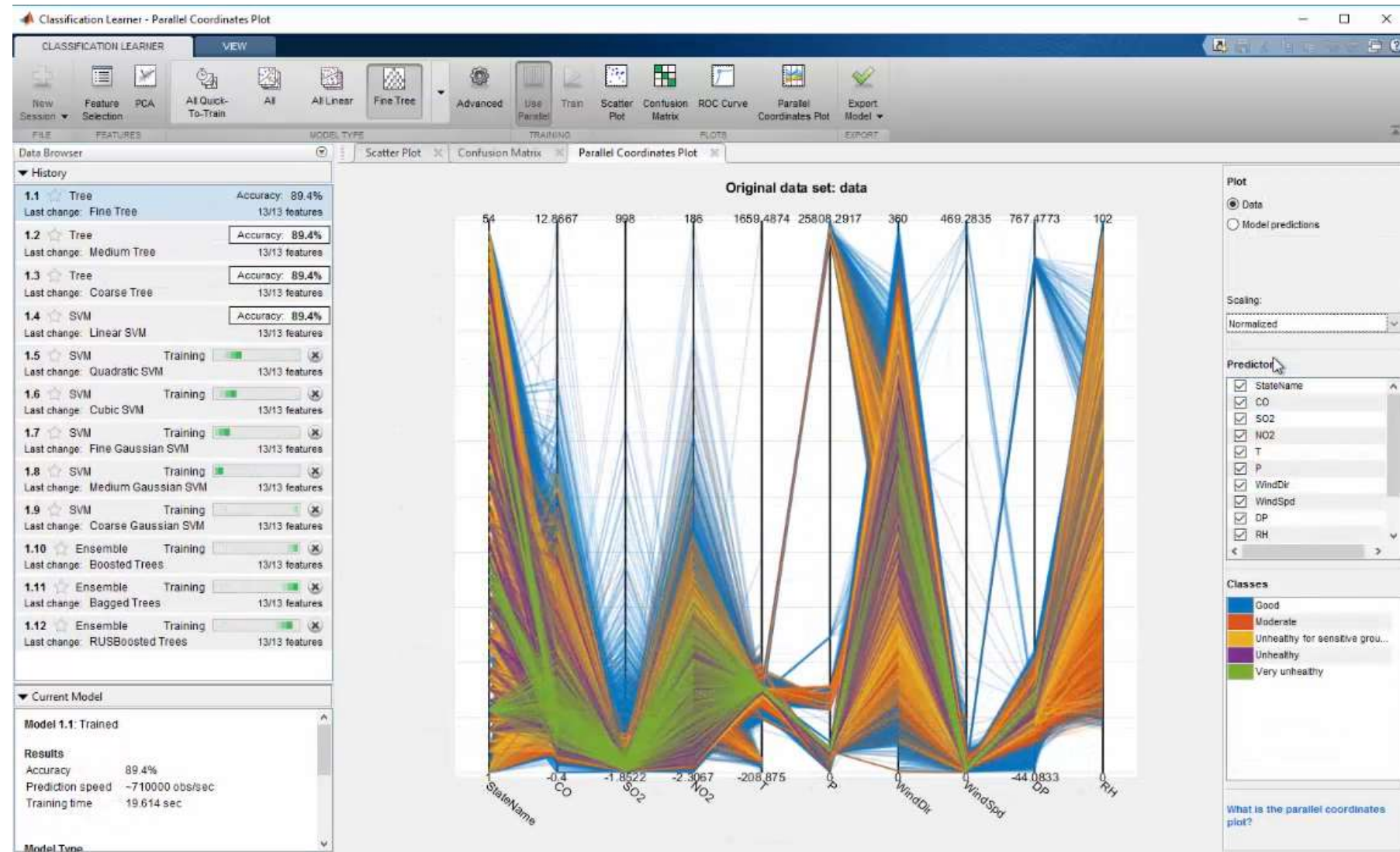
Train the Air Quality Prediction Model

Data
Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment



Model development is illustrated in this webinar [MATLAB with Data Science](#)

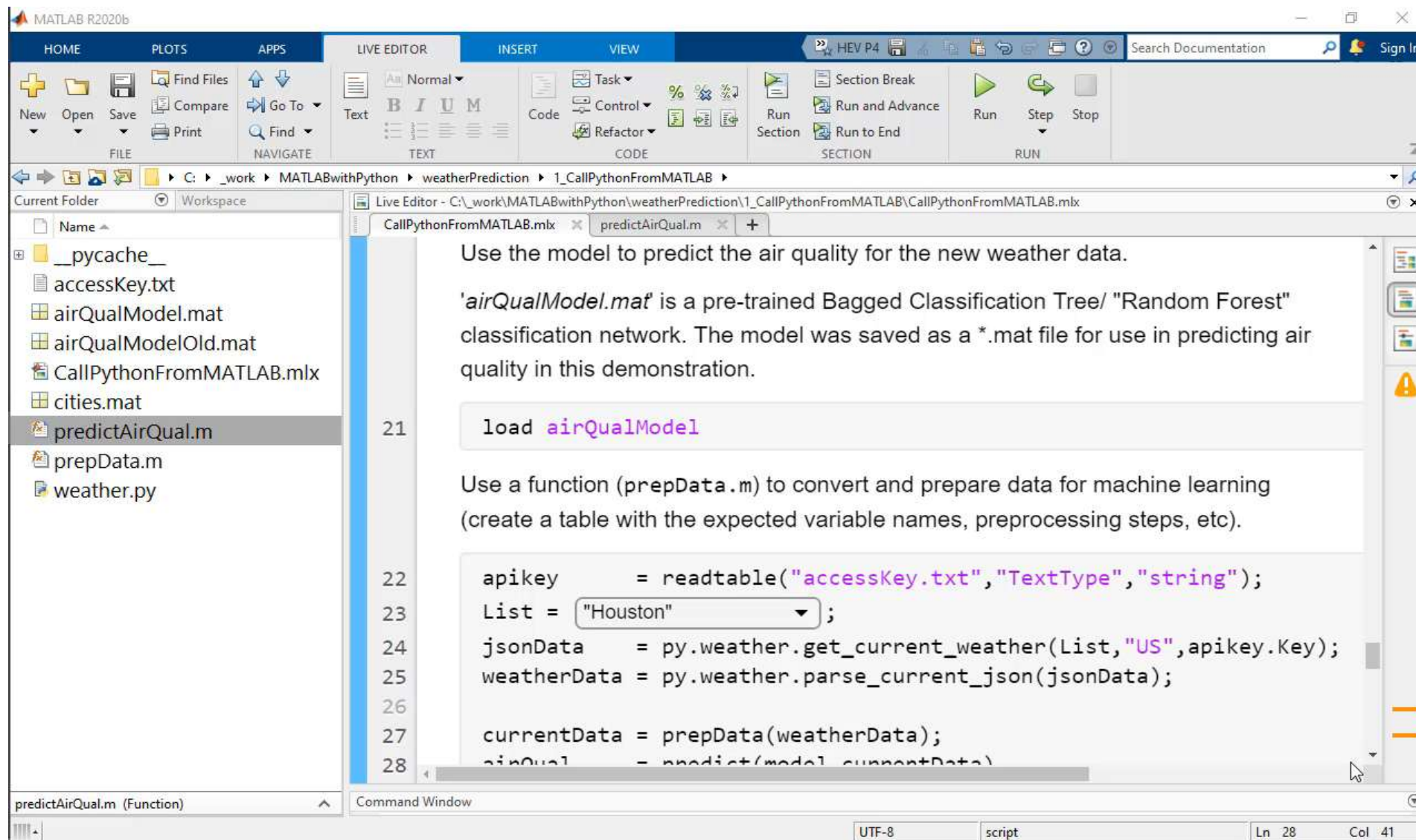
Call the Air Quality Prediction Model

Data
Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment



Data
Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

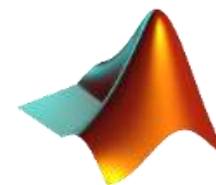
Deployment

Recap: Calling Python from MATLAB

Syntax differences when calling Python from MATLAB



Python



MATLAB

```
>>> import math  
>>> math.sqrt(42)
```



```
>> py.math.sqrt(42)
```

```
>>> print('hello', 'world', sep=', ')
```



```
>> py.print('hello', 'world', ...  
           pyargs('sep', ', '))
```

Data
Access

Co-Execution

- Call Python from MATLAB

- Call MATLAB from Python

Deployment

Data are automatically converted where possible

Otherwise convert explicitly

Data Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment

MATLAB to Python Data Type Mapping

R2021a

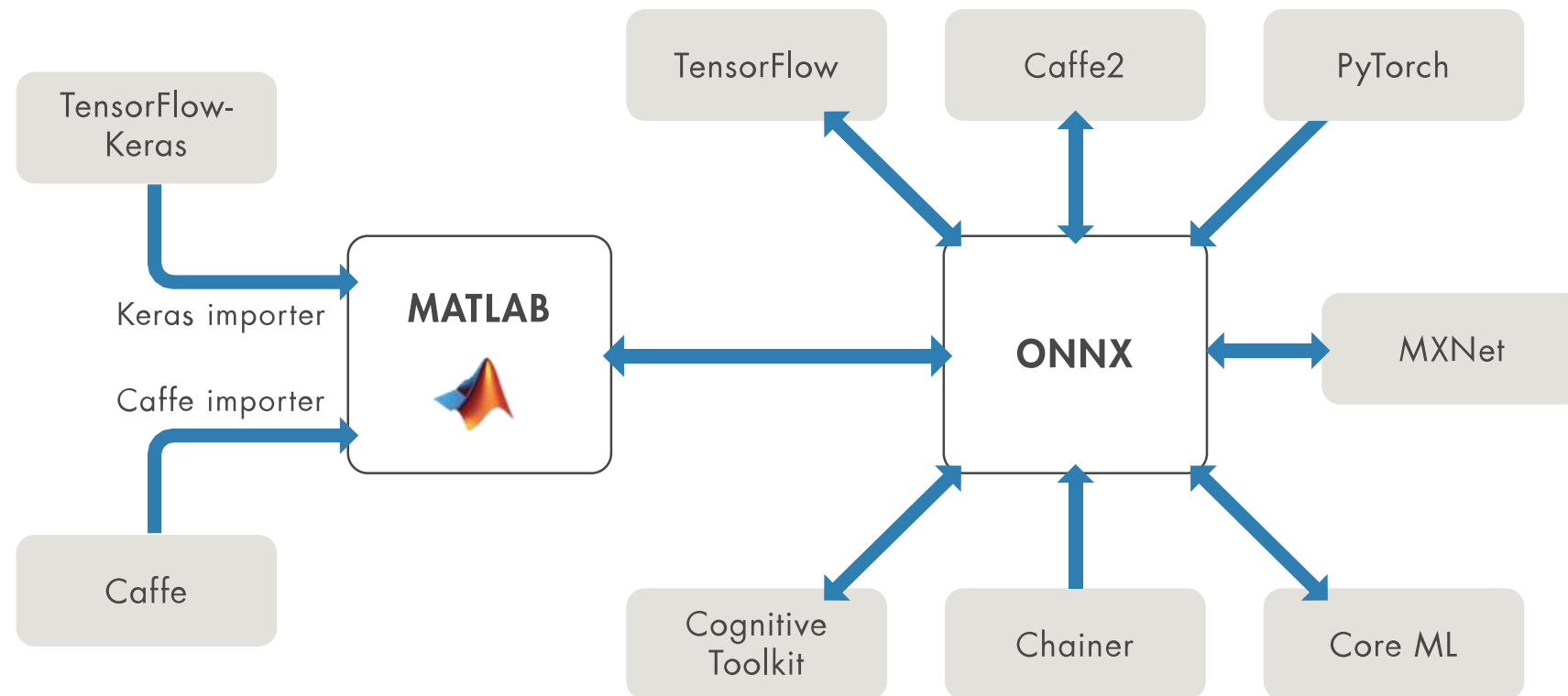
When calling a Python[®] function, MATLAB[®] converts MATLAB data into types that best represent the data to the Python language.

Pass Scalar Values to Python

MATLAB Input Argument Type — Scalar Values Only	Resulting Python py. Type	Examples
double single	float	Use Python Numeric Variables in MATLAB
Complex single Complex double	complex	<pre>z = complex(1,2); py.math.polar(z)</pre> <pre>ans = Python tuple with no properties. (2.23606797749979, 1.1071487177940904)</pre>
int8 uint8 int16 uint16 int32	int	
uint32 int64 uint64	int long (version 2.7 only)	
NaN	float("nan")	
Inf	float("inf")	
string scalar	str	Use Python str Variables in MATLAB

https://mathworks.com/help/matlab/matlab_external/passing-data-to-python.html

Model Interoperability



<https://www.mathworks.com/solutions/deep-learning/models.html>

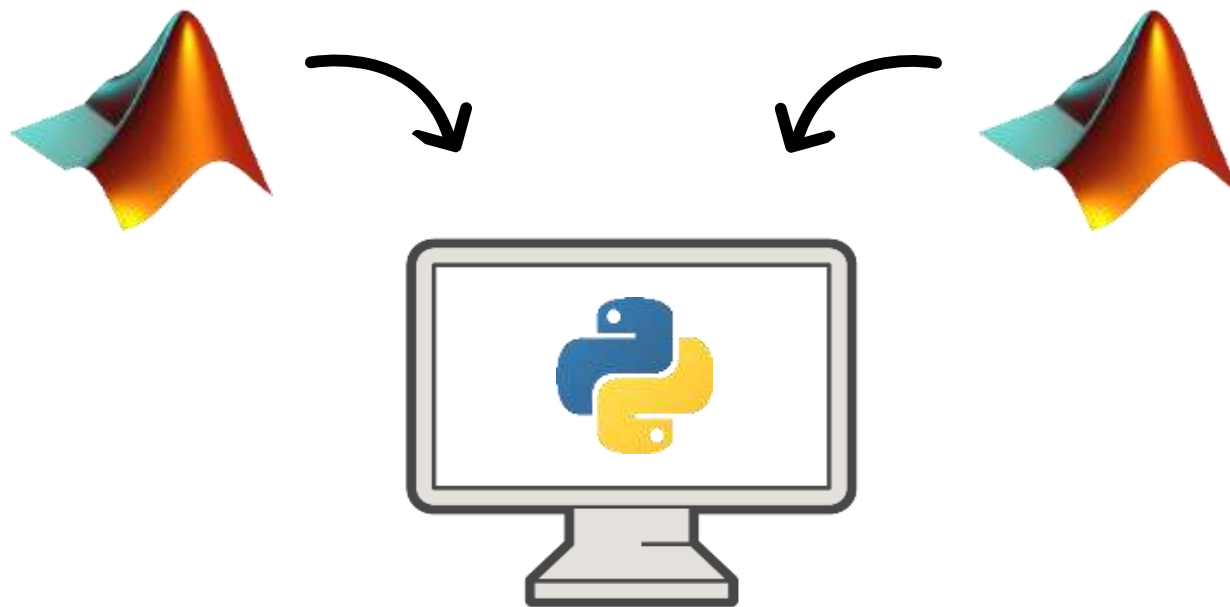


Call MATLAB from Python

Why call MATLAB from Python?

Already working in Python, and:

- Want to reuse existing MATLAB code
- Need functionality available in MATLAB
- Want to collaborate with MATLAB users



Data
Access

Co-Execution

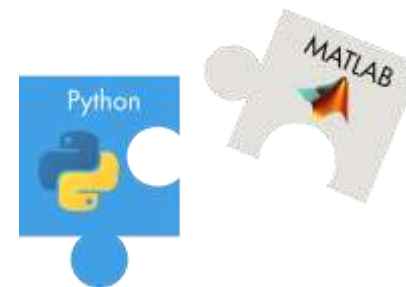
- Call Python
from MATLAB

- Call MATLAB
from Python

Deployment

Call MATLAB from Python

To perform advanced analytics



- Calling MATLAB from Python
 - via MATLAB Engine API

```
cd (fullfile(matlabroot, 'extern', 'engines', 'python'))  
system('python setup.py install')
```

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

1. Start a MATLAB process

```
[1] ▶ M↓  
  
import matlab.engine  
eng = matlab.engine.start_matlab()
```

2. Call MATLAB functions

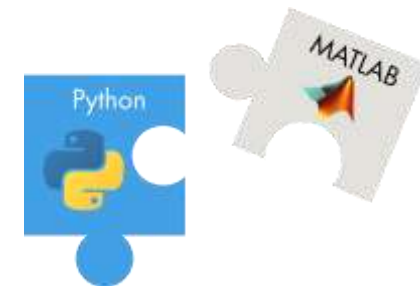
```
[2] ▶ M↓  
  
x = eng.sqrt(float(43))  
print(x)
```

6.557438524302

Deployment

Integrate the MATLAB model in a Python environment

[CallMATLABfromPython.ipynb](#)

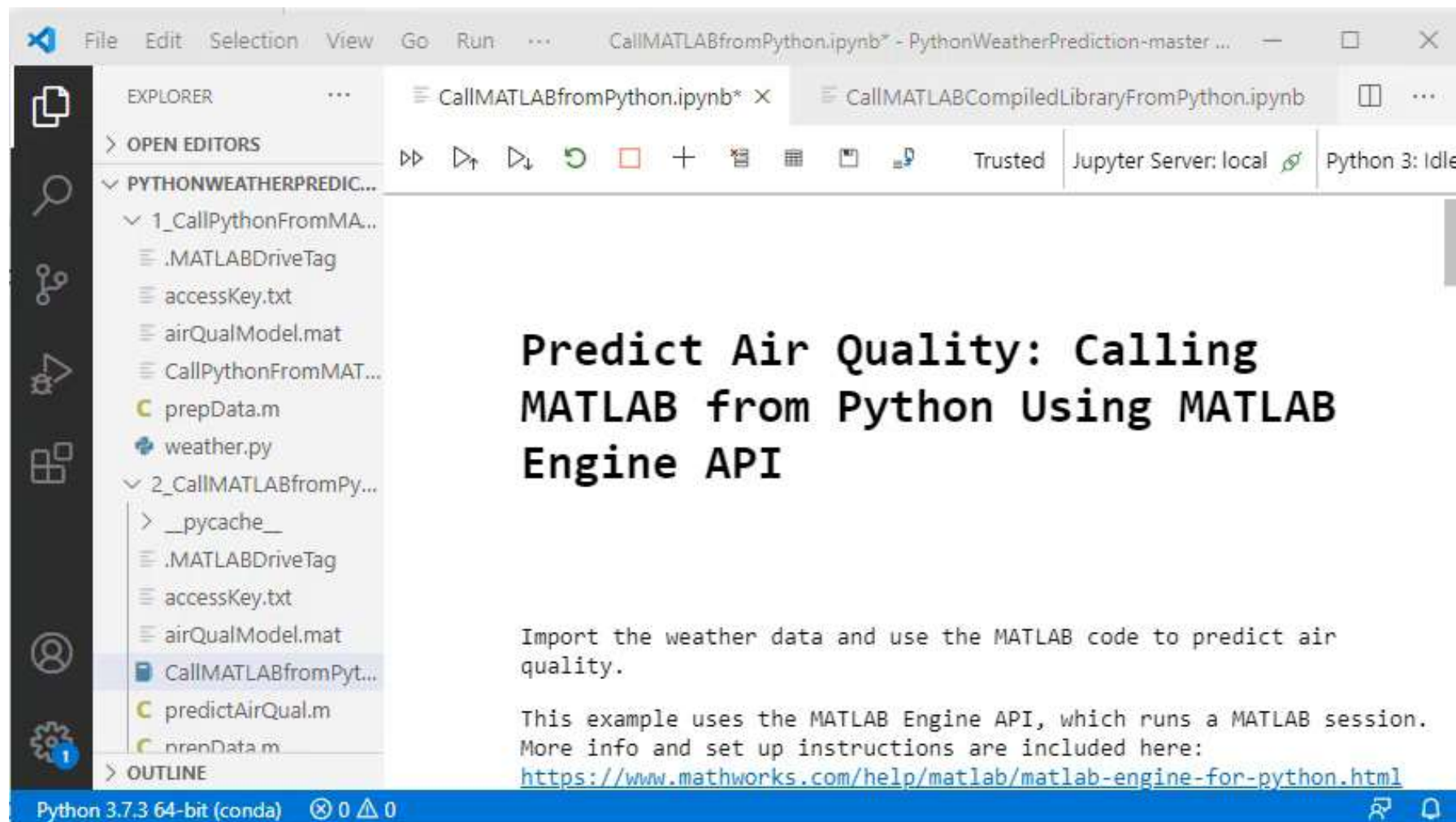


Data
Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment

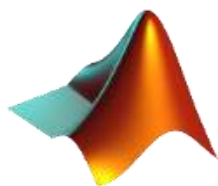


- Call Python from MATLAB
- Call MATLAB from Python

Recap: Calling MATLAB from Python

Syntax differences when calling MATLAB from Python

Data Access



MATLAB



Python

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

```
>> [s, idx] = sort(x)
```

```
>> foo(x)
```



```
>>> s = eng.sort(x, nargout=2)
```

```
>>> eng.foo(x, nargout=0)
```

Deployment

```
>> C = A + B
```



```
>>> C = eng.plus(A, B)
```

Data are automatically converted where possible

Data Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment

Pass Data to MATLAB from Python

R2021a

Python Type to MATLAB Scalar Type Mapping

When you pass Python® data as input arguments to MATLAB® functions, the MATLAB Engine for Python converts the data into equivalent MATLAB data types.

Python Input Argument Type — Scalar Values Only	Resulting MATLAB Data Type
float	double
complex	Complex double
int	int64
long (Python 2.7 only)	int64
float(nan)	NaN
float(inf)	Inf
bool	logical
str	char
unicode (Python 2.7 only)	char
dict	Structure if all keys are strings not supported otherwise

Python Container to MATLAB Array Type Mapping

Python Input Argument Type — Container	Resulting MATLAB Data Type
matlab numeric array object (see MATLAB Arrays as Python Variables)	Numeric array
bytearray	uint8 array
bytes (Python 3.x)	uint8 array
bytes (Python 2.7)	char array
list	Cell array
set	Cell array
tuple	Cell array

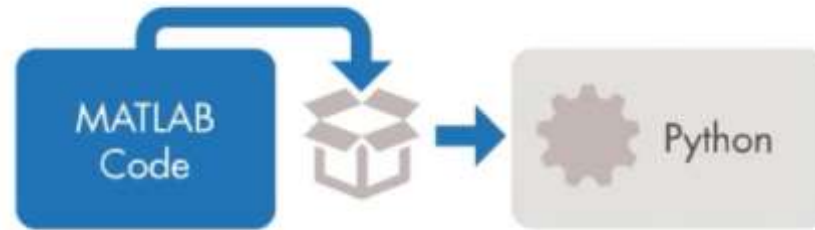
https://mathworks.com/help/matlab/matlab_external/pass-data-to-matlab-from-python.html



Deployment

Generate Python library from MATLAB functions

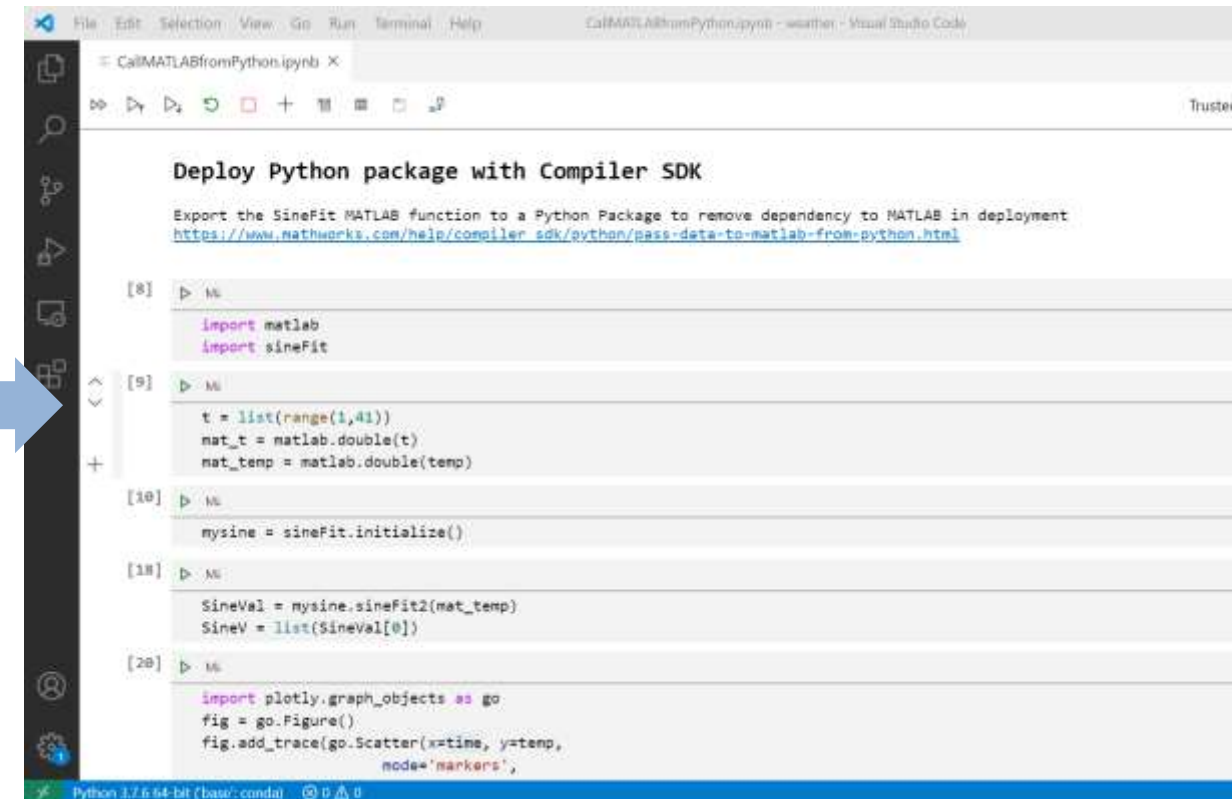
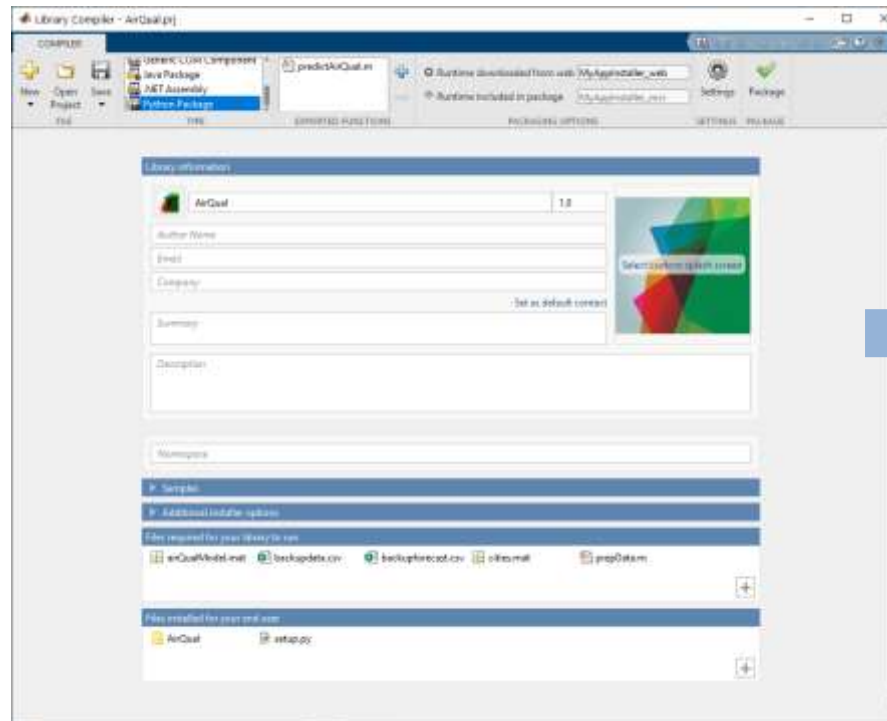
Data Access



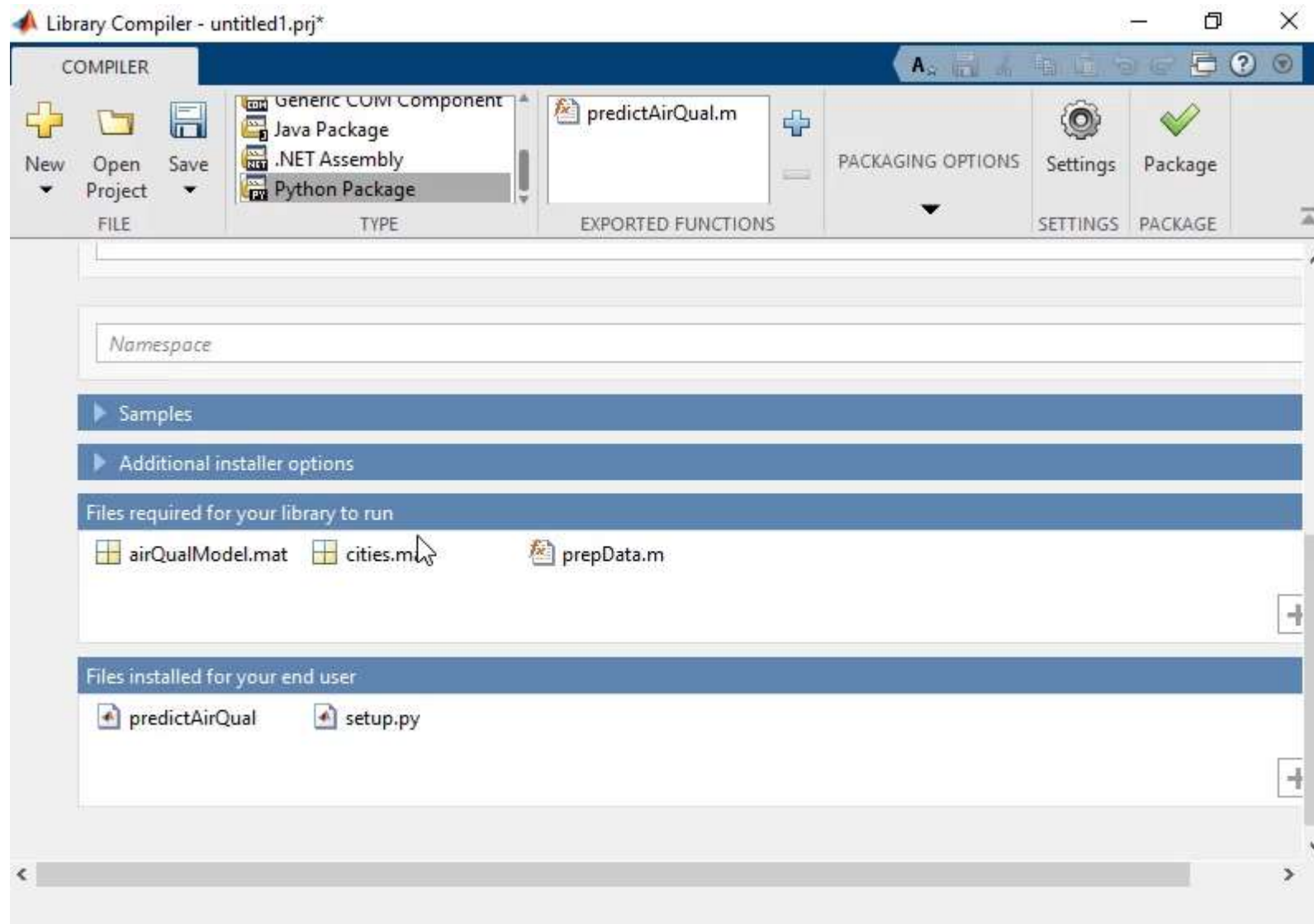
Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment



Generate Python library from MATLAB functions



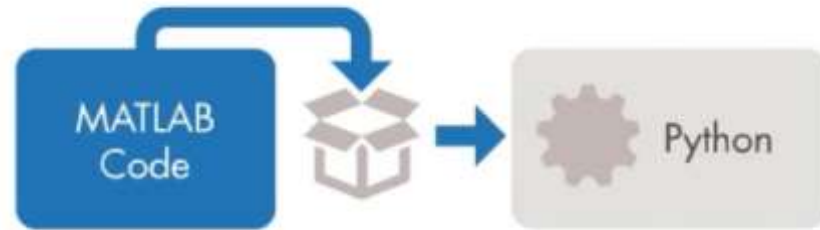
Data
Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment

Generate Python library from MATLAB functions

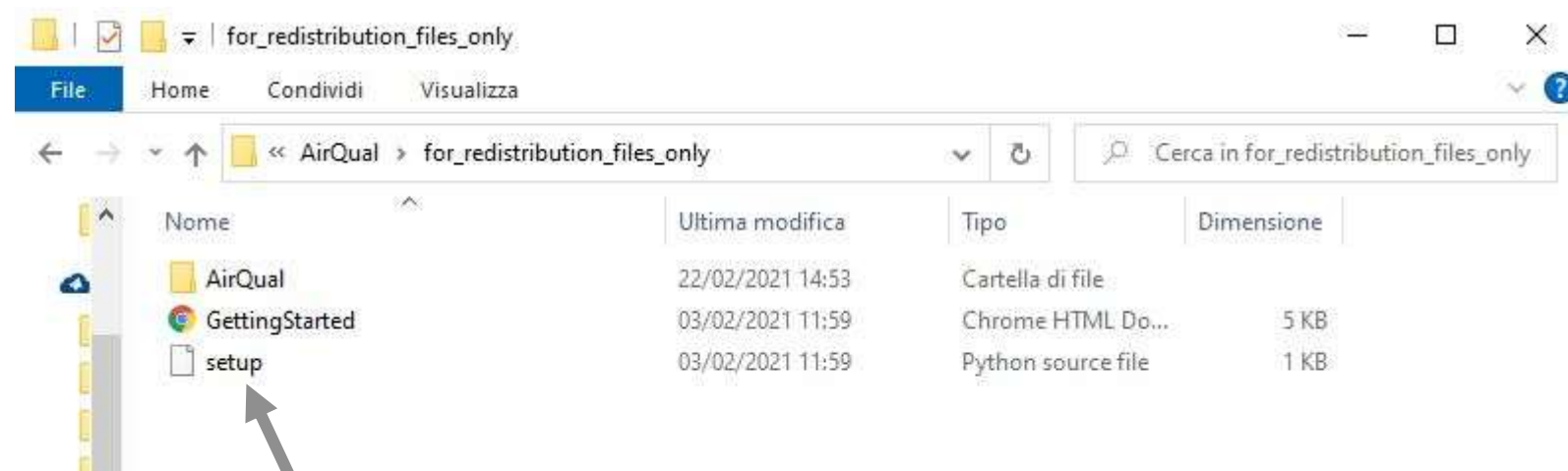


Data
Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment



Install library

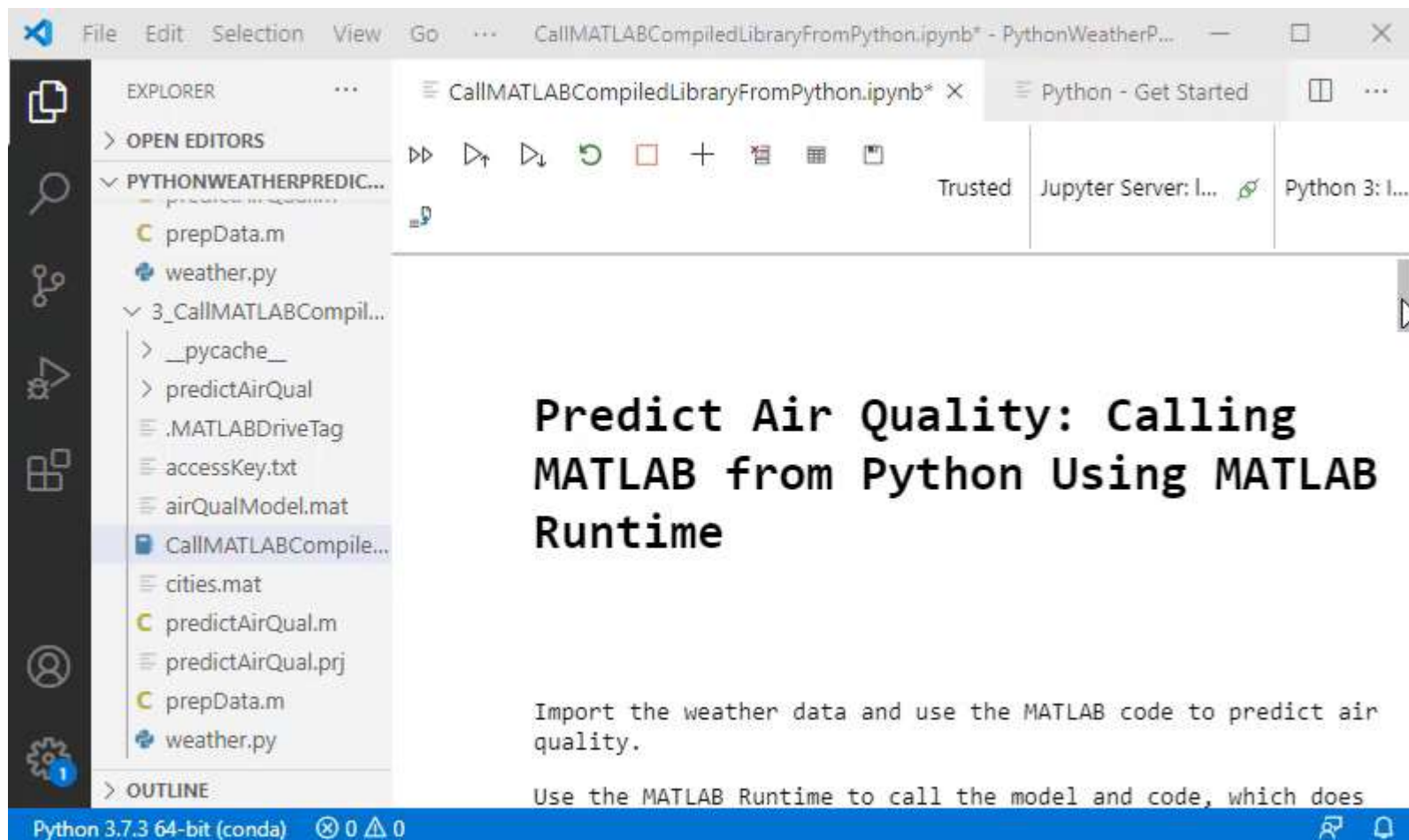
Execute Python library from MATLAB functions

Data Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment



MATLAB Production Server

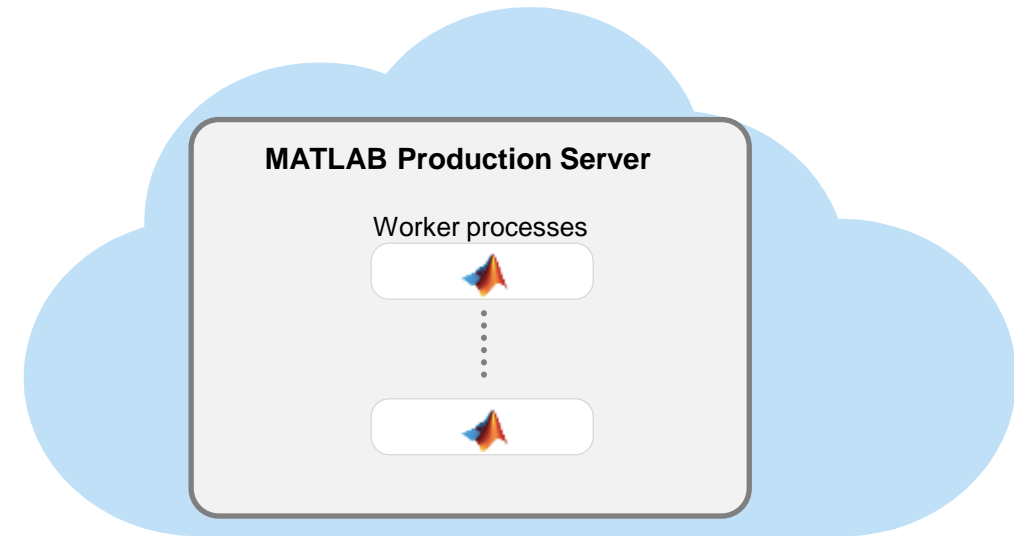
Access functions as web services

Data
Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment



Calling our function:

```
{"nargout":1,"rhs":["input"]}
```

Getting the result:

```
{"lhs":[{"mwdata":["output"],"mwsizе":[1,6],"mwtype":"char"}]}
```

Execute Python library from MATLAB functions

Data Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment

Air Quality Calculator

Archivio | C:/Users/gcarniel/MATLAB%20Drive/Python/PythonWeatherPrediction-master/4_Call...

Air Quality Conditions

Determine air quality conditions in your area.

Location:

The air quality is **Good**.
The current temperature is **30.15 F**.

Copyright © 2018-2020 MathWorks, Inc.

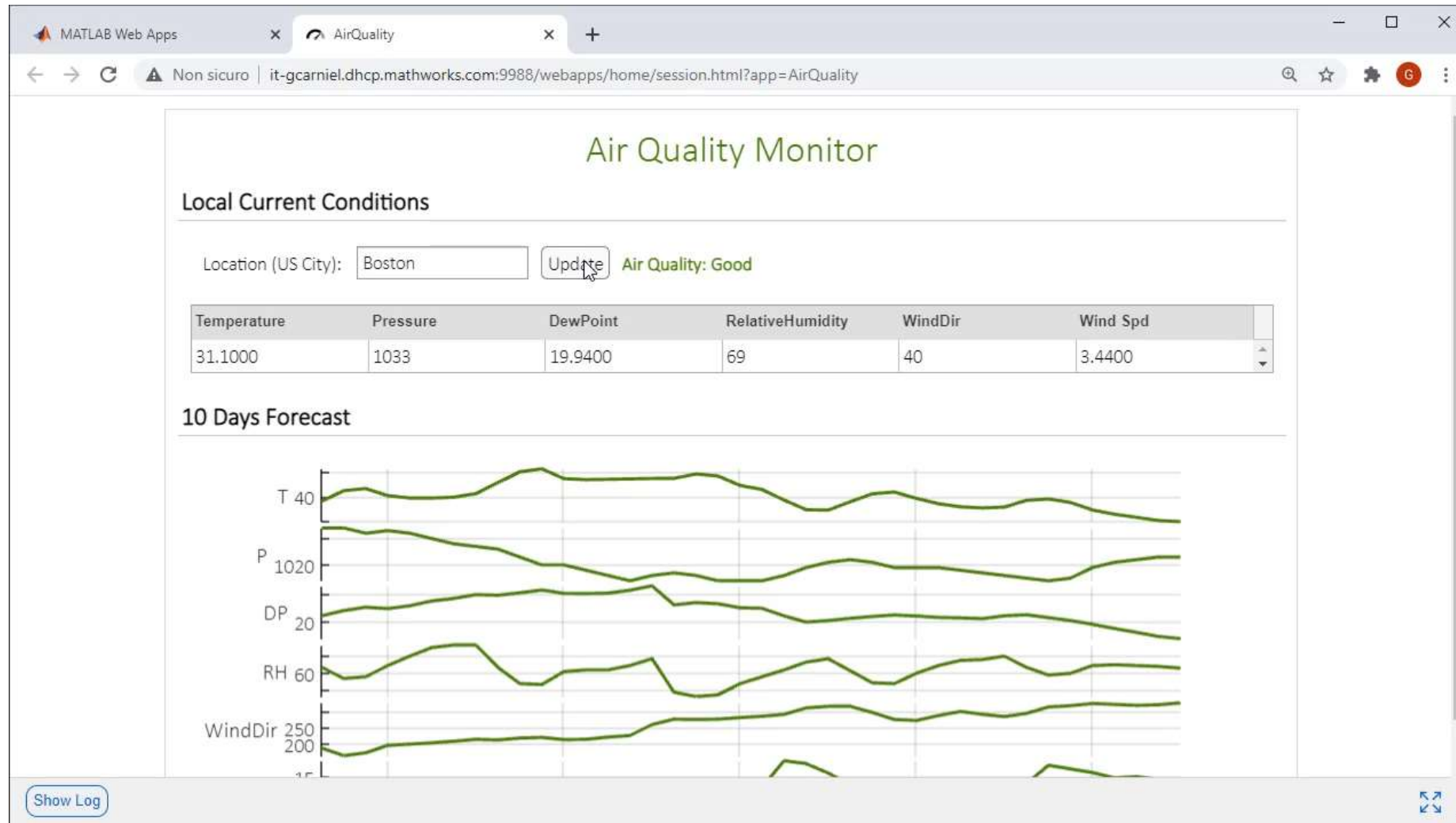
Share MATLAB App in the Web – Central Deployment

Data Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment



MATLAB App Designer

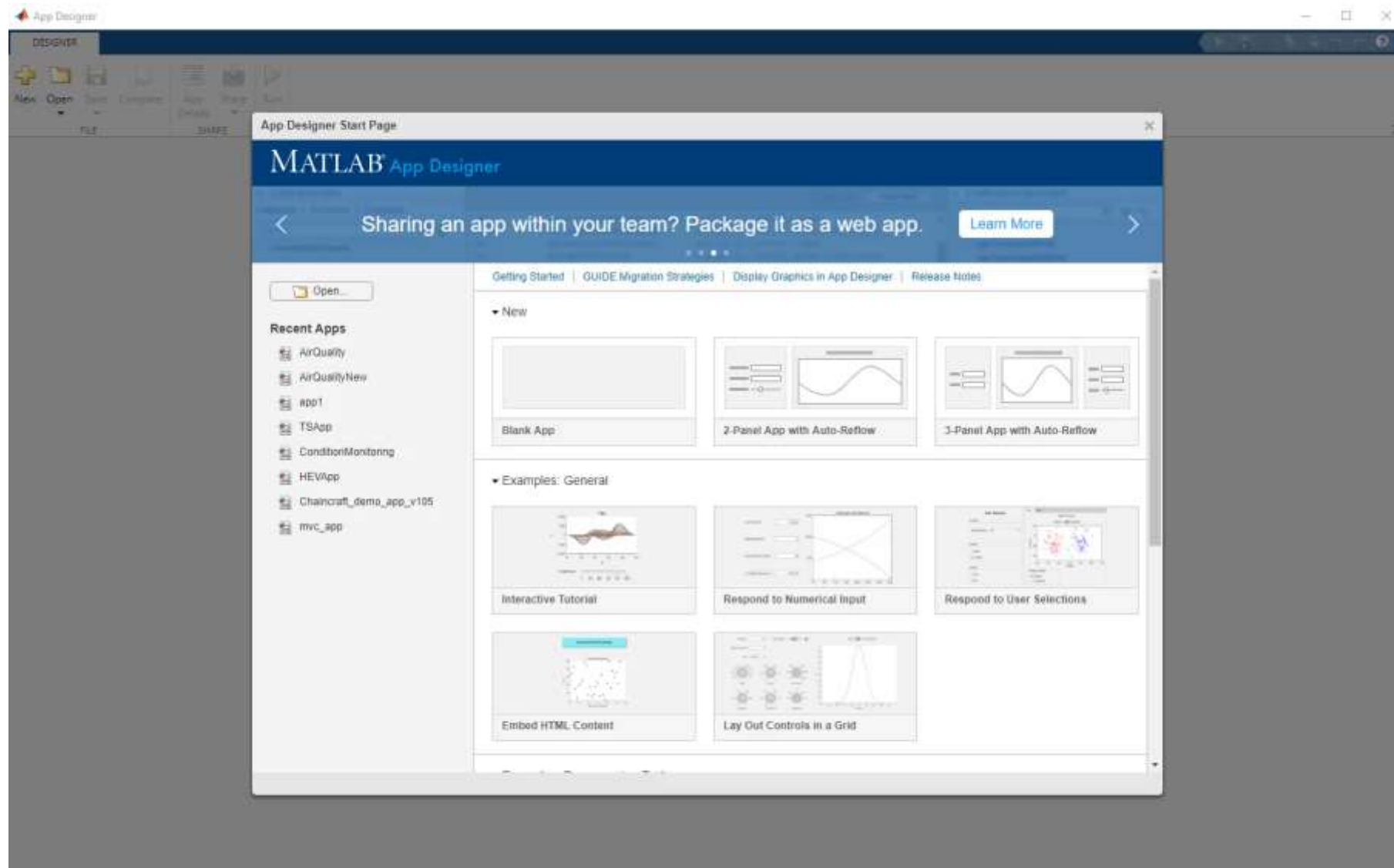
App development for Non-Programmers

Data
Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment



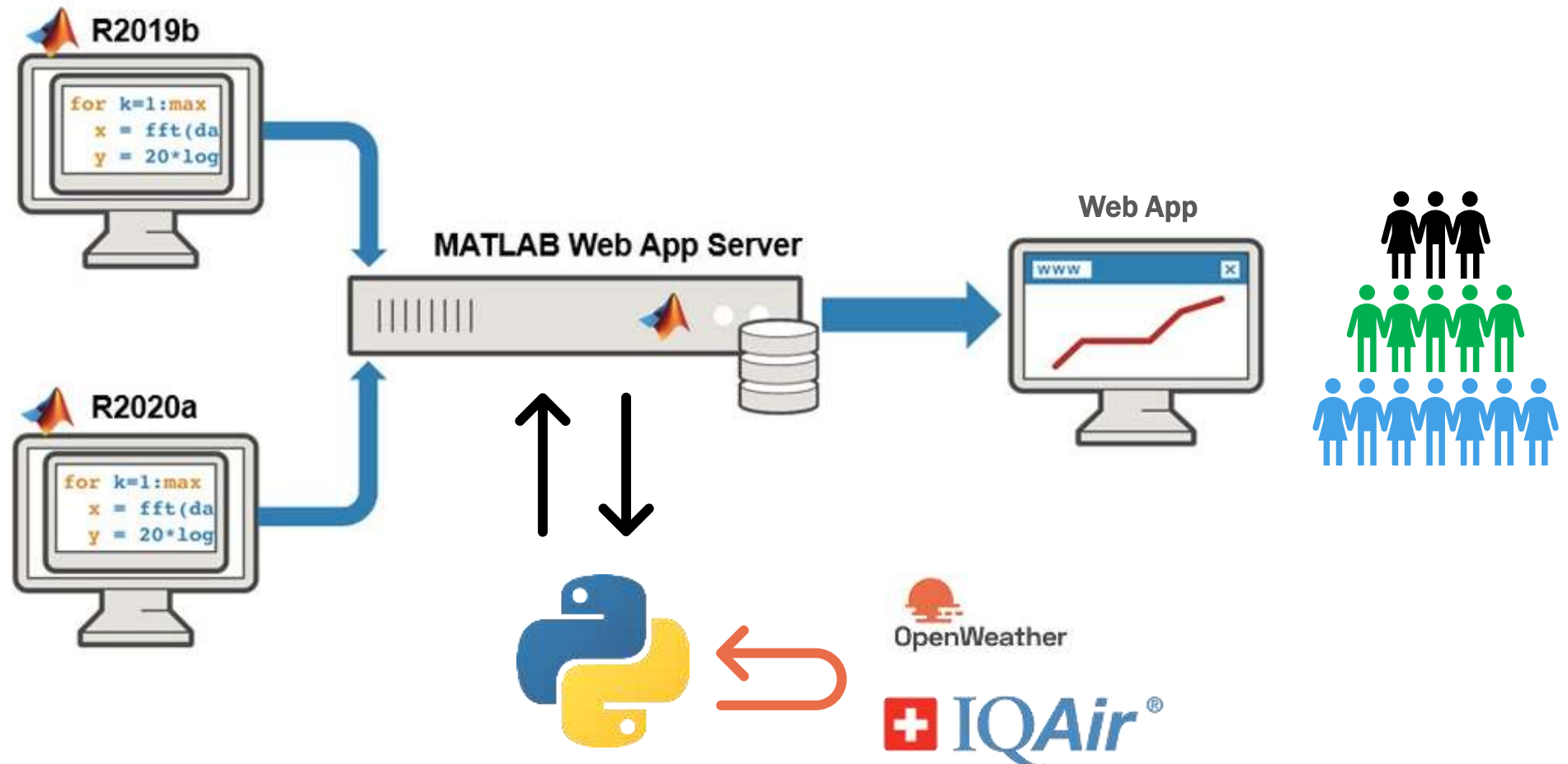
MATLAB Web App Server – Central Deployment

Data
Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment



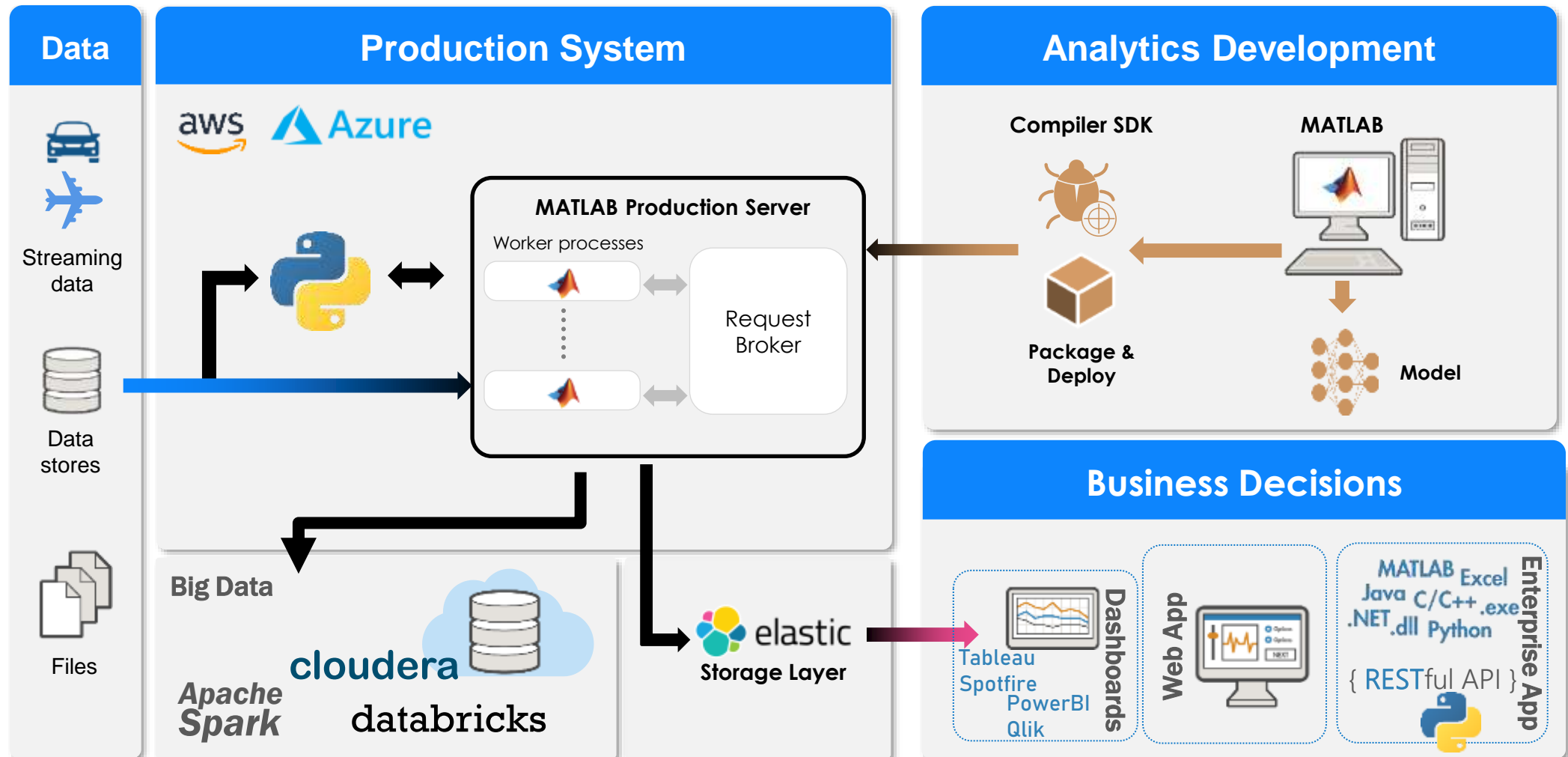
Integrate your Production System in an IT ecosystem

Data
Access

Co-Execution

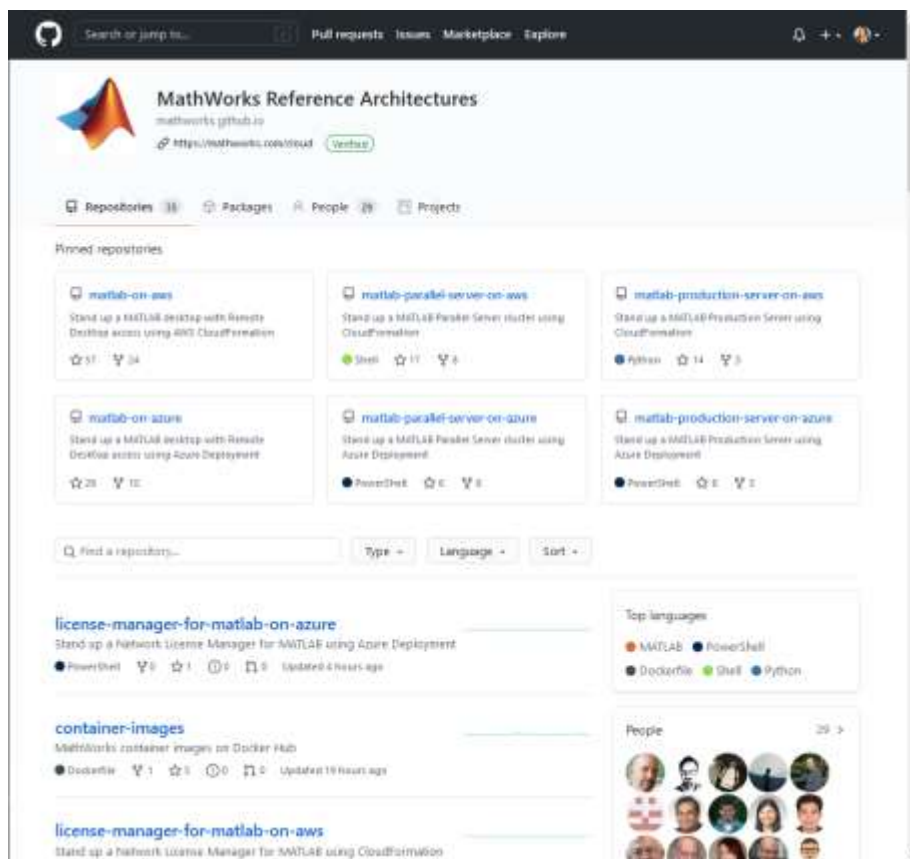
- Call Python from MATLAB
- Call MATLAB from Python

Deployment



Use MATLAB Reference Architectures for easy cloud setup, Dockerfiles, and interfaces to OSS

- <https://github.com/mathworks-ref-arch/matlab-dockerfile>



<https://github.com/mathworks-ref-arch>

Data
Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

Deployment

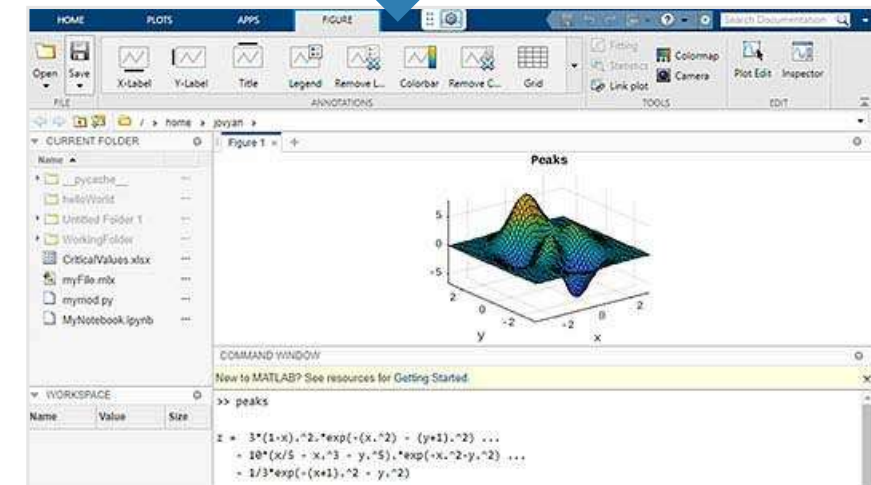
MATLAB Integration for Jupyter

Data
Access

Co-Execution

- Call Python from MATLAB
- Call MATLAB from Python

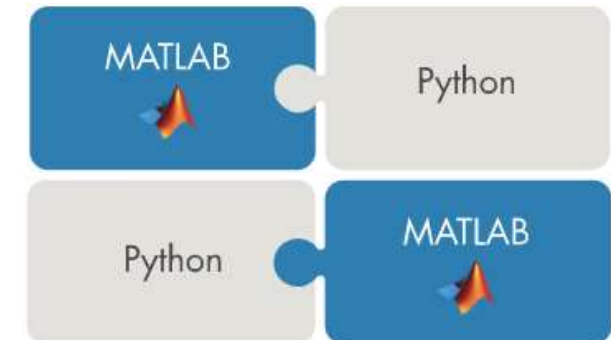
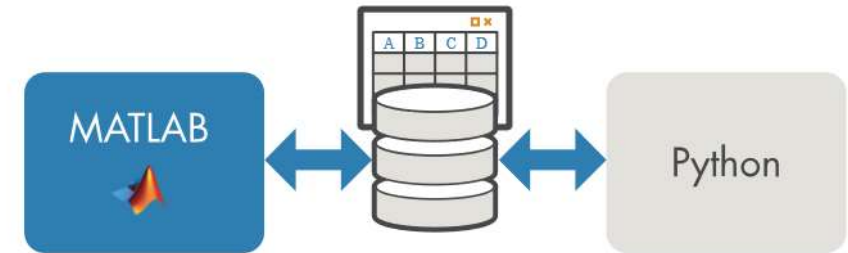
Deployment



[MATLAB Integration for Jupyter \(mathworks.com\)](https://mathworks.com)

Summary: Using MATLAB with Python

- Access Data
 - Weather App example
- Interoperability
 - Calling libraries written in Python from MATLAB
 - Calling MATLAB from Python
- Deploy Apps & Algos
 - Web App
 - Production API



Additional resources


Resources

- General:
 - <https://www.mathworks.com/products/matlab/matlab-and-python.html>
- Python from MATLAB:
 - <https://www.mathworks.com/help/matlab/call-python-libraries.html>
- MATLAB from Python:
 - MATLAB Engine API:
 - <https://www.mathworks.com/help/matlab/matlab-engine-for-python.html>
 - MATLAB Compiler SDK:
 - https://www.mathworks.com/help/compiler_sdk/python_packages.html
 - Data type conversions:
 - <https://www.mathworks.com/help/matlab/python-data-types.html>
- Example:
 - <https://github.com/mathworks/matlab-with-python>

Cheatsheet



Using MATLAB® and Python® Together

The  icon provides links to relevant sections of the MATLAB documentation to learn more.

Call Python in MATLAB

Access settings and status of Python interpreter:

```
>> pe = pyenv
```

Specify version to use:

```
>> pe = pyenv("Version",3.7)
```

Call Python modules and functions:

```
py.module_name.function_name
```

```
>> py.math.sqrt(42)
```

Pass keyword arguments

Use pyargs to pass keyword arguments

```
>>> foo(5,bar=42)
```


```
>> py.foo(5,pyargs('bar',42))
```

Reload modules

Reload the module after making updates:

```
>> py.importlib.reload(module)
```

Call MATLAB in Python

Install MATLAB Engine API for Python 

Run `setup.py` from an OS command window

```
$ cd [matlabroot]/extern/engines/python
```

```
$ python setup.py install
```

Call MATLAB functions

Import the module and start the engine

```
>>> import matlab.engine
```

```
>>> eng =
    matlab.engine.start_matlab()
```

Call functions through the engine

```
>>> x = eng.sqrt(42.0)
```

Capture multiple outputs

```
>>> x = eng.gcd(42.0,8.0,nargout=3)
```

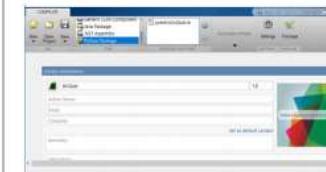
Stop the engine

```
>>> eng.exit()
```

Create Python Package

Package MATLAB functions 

Use the Library Compiler App to create a Python package for MATLAB functions



Invoke MATLAB functions from the Python package

```
>>> import PackageName
```

```
>>> pkg =
```

```
    PackageName.initialize()
```

```
>>> result = pkg.foo()
```

Close package

```
>>> pkg.terminate()
```

MATLAB Answers – tag:"python"

The screenshot shows the MATLAB Answers web interface. The browser address bar displays the URL: <https://www.mathworks.com/matlabcentral/answers/?term=tag%3A%22python%22>. The page header includes the MathWorks logo and navigation links: Products, Solutions, Academia, Support, Community, and Events. A search bar in the header contains the text "tag:'python'" and a search button. Below the header, a navigation bar shows "MATLAB Answers" and a search bar with the same text. The main content area displays "Results for tag:'python' (614 results)". On the left, there are filter sections for "Status", "Source", and "Product". The "Status" section shows 370 Answered, 244 Unanswered, and 161 Answer Accepted. The "Source" section shows 603 Community and 11 MathWorks Support. The "Product" section shows 215 MATLAB, 10 Simulink, 1 Audio Toolbox, 1 Communications Toolbox, 1 Database Toolbox, and 5 Deep Learning Toolbox. The main results list shows three items: 1. "Matlab 2019a pyenv.Library empty when using Python 2.7" with 0 answers, 0 votes, and 6 views. 2. "Error 127 Details: Unable to load bundle binary C:\Program Files\MATLAB\R2020b\bin\win64\builins\matlab_toolbox_general_builtins\mwwkspintrospect_builtinimpl.dll" with 1 answer, 0 votes, and 35 views. 3. "problem with python numpy" with 5 answers, 0 votes, and 35 views.

Search results: tag:"python" - M

<https://www.mathworks.com/matlabcentral/answers/?term=tag%3A%22python%22>

MATLAB Answers

Products Solutions Academia Support Community Events

Get MATLAB

tag:"python" Answers

MATLAB Central Home Ask Answer Browse MATLAB FAQs Contributors More Help

Filter by

Status

Answered	370
Unanswered	244
Answer Accepted	161

Source

Community	603
MathWorks Support	11

Product

MATLAB	215
Simulink	10
Audio Toolbox	1
Communications Toolbox	1
Database Toolbox	1
Deep Learning Toolbox	5

Results for tag:"python" (614 results)

Sort by: Date updated (Newest–Oldest)

Subscribe to this View

Results 1 - 50 of 614

0 answers

Matlab 2019a pyenv.Library empty when using Python 2.7

Asked by Simon Geoffroy-Gagnon on 9 Oct 2020 at 22:29

Tags python, pyenv, load python

0 votes
6 views

1 answer

Error 127 Details: Unable to load bundle binary C:\Program Files\MATLAB\R2020b\bin\win64\builins\matlab_toolbox_general_builtins\mwwkspintrospect_builtinimpl.dll

Asked by Mehul Garg on 1 Oct 2020 at 7:18

Latest activity Commented on by Mehul Garg on 9 Oct 2020 at 4:26

Tags bundle 117, error 127, flask, python, mcr

Products MATLAB Compiler, MATLAB Compiler SDK

0 votes
35 views

5 answers

problem with python numpy

Asked by Johann Thurn on 31 Oct 2018

<https://www.mathworks.com/matlabcentral/answers/?term=tag%3A%22python%22>

Additional Resources

- **NEW** Release
 - https://www.mathworks.com/products/new_products/latest_features.html
- Predictive Analytics
 - <https://www.mathworks.com/discovery/predictive-analytics.html>
- Deep Learning
 - <https://www.mathworks.com/discovery/deep-learning.html>
- Reinforcement Learning
 - <https://www.mathworks.com/products/reinforcement-learning.html>

