



GreenPoint
Financial

FORDHAM UNIVERSITY

FRTB CAPSTONE COURSE 2019

Technology Tools everybody should know



Tim Glauner

Head Capital Markets Americas - Global Solution Consulting

tim.glauner@finastra.com

SOME REMARKS

- There is a fine line between programming and using just technology tools to do your work
 - VBA and Excel
 - Low Code Development environments such as Salesforce or Node-Red
 - High level AI/ML libraries
 - Mathematica
- Even the business is being asked to know technology
 - Goldman Sachs requires traders to take python course
- Easy for students to use technology tools but programming and scripting will make you much more efficient

TOOLS

➤ Ping

➤ Curl

- Send http requests to REST calls

➤ Postman

- Sends http requests to REST calls

➤ Python

- Obvious ones: Panda, Numpy, Anaconda, ...
- XML parser try bs4 (BeautifulSoup). It's the most common HTML parser and can also parse XML. Source: Daniel Groner, Assistant Clinical Professor
- Flask web server to build REST calls and Microservice

➤ Stackoverflow

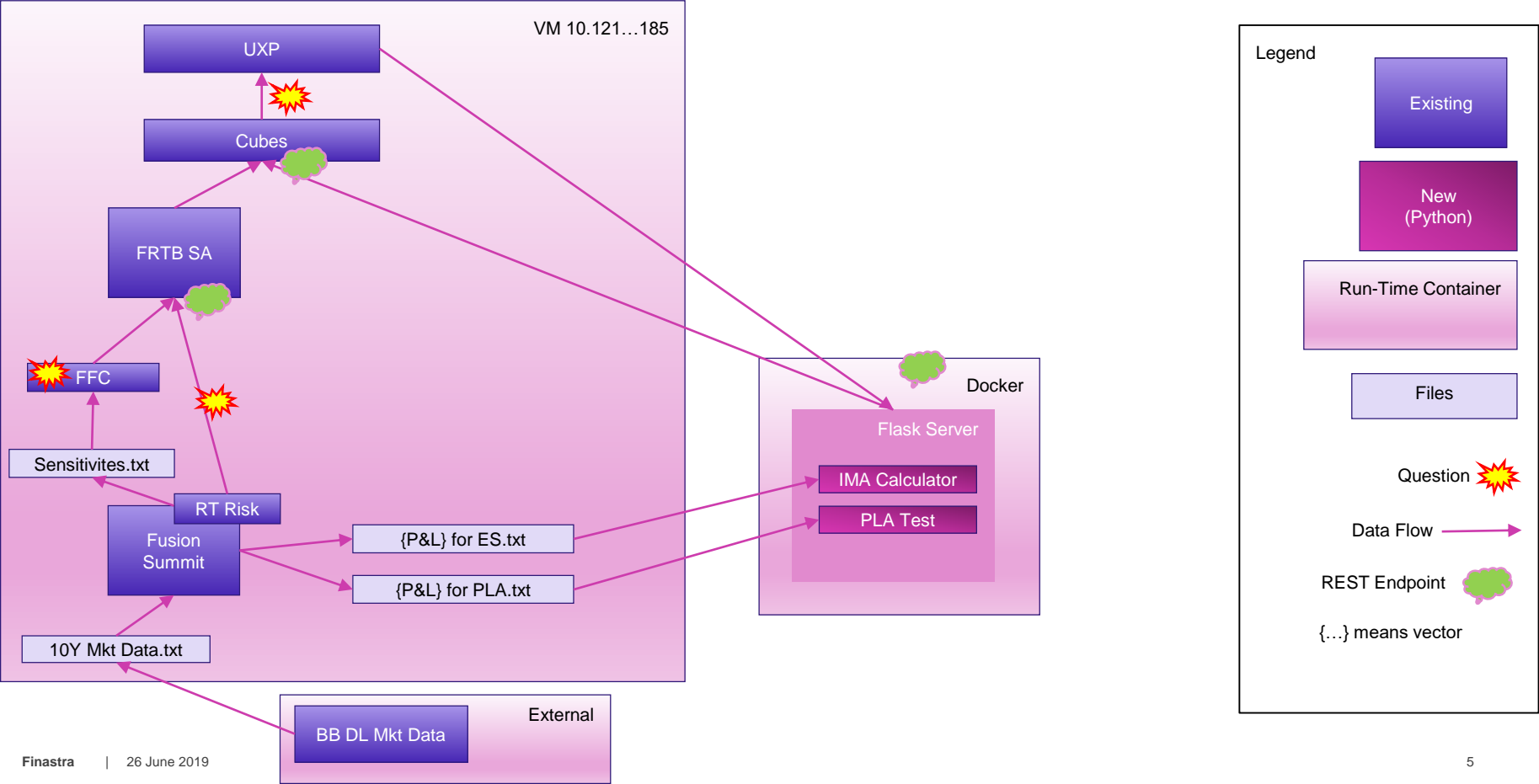
- Write good replication steps
- Stuck for 2 days (up to you how long to wait) post and it usually gets resolved really quickly
- <https://stackoverflow.com/questions/56727763/cannot-connect-from-chrome-to-flask-server-that-runs-in-docker-container-on-wind>

TOOLS (CONT.)



.

FORDHAM 2019 AND BANK'S CUSTOM APPLICATION ARCHITECTURE



DOCKER

Adjust to your liking

- Install python 2.7 or 3.x
- Install Docker Desktop Windows 10
- Install Flask with pip
- Have Chrome around

```
Dockerfile
FROM ubuntu:latest
RUN apt-get update -y
RUN apt-get install -y python-pip
RUN apt-get install -y python-dev
RUN apt-get install -y build-essential
COPY . /app
WORKDIR /app
RUN pip install -r requirements.txt
ENTRYPOINT ["python"]
CMD ["app.py"]
```

```
app.py
requirements.txt

from flask import Flask

app = Flask(__name__)

@app.route('/')
def main():
    print 'I am here'
    return 'Hello Stranger!'

@app.route('/insert')
def insert():
    print 'define some sample data'
    #Do REST call to Cube
    return 'Insert Called'

if __name__ == '__main__':
    app.run(host='0.0.0.0')
```

```
requirements.txt
Flask==0.12
```

DOCKER (CONT)

➤ To Test

- c:\Temp\web>**docker build -t tg-flask-sample .**
Sending build context to Docker daemon 4.096kB
...
Successfully built c6ded0c46f87
Successfully tagged tg-flask-sample:latest
- c:\Temp\web> **docker run -d -p 5000:5000 tg-flask-sample**
- c:\Temp\web> **curl localhost:5000**
Hello Stranger!
- c:\Temp\web> curl localhost:5000/insert
Insert Called

RUN THROUGH

C:\Temp\web>docker ps

CONTAINER ID	IMAGE	NAMES
3c8991c9e920	tg-flask-sample	cocky_yalow

C:\Temp\web>docker stop cocky_yalow
cocky_yalow

C:\Temp\web>docker ps

CONTAINER ID	IMAGE	COMMAND
--------------	-------	---------

C:\Temp\web>docker ps -a

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS
9a049681b9e9	tg-flask-sample	"python app.py"	About a minute ago	Exited (137) 45 seconds ago
3c8991c9e920	tg-flask-sample	"python app.py"	About a minute ago	Exited (137) 16 seconds ago

C:\Temp\web>docker rm musing_mestorf
musing_mestorf

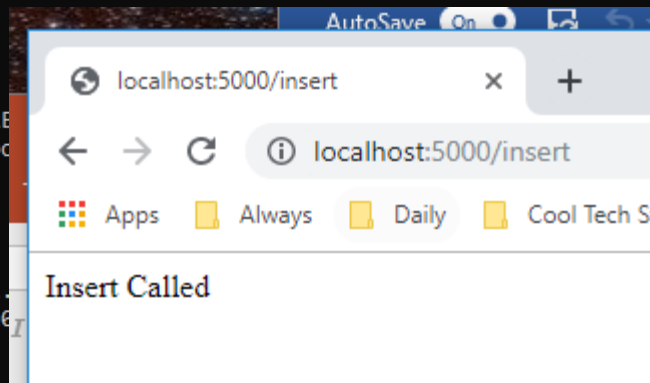
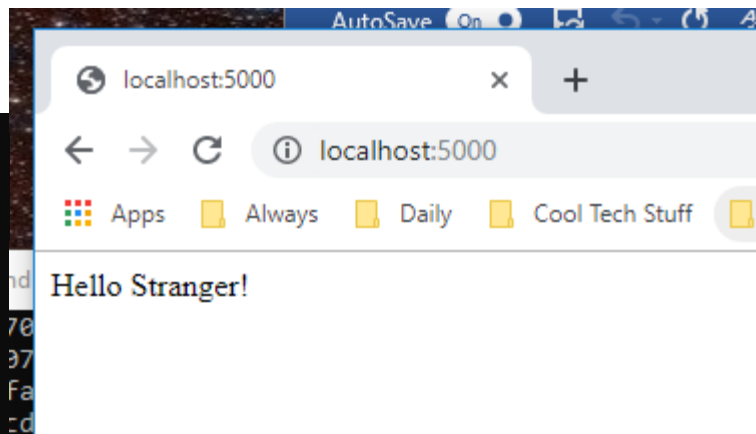
C:\Temp\web>docker ps -a

CONTAINER ID	IMAGE	COMMAND
3c8991c9e920	tg-flask-sample	"python app.py"

C:\Temp\web>docker run -p 5000:5000 tg-flask-sample

```
* Running on http://0.0.0.0:5000/ (Press CTRL+C to quit)
172.17.0.1 - - [25/Jun/2019 18:34:04] "GET / HTTP/1.1" 200 -
172.17.0.1 - - [25/Jun/2019 18:34:10] "GET /favicon.ico HTTP/1.1" 404 -
172.17.0.1 - - [25/Jun/2019 18:35:21] "GET /insert HTTP/1.1" 200 -
```

C:\Temp\web>curl localhost:5000
Hello Stranger!
C:\Temp\web>



FORDHAM 2019 AND BANK'S CUSTOM APPLICATION ARCHITECTURE

