

Introducing Docker

Application Containerization
&
Service Orchestration

Sawood Alam <[@ibnesayeed](#)>

Old Dominion University
Norfolk, Virginia - 23529 (USA)

1

W

S

D

L



docker

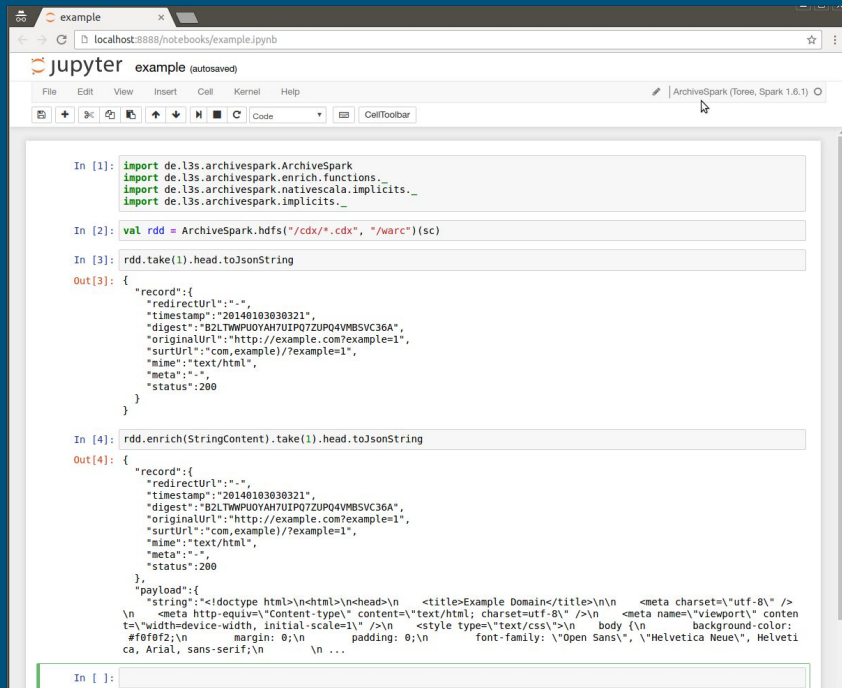
Dockerizing ArchiveSpark

A 30 minutes of *Pair Hacking* made running a complex software as easy as:

```
$ docker run -p 8888:8888 ibnesayeed/archivespark
```

Read the full story at:

<https://ws-dl.blogspot.com/2016/07/2016-07-21-dockerizing-archivespark.html>



```
In [1]: import de.l3s.archivespark.ArchiveSpark
import de.l3s.archivespark.enrich.functions._
import de.l3s.archivespark.nativescala.implicit._
import de.l3s.archivespark.implicit._

In [2]: val rdd = ArchiveSpark.hdfs("/cdx/*.*cdx", /*warc*/(sc))

In [3]: rdd.take(1).head.toJsonString

Out[3]: {
  "record": {
    "redirectUrl": "-",
    "timestamp": "20140103030321",
    "digest": "B2L7WMPUOYAH7UIP07ZUPQ4VMB5VC36A",
    "originalUrl": "http://example.com?example=1",
    "surtUrl": "com,example)/?example=1",
    "mime": "text/html",
    "meta": "-",
    "status": 200
  }
}

In [4]: rdd.enrich(StringContent).take(1).head.toJsonString

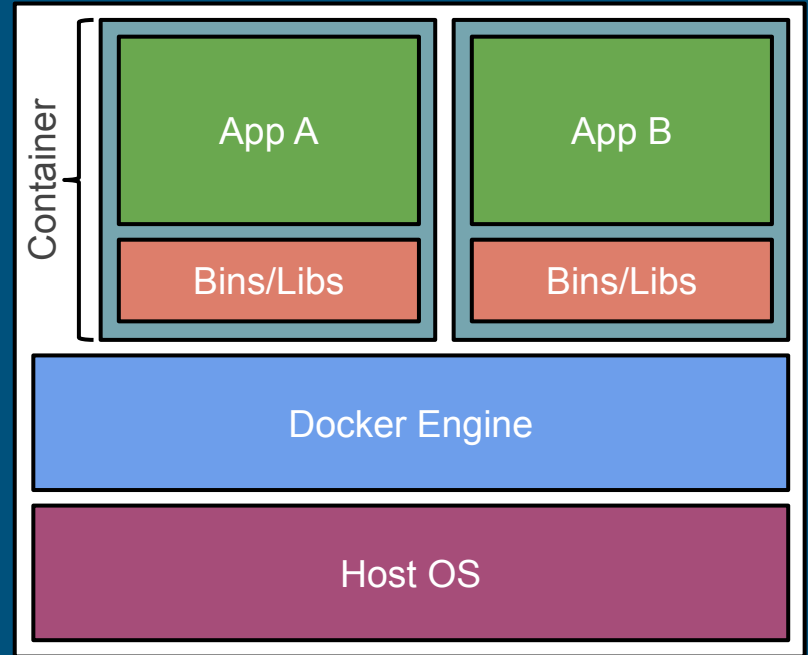
Out[4]: {
  "record": {
    "redirectUrl": "-",
    "timestamp": "20140103030321",
    "digest": "B2L7WMPUOYAH7UIP07ZUPQ4VMB5VC36A",
    "originalUrl": "http://example.com?example=1",
    "surtUrl": "com,example)/?example=1",
    "mime": "text/html",
    "meta": "-",
    "status": 200
  },
  "payload": {
    "string": "<doctype html>\n<html>\n<head>\n  <title>Example Domain</title>\n\n  <meta charset='utf-8' />\n  <meta http-equiv='Content-type' content='text/html; charset=utf-8' />\n  <meta name='viewport' content='width=device-width, initial-scale=1' />\n  <style type='text/css'>\n    body {\n      background-color: #f0f0f2;\n      margin: 0;\n      padding: 0;\n      font-family: 'Open Sans', 'Helvetica Neue', Helvetica, Arial, sans-serif;\n      \n    ...
  }
}
```

Relatables

- The *Magic Laptop* problem
- Wish, these researchers document their code
- Cannot upgrade *X* to run *Y* because *Z* will break
- This project is the spaghetti of a dozen different technologies
- Yay! Got new machine :-) ... Wait! I need to install & configure everything again ;-(
- Too many VMs to manage

What Is Docker?

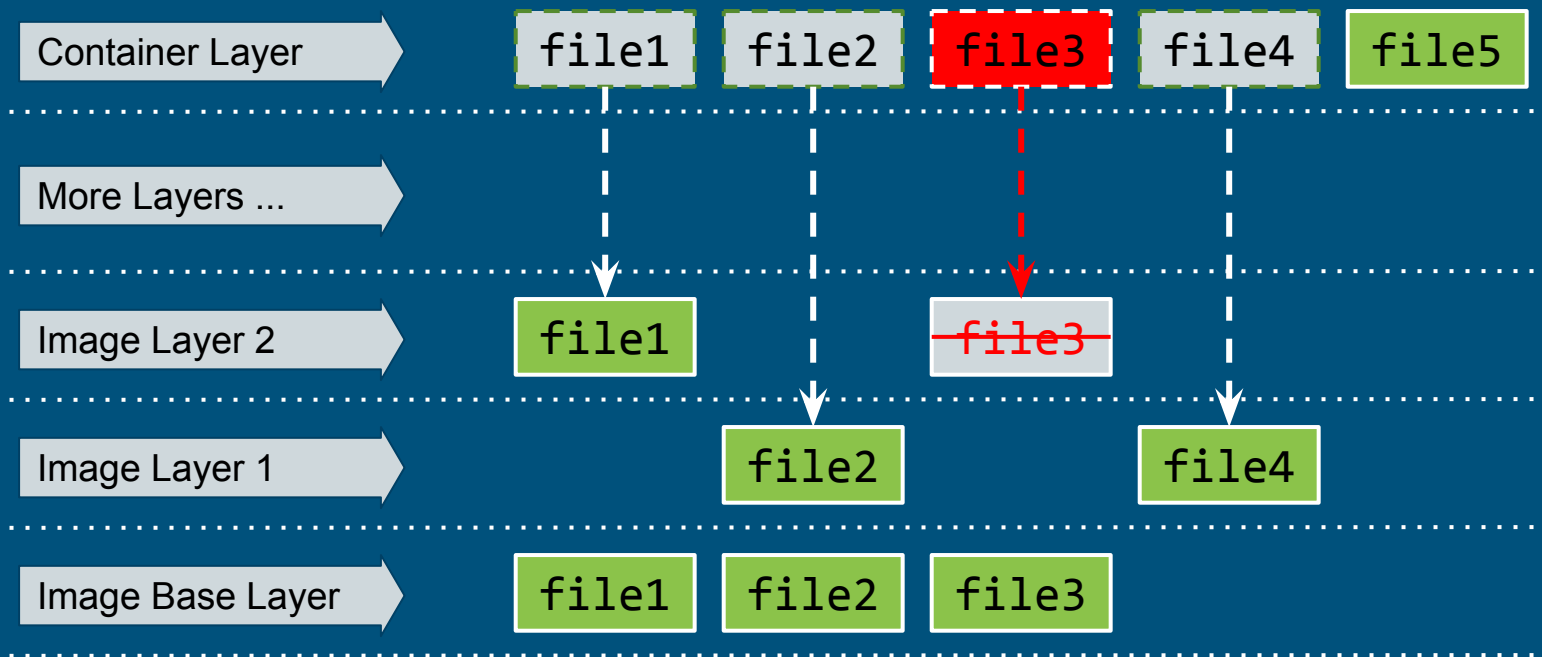
- An application container
- Packages dependencies
- Isolates applications
- Processes run on the host OS
- Uses Linux cgroups and namespaces
- Lighter than a virtual machine
- Open-source



Dockerization Process



AUFS



Link Extractor Script

linkextractor.py

```
#!/usr/bin/env python

import sys
import requests
from bs4 import BeautifulSoup

res = requests.get(sys.argv[-1])
soup = BeautifulSoup(res.text, "html.parser")
for link in soup.find_all("a"):
    print(link.get("href"))
```

- Is it executable?
- Is Python installed?
- Are necessary libraries installed?
- Can you install needed software?

Dockerized Link Extractor Script

linkextractor.py

```
#!/usr/bin/env python

import sys
import requests
from bs4 import BeautifulSoup

res = requests.get(sys.argv[-1])
soup = BeautifulSoup(res.text, "html.parser")
for link in soup.find_all("a"):
    print(link.get("href"))
```

Dockerfile

```
FROM python
LABEL maintainer="Sawood Alam <@ibnesayeed>"

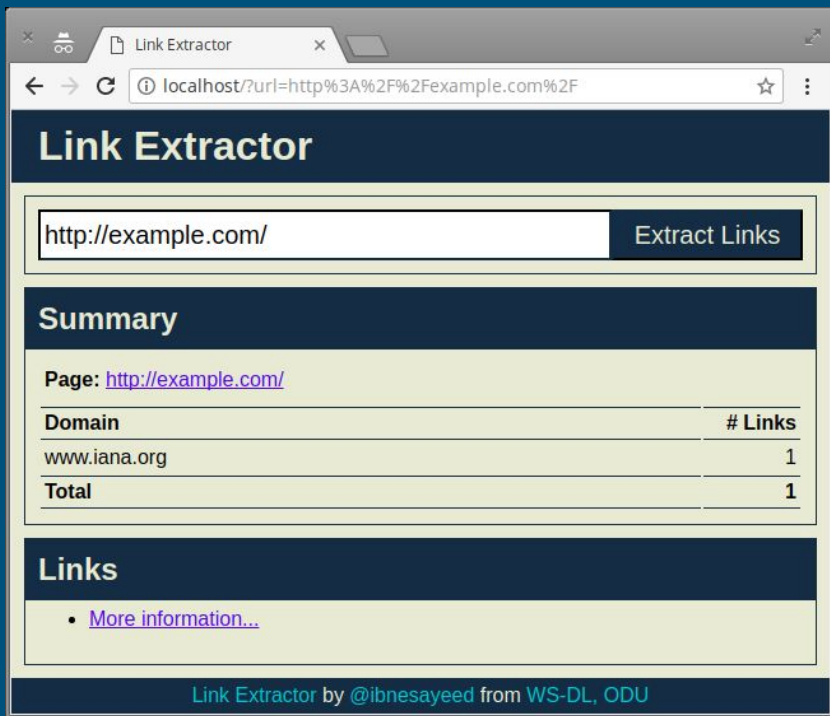
RUN pip install beautifulsoup4
RUN pip install requests
COPY linkextractor.py /app/
WORKDIR /app
RUN chmod a+x linkextractor.py

ENTRYPOINT ["./linkextractor.py"]
```


Try It

```
me@thishost$ ls
>> Dockerfile  linkextractor.py
# Build an image from the Dockerfile (change "ibnesayeed" with your Docker ID)
me@thishost$ docker image build -t ibnesayeed/linkextractor .
>> Layered docker image...
# Run a container from the locally built image
me@thishost$ docker container run ibnesayeed/linkextractor https://odu.edu/
>> Extracted links...
# Push the image to the registry
me@thishost$ docker image push ibnesayeed/linkextractor
# Log in to a different host
me@thishost$ ssh you@otherhost
# Run a container on the other host using the image in the registry
you@otherhost$ docker container run ibnesayeed/linkextractor https://example.com/
>> Pull the image from the registry (if not cached already)
>> Extracted links...
```

Orchestration Demo

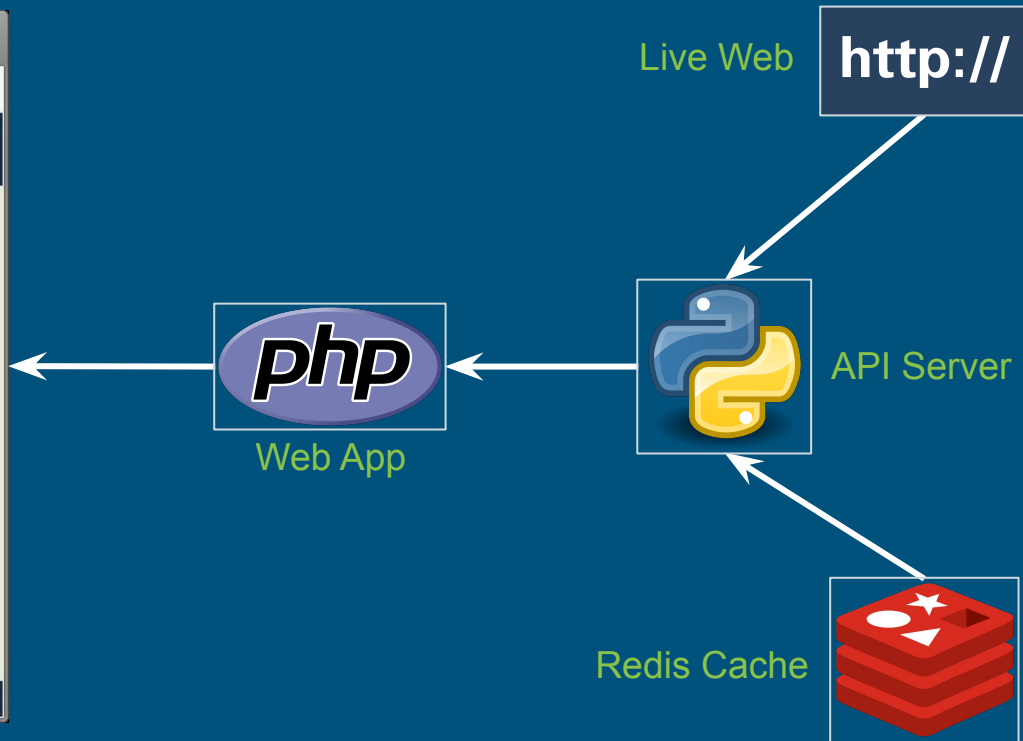


The screenshot shows a web browser window titled "Link Extractor" with the URL `localhost?url=http%3A%2F%2Fexample.com%2F`. The page content includes:

- A search input field containing `http://example.com/` and an "Extract Links" button.
- A "Summary" section with the following table:

Domain	# Links
www.iana.org	1
Total	1
- A "Links" section with a single link: [More information...](#)
- Footer text: "Link Extractor by @ibnesayeed from WS-DL, ODU"

<https://github.com/ibnesayeed/linkextractor>



Useful Links

- <https://github.com/ibnesayeed/linkextractor>
- <https://docs.docker.com/>
- <http://training.play-with-docker.com/>
- <https://training.play-with-docker.com/microservice-orchestration/>
- <https://ws-dl.blogspot.com/>
- <https://ws-dl.blogspot.com/2016/07/2016-07-21-dockerizing-archivespark.html>