

BIG DATA EUROPE

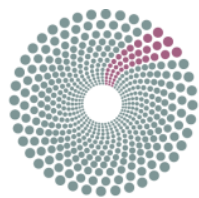
Empowering Communities
with Data Technologies



Planning BDI Stack for your Big Data Application







Ivan Ermilov @ ICTCS, Amman, Jordan








- ◎ Dockerization of Big Data Frameworks (BDF):
Why
- ◎ What is BDI Stack?
- ◎ BDI Stack Lifecycle
- ◎ BDI Stack Assembly
- ◎ Examples
 - New Spark application
 - Existing Spark application




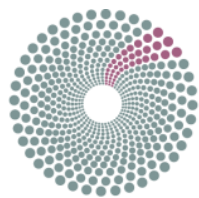
Dockerization of BDF: Why

3

	Static website	?	?	?	?	?	?	?
	Web frontend	?	?	?	?	?	?	?
	Background workers	?	?	?	?	?	?	?
	User DB	?	?	?	?	?	?	?
	Analytics DB	?	?	?	?	?	?	?
	Queue	?	?	?	?	?	?	?
		Development VM	QA Server	Single Prod Server	Onsite Cluster	Public Cloud	Contributor's laptop	Customer Servers

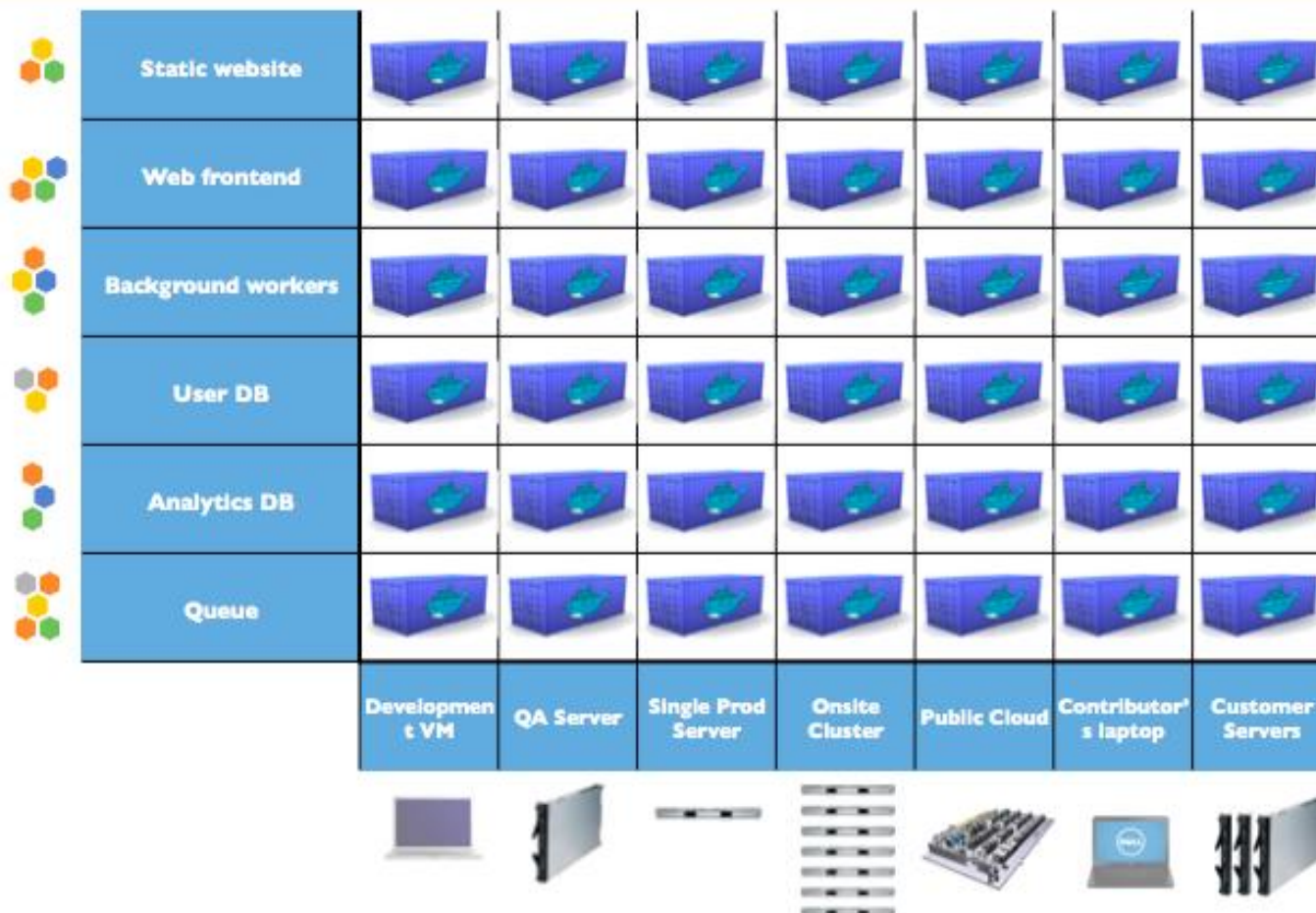






Dockerization of BDF: Why

4





Dockerization of BDF: Why

5

- ⊙ Development environment
- ⊙ Testing environment
- ⊙ Staging environment
- ⊙ Production environment

They all the same!



Less Duplication = Less Bugs



What is BDI Stack?

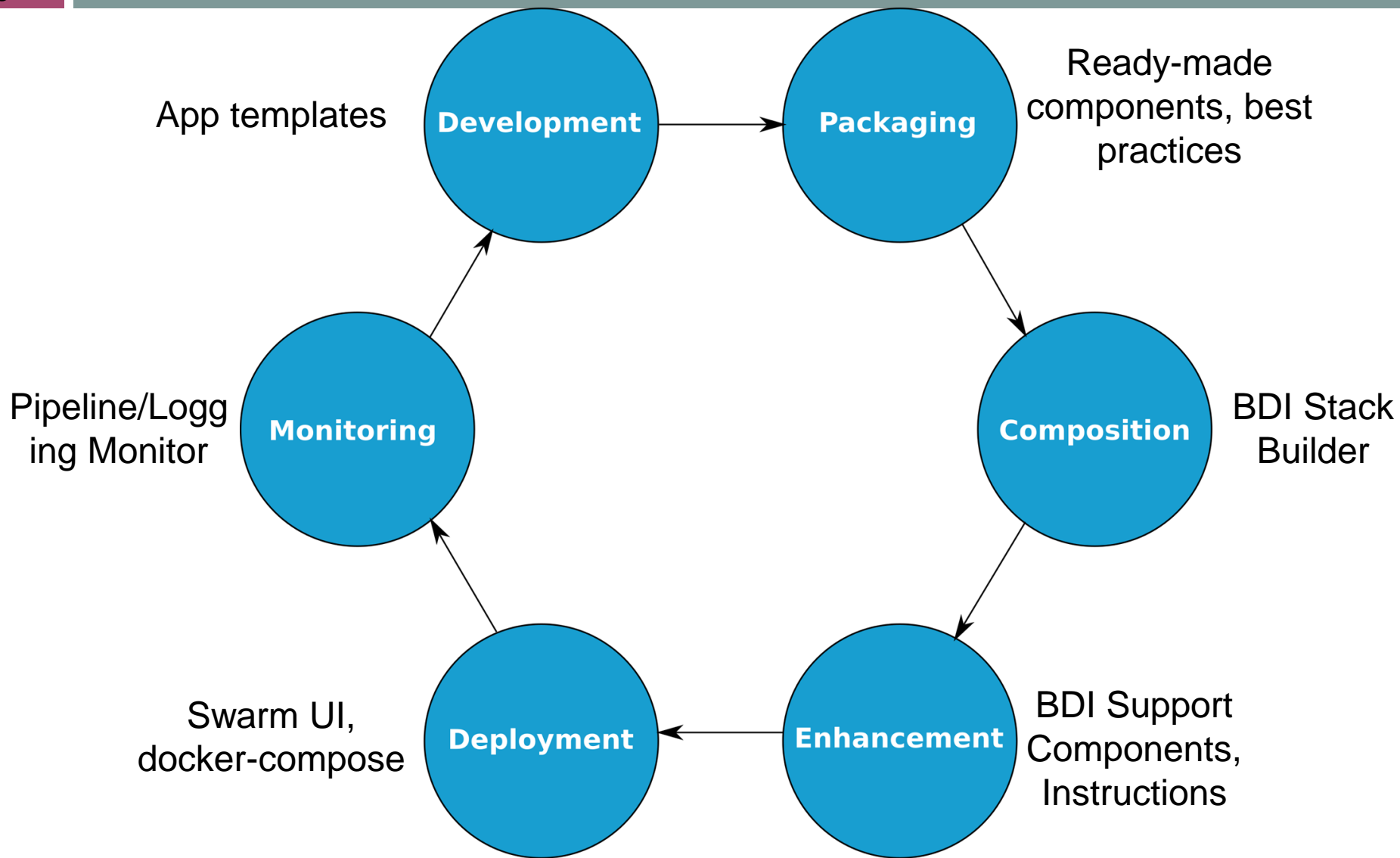
7

- ⊙ Dockerized BDF
- ⊙ In one bundle
- ⊙ With custom applications
- ⊙ `docker-compose.yml`



BDI Stack Lifecycle

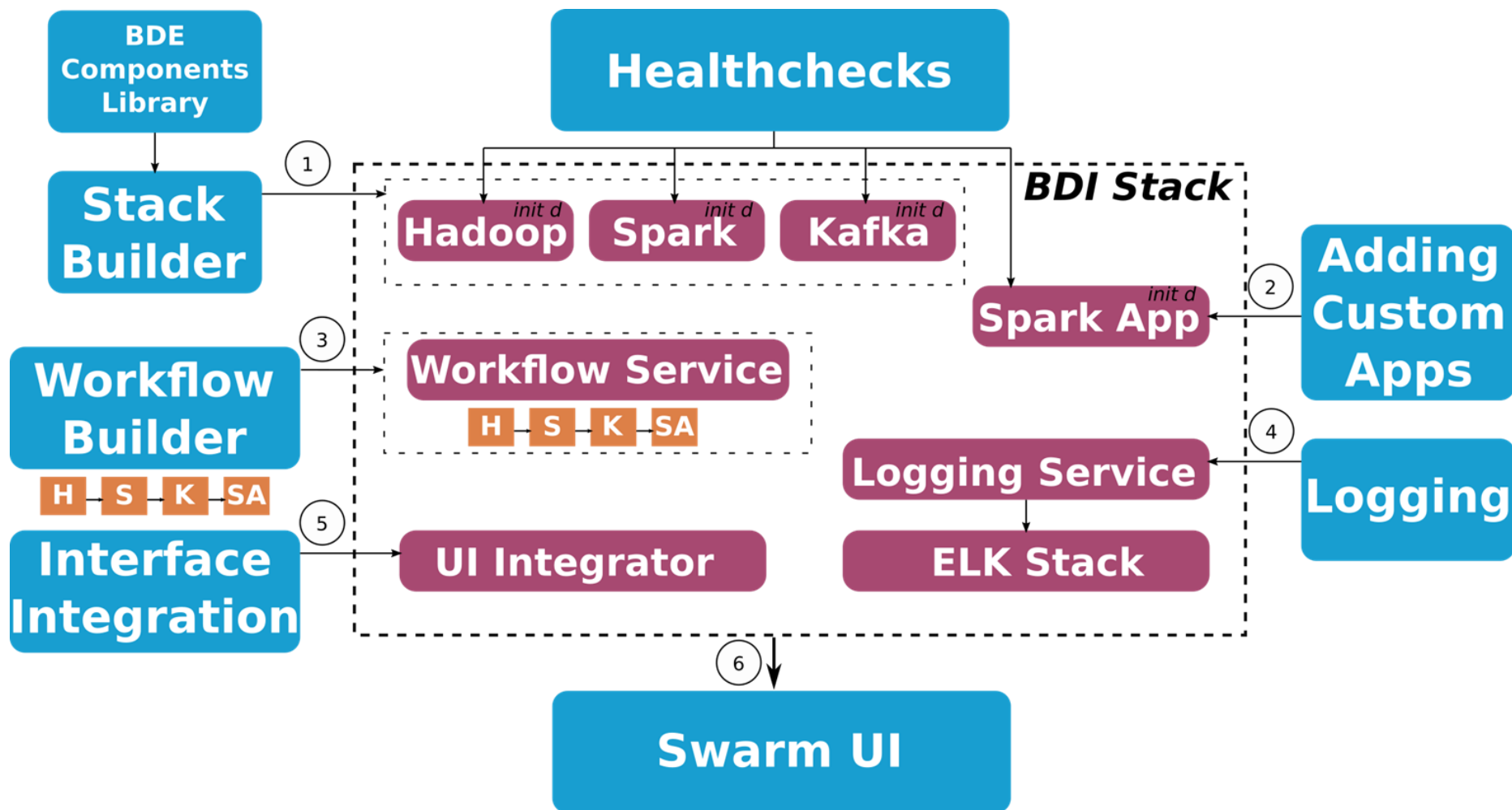
8

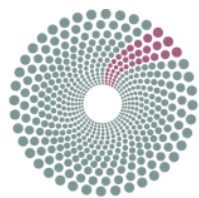




BDI Stack Assembly

9





Stack Builder

10



Stack Builder

Compact list view

Title

docker-spark

Text

version: "2"

services:

spark-master:

image: bde2020/spark-master:2.1.0-hadoop2.7

container_name: spark-master

ports:

- "8080:8080"

- "7077:7077"

environment:

- INIT_DAEMON_STEP=setup_spark

- "constraint:node==<yourmasternode>"

spark-worker-1:

image: bde2020/spark-worker:2.1.0-hadoop2.7

container_name: spark-worker-1

depends_on:

- spark-master

ports:

- "8081:8081"

environment:



Search

To add items from the list to your docker-compose file simply drag and drop them into the text field.

bde2020/4store

bde2020/4store-master

flink

2

hdfs

6





Adding Custom Apps

11

```
FROM bde2020/spark-submit:2.1.0-hadoop2.7
```

```
ENV ENABLE_INIT_DAEMON=false
```

```
ENV SPARK_APPLICATION_PYTHON_LOCATION=
```

```
ENV SPARK_MASTER_NAME=sc6-spark-master
```

```
ENV SPARK_APPLICATION_ARGS=
```

```
ENV SPARK_MASTER_URL=spark://sc6-spark-master:7077
```

```
ENV SPARK_MASTER_PORT=7077
```



WorkFlow Builder

12



BDE Workflow Builder

Workflows

My Workflow

My Workflow

Steps



Init Hadoop

Initialization of Hadoop

init-hadoop

DELETE



Init Spark

Initialization of Spark

init-spark

DELETE

step_code





Logging Monitor

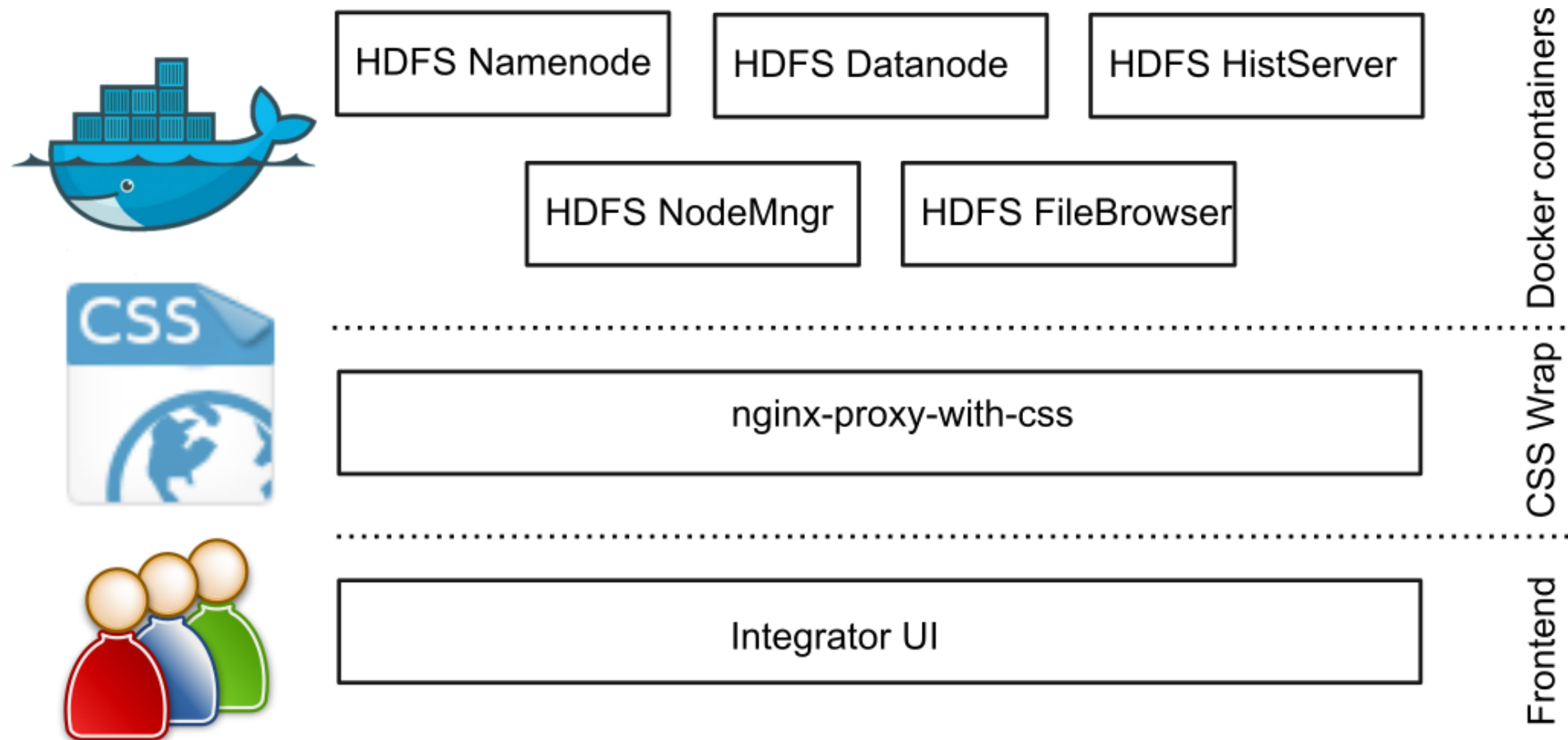
13

- ◎ Network logging for HTTP
 - Capture network interface as PCAP
 - Convert to HAR (json)
 - Expand HAR
 - Dump into ELK stack



UI Integrator Application

14





UI Integrator

15



BIG DATA EUROPE

Dashboard

Monitor

Visualization

Spark Master

Spark Worker

HDFS Namenode

Hue

Virtuoso



1.6.2

Spark Master at spark://spark-master:7077

URL: spark://spark-master:7077
REST URL: spark://spark-master:6066 (cluster mode)
Alive Workers: 1
Cores in use: 3 Total, 0 Used
Memory in use: 28.4 GB Total, 0.0 B Used
Applications: 0 Running, 0 Completed
Drivers: 0 Running, 0 Completed
Status: ALIVE

Workers

Worker Id	Address	State	Cores	Memory
worker-20160901163503-172.18.0.13-33120	172.18.0.13:33120	ALIVE	3 (0 Used)	28.4 GB (0.0 B Used)

Running Applications

Application ID	Name	Cores	Memory per Node	Submitted Time	User	State	Duration
----------------	------	-------	-----------------	----------------	------	-------	----------

Completed Applications

Application ID	Name	Cores	Memory per Node	Submitted Time	User	State	Duration
----------------	------	-------	-----------------	----------------	------	-------	----------



Reverse Proxy/CSSWrapper

16

◎ Simple injection of custom CSS

strabon:

image: bde2020/strabon

links:

- csswrapper

expose:

- "8080"

environment:

VIRTUAL_HOST: "strabon.big-data-europe.aksw.org"

VIRTUAL_PORT: "8080"

CSS_SOURCE: "strabon"

<https://www.big-data-europe.eu/using-reverse-proxy-inside-bde-platform-jwildernginx-setup-for-docker-swarm/>

<https://github.com/big-data-europe/demo-integrator-ui>



Repositories

test

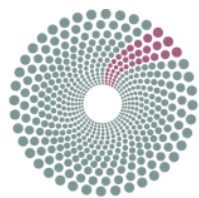
Located at <https://github.com/big-data-europe/demo-spark-sensor-data>. Has 1 connected pipelines.

EDIT

LAUNCH

Create new repository





Pipelines

test

Current status: down

UP

STOP

DOWN

RESTART



Services

pipeline

-

1

+

RESTART



nodemanager

-

1

+

RESTART



integratorui

-

1

+

RESTART



filebrowser

-

1

+

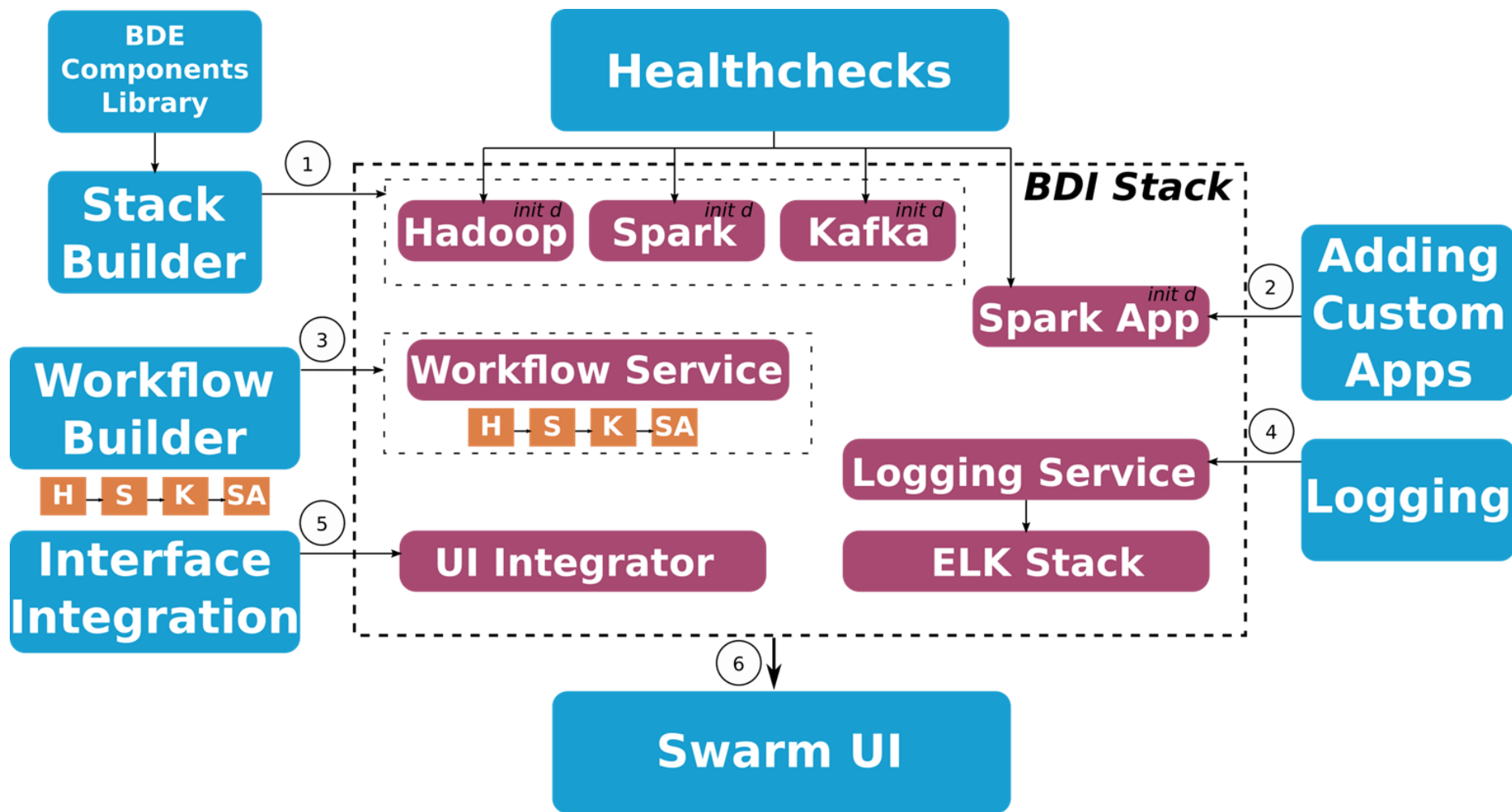
RESTART



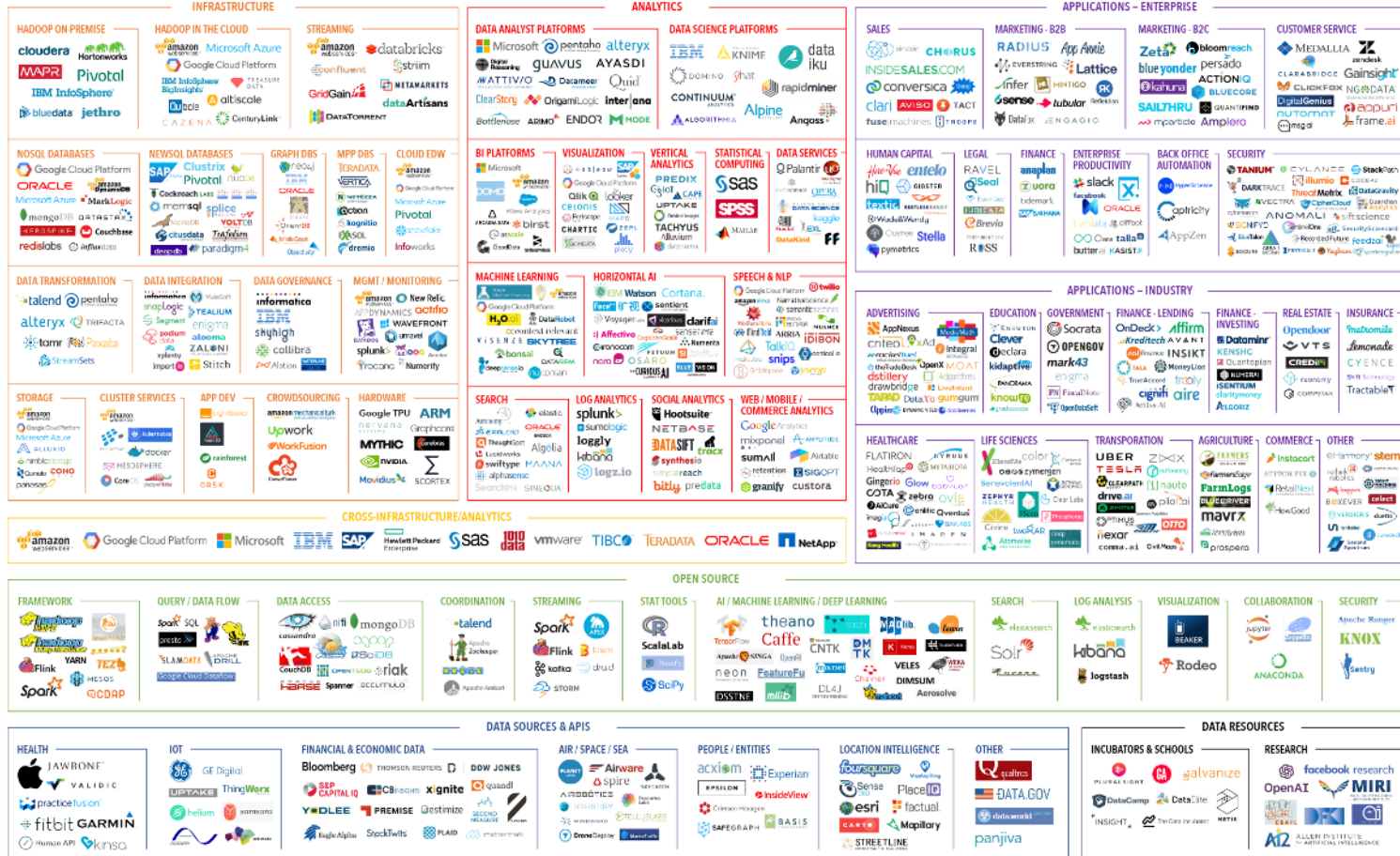


BDI Stack Assembly

19



BIG DATA LANDSCAPE 2017



Last updated 4/5/2017

© Matt Turck (@mattturck), Jim Hao (@jimrhao), & FirstMark (@firstmarkcap) mattturck.com/bigdata2017

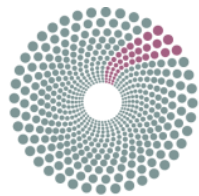
FIRSTMARK
EARLY STAGE VENTURE CAPITAL



Simplified Workflow

21

- ⊙ Outline your application requirements
- ⊙ Pick up components from BDE github repo
 - If it is not there search Docker Hub
 - Else dockerize it yourself
- ⊙ Create docker-compose.yml
- ⊙ Test with simple application
- ⊙ Develop your application on top of it
- ⊙ Proceed to enhancement step (if necessary)



Application Requirements

22

◎ Create user stories

- User wants to see the most trending recent hashtags
- User wants to see the most recent visualization of the hashtags
- User wants to see past visualizations as well



Core Functionality

23

- ◎ Fetch tweets
 - Spark streaming can do that
- ◎ Store tweets somewhere (big data)
 - HDFS
- ◎ Store trends (not too big)
 - MongoDB
- ◎ Visualize the trends
 - Kibana or custom application



Components from BDE github

24

◎ HDFS

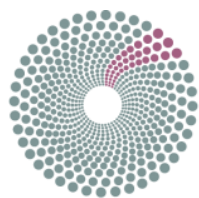
- Store data



◎ Spark

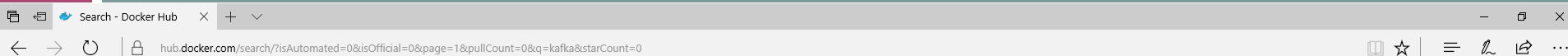
- Streaming
- Transformation
- Save/load data





Searching Docker Hub

25



Docker Store is the new place to discover public Docker content. [Check it out →](#)










[Explore](#) [Help](#) [Sign up](#) [Sign in](#)

Repositories (3097)

Windows Snap

All

	wurstmeister/kafka public automated build	417 STARS	10M+ PULLS	> DETAILS
	sheepkiller/kafka-manager public automated build	80 STARS	1M+ PULLS	> DETAILS
	ches/kafka public automated build	82 STARS	1M+ PULLS	> DETAILS
	confluentinc/cp-kafka-connect public	12 STARS	1M+ PULLS	> DETAILS
	spotify/kafka public automated build	225 STARS	500K+ PULLS	> DETAILS
	cgswong/confluent-kafka public automated build	7 STARS	1M+ PULLS	> DETAILS
	solsson/kafka	0	500K+	>



Searching Docker Hub

26

- ◎ Create a table (example for HBase)
 - Name
 - Java Version
 - Docs
 - Configurable
 - Standalone
 - Pseudodistributed
 - Distributed



Searching Docker Hub

27

- ◎ Select the best docker image from the table
 - Pay attention to the docker image license!
- ◎ Extend if necessary
- ◎ Example



Assembling docker-compose

28

- ⦿ BDE provides docker-compose.yml snippets
- ⦿ Docker images which follow the best practices does the same



Assembling docker-compose

29

◎ Copy/paste the snippets and adjust

```
version: '2'
```

```
services:
```

```
  namenode:
```

```
    image: bde2020/hadoop-namenode:1.1.0-hadoop2.8-  
java8
```

```
    container_name: namenode
```

```
    volumes:
```

```
      - ./data/namenode:/hadoop/dfs/name
```

```
    environment:
```

```
      - CLUSTER_NAME=TwitterTrendsCluster
```

```
...
```



Testing your BDI Stack

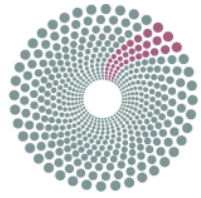
30

⊙ Manual testing

- All containers running?
- No errors in the initialization logs?

⊙ Automatic testing

- Deploy a simple application using the BDI stack along it
- Produce correct results?



Ready to develop your app!



Adding Application to the Stack

32

- ◎ Create Dockerfile
- ◎ Expose external interfaces
 - REST
 - SQL
 - SPARQL
- ◎ Upload to docker hub
 - Or your enterprise repository (e.g. gitlab)



Demo (15 mins)

33

- ◎ BDI Stack for
 - Hadoop
 - Spark
 - TwitterTrends
 - VisualizationApp



Packing existing application



Example BDI Stack: Halyard

35

- ◎ Which BD components does Halyard use?
 - HDFS
 - YARN (for MapReduce jobs)
 - HBase
- ◎ Which interfaces are supported?
 - Shell scripts (bulkload, export etc)
 - RDF4J console
 - RDF4J REST Server + Workbench



Halyard: BDI Stack

36

Hadoop

DFS

Namenode

Datanode

Resource
Manager

YARN

Node
Manager

History
Server

HBase

Master

Region
Server

Zookeeper

Zookeeper

- ⊙ Hadoop
 - DFS
 - YARN
- ⊙ HBase
- ⊙ Zookeeper



Halyard: BDI Stack (yaml)

37

namenode:

image: bde2020/hadoop-namenode:1.2.0-hadoop2.8-java8

container_name: namenode

networks:

- hbase

volumes:

- ./data/hadoop/namenode:/hadoop/dfs/name

environment:

- CLUSTER_NAME=test

ports:

- "50070:50070"

env_file:

- ./hadoop.env



Halyard: BDI Stack: Running

38

Simply execute the command:

```
docker-compose up -d
```



Halyard: BDI Stack: Simple Test

39

```
$ docker exec -it hbase /bin/bash
```

```
$ hbase shell
```

```
> list
```

```
> create 't1', 'f1'
```



Adding Halyard to BDI Stack

40

```
FROM bde2020/hadoop-base:1.2.0-hadoop2.8-java8 as hadoop-base
FROM bde2020/hbase-base:1.0.0-hbase1.2.6 as hbase-base
FROM openjdk:8
MAINTAINER Ivan Ermilov <ivan.s.ermilov@gmail.com>
ENV HADOOP_VERSION=2.8.0
COPY --from=hadoop-base /opt/hadoop-$HADOOP_VERSION /opt/hadoop-$HADOOP_VERSION
RUN ln -s /opt/hadoop-$HADOOP_VERSION/etc/hadoop /etc/hadoop
ENV PATH /opt/hadoop-$HADOOP_VERSION/bin:$PATH
ENV HBASE_VERSION=1.2.6
COPY --from=hbase-base /opt/hbase-$HBASE_VERSION /opt/hbase-$HBASE_VERSION
RUN ln -s /opt/hbase-$HBASE_VERSION/conf /etc/hbase
ENV PATH /opt/hbase-$HBASE_VERSION/bin:$PATH
ENV HALYARD_VERSION 1.2
...
```




Running Halyard SDK

41

```
$ docker run -it --rm --network hbase --  
  env-file ./hbase.env bde2020/halyard-  
  sdk:1.0.0-halyard1.2 /bin/bash  
$ ./console
```



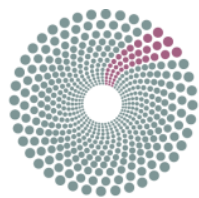
Running Halyard SDK

42

- > create hbase
- > open halyard
- > load <http://danbri.org/foaf.rdf>

```
make-run-standalone-hadoop  make-run-sdk  ivan@LatitudeE6520: ~/Workspace/Projects/BDE/docker/docker-halyard
Step 19/22 : ADD entrypoint.sh /entrypoint.sh
--> Using cache
--> 4657b083ce2a
Step 20/22 : RUN chmod a+x /entrypoint.sh
--> Using cache
--> 8679b6fa03a5
Step 21/22 : WORKDIR $HALYARD_PREFIX
--> Using cache
--> abad175e4674
Step 22/22 : ENTRYPOINT /entrypoint.sh
--> Using cache
--> 8bf29ebee8ea
Successfully built 8bf29ebee8ea
Successfully tagged bde2020/halyard-sdk:1.0.0-halyard1.2
docker run -it --rm --network hbase --env-file ./hbase.env bde2020/halyard-sdk:1.0.0-halyard1.2 /bin/bash
Configuring core
- Setting fs.defaultFS=hdfs://3144344dec0e:8020
Configuring hdfs
Configuring yarn
Configuring httpfs
Configuring kms
Configuring mapred
Configuring hbase
- Setting hbase.cluster.distributed=true
- Setting hbase.zookeeper.quorum=hbase
- Setting hbase.rootdir=hdfs://namenode:9000/hbase
root@3144344dec0e:/opt/halyard-1.2# ls
LICENSE LICENSES THIRD PARTY bulkload bulkupdate console export hiveload lib pexport presplit readme.md stats update
root@3144344dec0e:/opt/halyard-1.2# ./console
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/opt/halyard-1.2/lib/logback-classic-1.1.2.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/opt/halyard-1.2/lib/slf4j-log4j12-1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/opt/hadoop-2.8.0/share/hadoop/common/lib/slf4j-log4j12-1.7.10.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/hbase-1.2.0/lib/slf4j-log4j12-1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [ch.qos.logback.classic.util.ContextSelectorStaticBinder]
Connected to default data directory
RDF4J Console 2.1.3

2.1.3
Type 'help' for help.
> create hbase
Please specify values for the following variables:
Repository ID: halyard
Repository title: halyard
HBase Table Name: halyard
Create HBase Table if missing (true/false) [true]:
HBase Table presplit bits [0]:
Use Halyard Push Evaluation Strategy (true/false) [true]:
Query Evaluation Timeout [180]:
Repository created
> open halyard
Opened repository 'halyard'
halyard> load http://danbri.org/foaf.rdf
Loading data...
Data has been added to the repository (526 ms)
halyard> sparql
enter multi-line SPARQL query (terminate with line containing single '.')
select ?s ?p ?o where { ?s ?p ?o }
.
```



Running Halyard SDK

43

```
> sparql
select ?s ?p ?o {where ?s ?p ?o} .
```

```
halyard> sparql
enter multi-line SPARQL query (terminate with line containing single '.')
select ?s ?p ?o {where ?s ?p ?o}

Evaluating SPARQL query...
+-----+-----+-----+
| s | p | o |
+-----+-----+-----+
| :node1bltsosg9x41 | rdfs:resource | "http://perso.hirslmann.net/~ludo/foaf.rdf"@en |
| <http://www.glandscape.com/cgi-bin/prnxml.rss> | rdfs:title | "Channel of Filth"@en |
| <http://www.glandscape.com/cgi-bin/prnxml.rss> | rdfs:seeAlso | <http://www.glandscape.com/cgi-bin/prnxml.rss> |
| <http://www.glandscape.com/cgi-bin/prnxml.rss> | rdfs:type | rdfs:channel |
| <http://mnt.me.uk/foaf.rdf#mischa> | :name | "Mischa Tuffield"@en |
| <http://mnt.me.uk/foaf.rdf#mischa> | rdfs:type | :Person |
| :node1bltsosg9x44 | :name | "Margaret Hart"@en |
| :node1bltsosg9x44 | :img | <http://purl.org/net/danbri/2000/06/mh.jpg> |
| :node1bltsosg9x44 | :mbox | <mailto:mags@apocalypse.org> |
| :node1bltsosg9x44 | rdfs:seeAlso | <http://www.apocalypse.org/~mags/webwho.xrdf> |
| :node1bltsosg9x44 | rdfs:type | :Person |
| :node1bltsosg9x20 | :name | "Dean Jackson"@en |
| :node1bltsosg9x20 | :mbox | <mailto:dean@w3.org> |
| :node1bltsosg9x20 | :mbox | <mailto:dino@grorg.org> |
| :node1bltsosg9x20 | :homepage | <http://www.grorg.org/dean/> |
| :node1bltsosg9x20 | rdfs:seeAlso | <http://www.grorg.org/dean/foaf.rdf> |
| :node1bltsosg9x20 | rdfs:type | :Person |
| :node1bltsosg9x20 | :mbox_sha1sum | "6de4ff27ef927b9ba21ccc88257e41a2d7e7d293"@en |
| :node1bltsosg9x17 | :name | "Dan Brickley"@en |
| :node1bltsosg9x17 | :img | <http://www.geocities.com/danbfan/Ford3.jpg> |
| :node1bltsosg9x17 | :isPrimaryTopicOf | <http://www.geocities.com/danbfan/> |
| :node1bltsosg9x17 | rdfs:type | :Person |
| :node1bltsosg9x45 | :name | "Libby Miller"@en |
| :node1bltsosg9x45 | :mbox | <mailto:libby.miller@bristol.ac.uk> |
| :node1bltsosg9x45 | :workplaceHomepage | <http://lirt.org/> |
| :node1bltsosg9x45 | rdfs:seeAlso | <http://www.libbymiller.com/webwho.xrdf> |
| :node1bltsosg9x45 | rdfs:type | :Person |
| <http://website.lineone.net/~steve_c-t/Scientology/Pickets/10-03-2001/damien.jpg> | dc:description | "An anti-scientology protest"@en |
| <http://website.lineone.net/~steve_c-t/Scientology/Pickets/10-03-2001/damien.jpg> | rdfs:type | :Image |
| <http://website.lineone.net/~steve_c-t/Scientology/Pickets/10-03-2001/damien.jpg> | dc:thumbnail | <http://website.lineone.net/~steve_c-t/Scientology/Pickets/10-03-2001/damien.jpg> |
| <mailto:daniel.brickley@bristol.ac.uk> | dct:isReplacedBy | <mailto:danbri@danbri.org> |
| :node1bltsosg9x50 | :name | "Pastor N Plizzor"@en |
| :node1bltsosg9x50 | :made | <http://www.glandscape.com/cgi-bin/prnxml.rss> |
| :node1bltsosg9x50 | rdfs:type | :Person |
| :node1bltsosg9x19 | :mbox | <mailto:libby@asemantics.com> |
| :node1bltsosg9x19 | :mbox | <mailto:libby.miller@bristol.ac.uk> |
| :node1bltsosg9x19 | rdfs:type | :Person |
| :node1bltsosg9x15 | :accountServiceHomepage | <http://www.flickr.com/> |
| :node1bltsosg9x15 | :accountProfilePage | <http://www.flickr.com/people/danbri/> |
| :node1bltsosg9x15 | :accountName | "danbri"@en |
| :node1bltsosg9x15 | rdfs:type | :OnlineAccount |
| :node1bltsosg9x49 | :name | "Martin L Poulter"@en |
| :node1bltsosg9x49 | :mbox | <mailto:m.l.poulter@bristol.ac.uk> |
| :node1bltsosg9x49 | :knows | :node1bltsosg9x50 |
| :node1bltsosg9x49 | rdfs:type | :Person |
| :node1bltsosg9x13 | :accountServiceHomepage | <http://del.icio.us> |
| :node1bltsosg9x13 | :accountProfilePage | <http://del.icio.us/danbri/> |
| :node1bltsosg9x13 | :accountName | "danbri"@en |
| :node1bltsosg9x13 | rdfs:type | :OnlineAccount |
| :node1bltsosg9x33 | :name | "Jan Grant"@en |
| :node1bltsosg9x33 | :depiction | <http://loctl.org/jan/test/wizard.jpg> |
| :node1bltsosg9x33 | :mbox | <mailto:jan.grant@bristol.ac.uk> |
```



Halyard: BDI Stack (complete)

44

Hadoop

DFS

Namenode

Datanode

Resource
Manager

YARN

Node
Manager

History
Server

HBase

Master

Region
Server

Zookeeper

Zookeeper

BDE

UI Integrator

Workflow

Logging

Halyard

sdk

rdf4j-server

workbench



Thank you

45

Questions?

Github: <https://github.com/earthquakesan>

@AKSW: <http://aksw.org/IvanErmilov.html>

Email: iermilov@informatik.uni-leipzig.de

Twitter: @earthquakesan

LinkedIn: <https://www.linkedin.com/in/iermilov/>