



CIKLUM
EMPOWERING COLLABORATION



Microsoft



LOHIKA

Docker-compose, vagrant і інші «страшні» слова



bit.ly/msp-docker



DataArt

DEVELOPEX

DEVELOPMENT EXCELLENCE

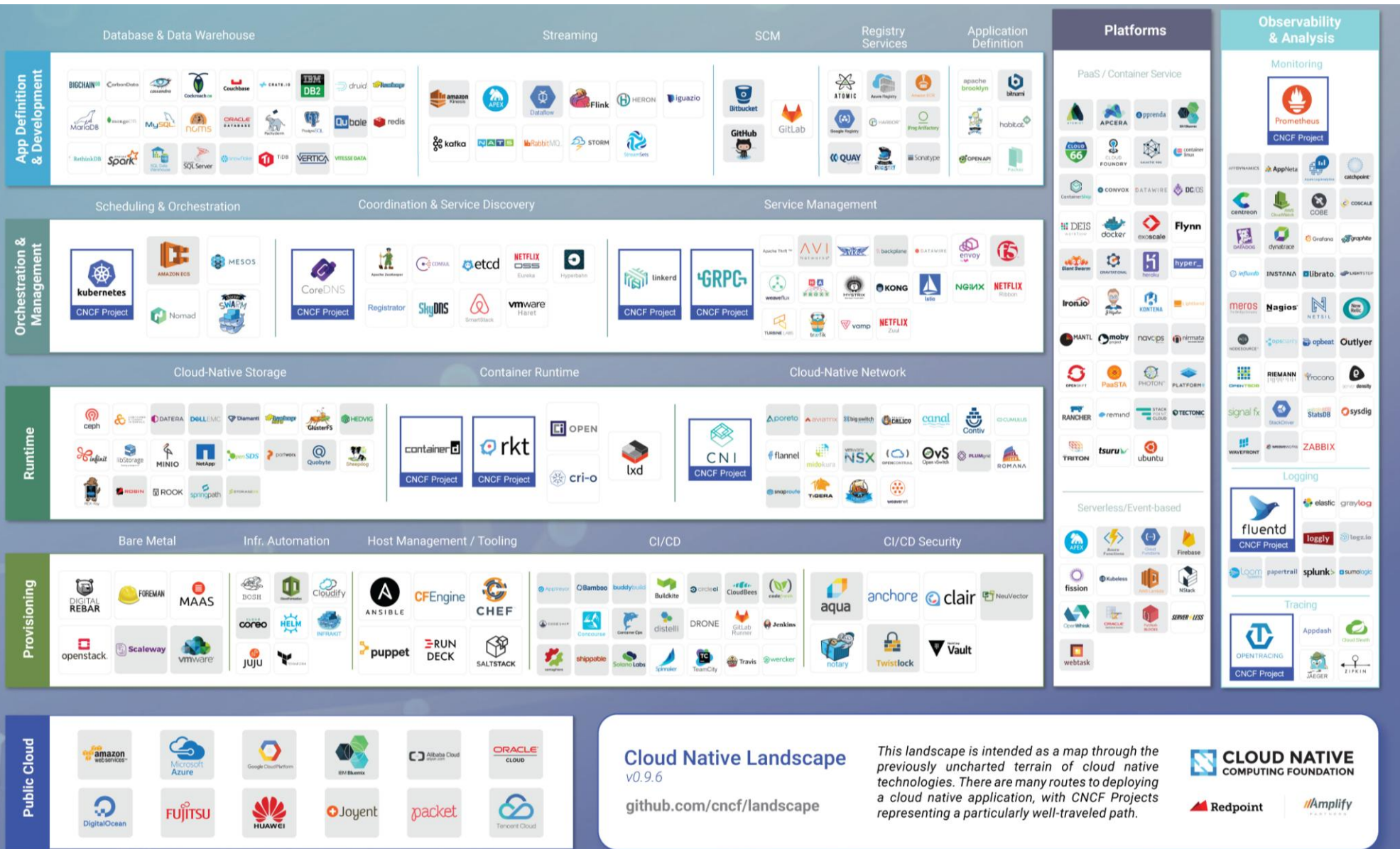


About me

- Work as DevOps Engineer at PDFfiller
- IT KPI (co-founder)
- Linux & Open Source enthusiast
- In love with JavaScript & Python
- Write on Bash and PHP



So many...



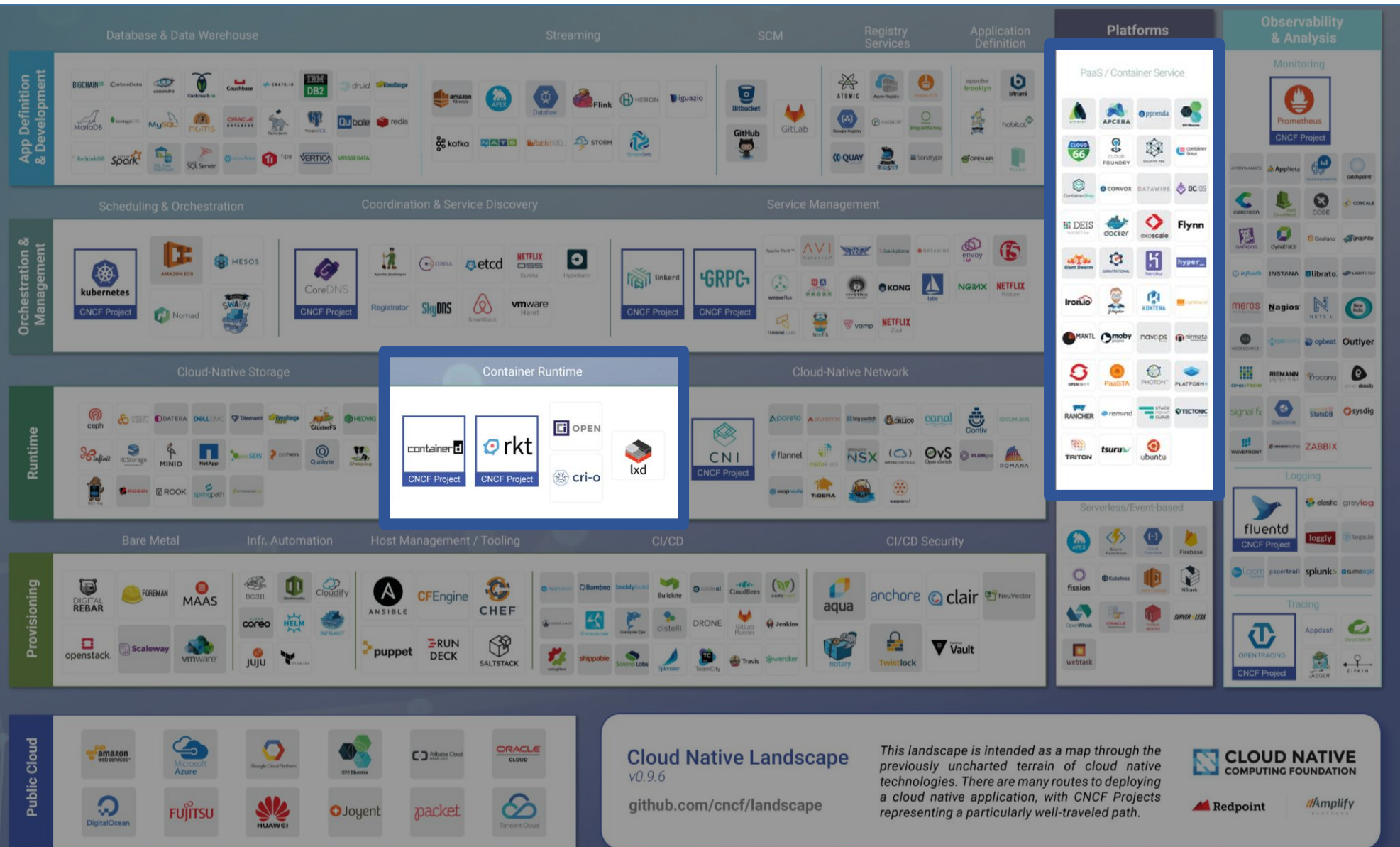
Cloud Native Landscape
v0.9.6
github.com/cncf/landscape

This landscape is intended as a map through the previously uncharted terrain of cloud native technologies. There are many routes to deploying a cloud native application, with CNCF Projects representing a particularly well-traveled path.

CLOUD NATIVE
COMPUTING FOUNDATION
Redpoint Amplify

ROAD SHOW
Microsoft Student Partners

So many...



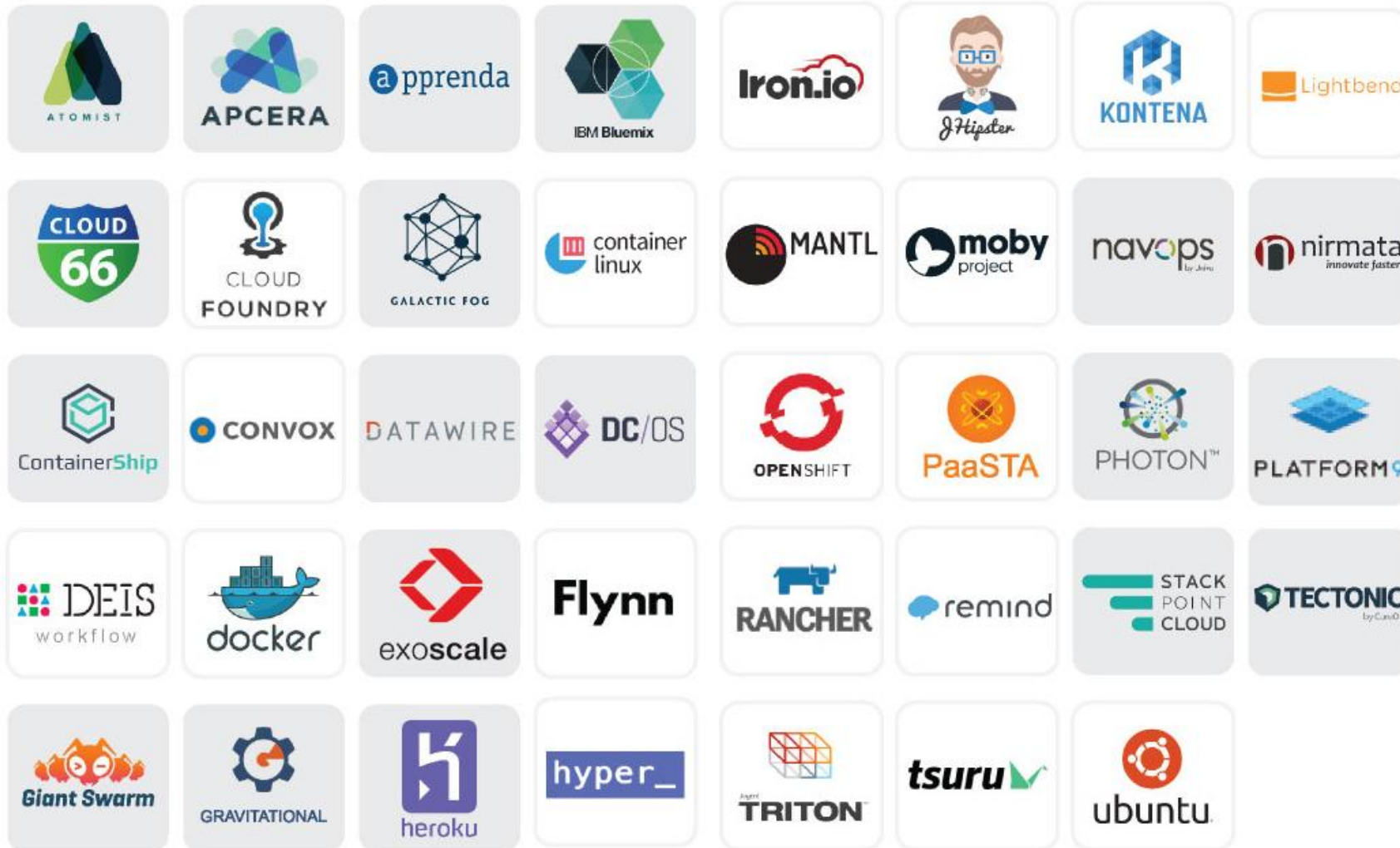
ROAD SHOW
Microsoft Student Partners

Container runtime



Container services & PaaS

PaaS / Container Service



ROAD SHOW

Microsoft Student Partners

Services, which can run local

PaaS / Container Service



Just container services, which can run local

Container Service



Docker

Dockerfile

Dockerfile

```
1  ARG VERSION=latest
2  FROM busybox:${VERSION}
3
4
5  ARG VERSION
6
7  RUN echo ${VERSION} > version
```


Dockerfile

```
1  ARG VERSION=latest
2  FROM busybox:${VERSION}
3
4  ENV VERSION previous
5  ARG VERSION
6
7  RUN echo ${VERSION} > version
```

Dockerfile

```
1  ARG VERSION=latest
2  FROM busybox:${VERSION}
3
4  ARG VERSION
5  ENV VERSION=${VERSION:-0}
6
7  RUN echo ${VERSION} > version
```

Dockerfile

```
1  ARG VERSION=latest
2  FROM busybox:${VERSION}
3
4  ARG VERSION
5  ENV VERSION=${VERSION:-0}
6
7  RUN echo ${VERSION} > version
8
9  ADD http://bit.ly/raw or binary file /app/file
10 COPY . /app
```


Dockerfile

```
1 FROM busybox:latest
2
3 WORKDIR /a
4 WORKDIR b
5 WORKDIR c
6
7 RUN pwd # Output: /a/b/c
```

Dockerfile

```
1  FROM busybox:latest
2
3  WORKDIR /a
4  WORKDIR b
5  WORKDIR c
6
7  RUN pwd # Output: /a/b/c
8
9  ENTRYPOINT ["ping"]
10 CMD ["8.8.8.8"]
```

Dockerfile

```
1 FROM busybox:latest
2
3 LABEL maintainer="Maksym Vlasov <m@il.com>"
```


Dockerfile

```
1 FROM busybox:latest
2
3 LABEL maintainer="Maksym Vlasov <m@il.com>"
4
5 EXPOSE 9000 9001
```

Dockerfile

```
1 FROM busybox:latest
2
3 LABEL maintainer="Maksym Vlasov <m@il.com>"
4
5 EXPOSE 9000 9001
6
7 VOLUME ["/data"]
```

Dockerfile

```
1 FROM busybox:latest
2
3 LABEL maintainer="Maksym Vlasov <m@il.com>"
4
5 EXPOSE 9000 9001
6
7 VOLUME ["/data"]
8
9 HEALTHCHECK --interval=30s \
10             --timeout=30s \
11             --start-period=0s \
12             --retries=3 \
13             CMD wget http://localhost/ || exit 1
```


Dockerfile

```
1 FROM busybox:latest
2
3 SHELL ["/bin/sh", "-c"]
```

Dockerfile

```
1 FROM busybox:latest
2
3 SHELL ["/bin/sh", "-c"]
4
5 RUN addgroup group && \
6     adduser -D -H user -G group
7
8 USER user:group
```



Microsoft Student Partners



Dockerfile

```
1 FROM busybox:latest
2
3 SHELL ["/bin/sh", "-c"]
4
5 RUN addgroup group && \
6     adduser -D -H user -G group
7
8 USER user:group
9
10 STOPSIGNAL 9 # SIGKILL
```



Microsoft Student Partners

Dockerfile

```
1  FROM busybox:latest
2
3  SHELL ["/bin/sh", "-c"]
4
5  RUN addgroup group && \
6      adduser -D -H user -G group
7
8  USER user:group
9
10 STOPSIGNAL 9 # SIGKILL
11
12 ONBUILD ADD . /app/src
13 ONBUILD RUN echo "based on busybox"
```


Docker CLI

Docker CLI

`docker login`

Docker CLI

```
docker login
```

```
docker run -ti --name name busybox
```

```
docker exec -ti name sh
```

Docker CLI

`docker login`

`docker run -ti --name name busybox`

`docker exec -ti name sh`

`docker attach name`

`docker rename name container`

Docker CLI

`docker pause container`

`docker unpause container`

Docker CLI

`docker pause container`

`docker unpause container`

`docker inspect container`

`docker top container`

`docker stats container`

Docker CLI

`docker pause container`

`docker unpause container`

`docker inspect container`

`docker top container`

`docker stats container`

`docker system prune --volumes`

Docker-compose

Docker-compose.yml

```
1  version: '3'
```

Docker-compose.yml

```
1  version: '3'
2
3  services:
```

Docker-compose.yml

```
1  version: '3'  
2  
3  services:  
4    consul:
```

Docker-compose.yml

```
1  version: '3'
2
3  services:
4    consul:
5      build: ../dev_docker_compose/consul
```

Docker-compose.yml

```
1  version: '3'
2
3  services:
4    consul:
5      build: ../dev_docker_compose/consul
6      network_mode: host
```


Docker-compose.yml

```
1  version: '3'
2
3  services:
4    consul:
5      build: ../dev_docker_compose/consul
6      network_mode: host
7      restart: always
```

Docker-compose.yml

```
1  version: '3'
2
3  services:
4    consul:
5      build: ../dev_docker_compose/consul
6      network_mode: host
7      restart: always
8      volumes:
9        - consuldb:/consul/data
10
11  volumes:
12    consuldb:
13      external: true
```

Docker-compose CLI

Docker-compose CLI

```
docker-compose ps
```

Docker-compose CLI

`docker-compose ps`

`docker-compose config` # validation & view

Docker-compose CLI

`docker-compose ps`

`docker-compose config` # validation & view

`docker-compose pause`

`docker-compose unpause`

Docker-compose CLI

`docker-compose run service ls`

`docker-compose exec consul ls`

Docker-compose CLI

`docker-compose run service ls`

`docker-compose exec consul ls`

`docker-compose top`

Docker-compose CLI

`docker-compose run service ls`

`docker-compose exec consul ls`

`docker-compose top`

`docker-compose images`

Docker-compose CLI

`docker-compose run service ls`

`docker-compose exec consul ls`

`docker-compose top`

`docker-compose images`

`docker-compose down # prune`

Vagga

vagga.yaml

1 containers:

vagga.yaml

```
1 containers:  
2   ubuntu-consul:
```

vagga.yaml

```
1  containers:
2    ubuntu-consul:
3      setup:
4        - !Ubuntu xenial
5        - !Install [unzip, wget, ca-certificates]
```

vagga.yaml

```
1  containers:
2    ubuntu-consul:
3      setup:
4        - !Ubuntu xenial
5        - !Install [unzip, wget, ca-certificates]
6        - !Sh |
7            cd /tmp
8            wget https://releases.hashicorp.com/consul/0.8.3/consul\_0.8.3\_
9            unzip consul_0.8.3_linux_amd64.zip
10           cp consul /usr/bin/consul
```



Microsoft Student Partners

vagga.yaml

```
1  containers:
2    ubuntu-consul:
3      setup:
4        - !Ubuntu xenial
5        - !Install [unzip, wget, ca-certificates]
6        - !Sh |
7            cd /tmp
8            wget https://releases.hashicorp.com/consul/0.8.3/consul\_0.8.3\_
9            unzip consul_0.8.3_linux_amd64.zip
10           cp consul /usr/bin/consul
11  commands:
12    consul-server: !Command
```



Microsoft Student Partners

vagga.yaml

```
1  containers:
2    ubuntu-consul:
3      setup:
4        - !Ubuntu xenial
5        - !Install [unzip, wget, ca-certificates]
6        - !Sh |
7            cd /tmp
8            wget https://releases.hashicorp.com/consul/0.8.3/consul\_0.8.3\_
9            unzip consul_0.8.3_linux_amd64.zip
10           cp consul /usr/bin/consul
11  commands:
12    consul-server: !Command
13      description: Start consul in server mode
14      container: ubuntu-consul
```



Microsoft Student Partners

vagga.yaml

```
1  containers:
2    ubuntu-consul:
3      setup:
4        - !Ubuntu xenial
5        - !Install [unzip, wget, ca-certificates]
6        - !Sh |
7            cd /tmp
8            wget https://releases.hashicorp.com/consul/0.8.3/consul\_0.8.3\_
9            unzip consul_0.8.3_linux_amd64.zip
10           cp consul /usr/bin/consul
11  commands:
12    consul-server: !Command
13      description: Start consul in server mode
14      container: ubuntu-consul
15      run: |
16          /usr/bin/consul agent -server -bootstrap-expect=1 \
17              -data-dir=/tmp/consul -log-level=debug \
18              -advertise=127.0.0.1
```

Dev-docker- compose

.generate.yaml

.generate.yaml

```
1  version: '3'
2
3  services:
```


.generate.yaml

```
1  version: '3'
```

```
2
```

```
3  services:
```

```
10  volumes:
```

```
11    consuldb:
```

```
12      external: true
```



Microsoft Student Partners



.generate.yaml

```
1  version: '3'
2
3  services:
4      # Key/Value storage with config for our app
5      consul: !include:consul.yaml |
6
7      # Search engine and document-based DB.
8      elasticsearch: !include:elasticsearch.yaml |
9
10 volumes:
11     consuldb:
12         external: true
```

Dev-... vs docker-compose

```
1  version: '3'
2
3  services:
4    # Add Key/Value to Consul.
5    add_keys_to_consul: !include:add_keys_to_consul.yaml |
6
7    # Key/Value storage with config for our app and other infrastructure.
8    consul: !include:consul.yaml |
9
10   # Search engine and document-based DB.
11   elasticsearch: !include:elasticsearch.yaml |
12
13   # Add logs from filesystem to Elasticsearch.
14   filebeat: !include:filebeat.yaml |
15
16   # Visualization tool for data in Elasticsearch.
17   kibana: !include:kibana.yaml |
18
19   # Queue with logs which should write to DB.
20   rabbit: !include:rabbit.yaml |
21
22   # Consul services registrator.
23   registrator: !include:registrator.yaml |
24
25
26  volumes:
27    rabbitdb:
28      external: true
29    consuldb:
30      external: true
31
32  networks:
33    default:
34      external:
35        name: dev-compose-network
```

Dev-... vs docker-compose

```
1 version: '3'
2
3 services:
4   # Add Key/Value to Consul.
5   add_keys_to_consul: !include:add_keys_to_consul.yaml |
6
7   # Key/Value storage with config for our app and other infrastructure.
8   consul: !include:consul.yaml |
9
10  # Search engine and document-based DB.
11  elasticsearch: !include:elasticsearch.yaml |
12
13  # Add logs from filesystem to Elasticsearch.
14  filebeat: !include:filebeat.yaml |
15
16  # Visualization tool for data in Elasticsearch.
17  kibana: !include:kibana.yaml |
18
19  # Queue with logs which should write to DB.
20  rabbit: !include:rabbit.yaml |
21
22  # Consul services registrator.
23  registrator: !include:registrator.yaml |
24
25
26 volumes:
27   rabbitdb:
28     external: true
29   consuldb:
30     external: true
31
32 networks:
33   default:
34     external:
35     name: dev-compose-network
```

```
1 networks:
2   default:
3     external:
4       name: dev-compose-network
5 services:
6   add_keys_to_consul:
7     build: ../dev_docker_compose/yaml_files/add_keys_to_consul
8     depends_on:
9       - consul
10    environment:
11      CONSUL_SERVICE_NAME: dev-app
12      REGION: us-east-1
13    network_mode: host
14    restart: on-failure
15    volumes:
16      - ../consul_conf:/consul_conf/
17  consul:
18    build: ../dev_docker_compose/yaml_files/consul
19    network_mode: host
20    restart: always
21    volumes:
22      - consuldb:/consul/data
23  elasticsearch:
24    deploy:
25      resources:
26        limits:
27          memory: 1g
28    environment:
29      - cluster.name=EScluster
30      - bootstrap.memory_lock=true
31      - ES_JAVA_OPTS=-Xms512m -Xmx512m
32      - xpack.security.enabled=false
33      - node.name=es_node1
34    image: docker.elastic.co/elasticsearch/elasticsearch:5.4.0
35    network_mode: host
36    ports:
37      - 9200:9200
38    restart: always
39    ulimits:
40      memlock:
41        hard: -1
42        soft: -1
43  filebeat:
44    build: ../dev_docker_compose/yaml_files/filebeat
45    depends_on:
46      - elasticsearch
47    environment:
48      CONSUL_SERVICE_NAME: dev-app
49    network_mode: host
50    restart: always
51    volumes:
52      - ../elastic_logs:/elastic_logs/
53  kibana:
54    environment:
55      ELASTICSEARCH_URL: http://localhost:9200
56      SERVER_NAME: http://localhost:5601
57      XPACK_MONITORING_UI_CONTAINER_ELASTICSEARCH_ENABLED: 'false'
58    image: docker.elastic.co/kibana/kibana:5.4.0
59    network_mode: host
60    restart: always
61  rabbit:
62    domainname: rabbit.local
63    hostname: rabbitmq
64    image: pdffiller/rabbitmq:latest
65    network_mode: host
66    ports:
67      - 4369:4369
68      - 5671:5671
69      - 5672:5672
70      - 15671:15671
71      - 15672:15672
72      - 25672:25672
73    restart: always
74    volumes:
75      - rabbitdb:/var/lib/rabbitmq
76  registrator:
77    command: consul://127.0.0.1:8500
78    depends_on:
79      - consul
80    image: gliderlabs/registrator:latest
81    network_mode: host
82    restart: always
83    volumes:
84      - /var/run/docker.sock:/tmp/docker.sock
85  version: '3'
86  volumes:
87    consuldb:
88      external: true
89    rabbitdb:
90      external: true
```



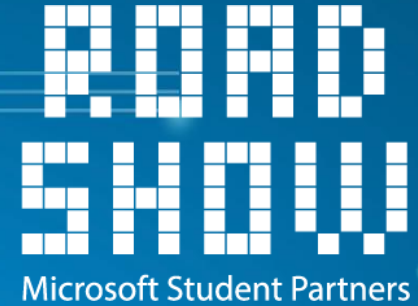
Microsoft Student Partners

Dev-... vs docker-compose

```
1 version: '3'
2
3 services:
4   # Add Key/Value to Consul.
5   add_keys_to_consul: !include:add_keys_to_consul.yaml |
6
7   # Key/Value storage with config for our app and other
8   consul: !include:consul.yaml |
9
10  # Search engine and document-based DB.
11  elasticsearch: !include:elasticsearch.yaml |
12
13  # Add logs from filesystem to Elasticsearch.
14  filebeat: !include:filebeat.yaml |
15
16  # Visualization tool for data in Elasticsearch.
17  kibana: !include:kibana.yaml |
18
19  # Queue with logs which should write to DB.
20  rabbit: !include:rabbit.yaml |
21
22  # Consul services registrator.
23  registrator: !include:registrator.yaml |
24
25
26 volumes:
27   rabbitdb:
28     external: true
29   consuldb:
30     external: true
31
32 networks:
33   default:
34     external:
35     name: dev-compose-network
```

```
1 version: '3'
2
3 services:
4   # Add Key/Value to Consul.
5   add_keys_to_consul: !include:add_keys_to_consul.yaml |
6
7   # Key/Value storage with config for our app and other infrastructure.
8   consul: !include:consul.yaml |
9
10  # Search engine and document-based DB.
11  elasticsearch: !include:elasticsearch.yaml |
12
13  # Add logs from filesystem to Elasticsearch.
14  filebeat: !include:filebeat.yaml |
15
16  # Visualization tool for data in Elasticsearch.
17  kibana: !include:kibana.yaml |
18
19  # Queue with logs which should write to DB.
20  rabbit: !include:rabbit.yaml |
21
22  # Consul services registrator.
23  registrator: !include:registrator.yaml |
24
25
26 volumes:
27   rabbitdb:
28     external: true
29   consuldb:
30     external: true
31
32 networks:
33   default:
34     external:
35     name: dev-compose-network
```

```
1 networks:
2   default:
3     external:
4     name: dev-compose-network
5
6 services:
7   add_keys_to_consul:
8     build: ../dev_docker_compose/yaml_files/add_keys_to_consul
9     depends_on:
10       - consul
11     environment:
12       CONSUL_SERVICE_NAME: dev-app
13       REGION: us-east-1
14     network_mode: host
15     restart: on-failure
16     volumes:
17       - ../consul_conf:/consul_conf/
18   consul:
19     build: ../dev_docker_compose/yaml_files/consul
20     network_mode: host
21     restart: always
22     volumes:
23       - consuldb:/consul/data
24   elasticsearch:
25     deploy:
26       resources:
27         limits:
28           memory: 1g
29     environment:
30       - cluster.name=EScluster
31       - bootstrap.memory_lock=true
32       - ES_JAVA_OPTS=-Xms512m -Xmx512m
33       - xpack.security.enabled=false
34       - node.name=es_node1
35     image: docker.elastic.co/elasticsearch/elasticsearch:5.4.0
36     network_mode: host
37     ports:
38       - 9200:9200
39     restart: always
40     ulimits:
41       memlock:
42         hard: -1
43         soft: -1
44   filebeat:
45     build: ../dev_docker_compose/yaml_files/filebeat
46     depends_on:
47       - elasticsearch
48     environment:
49       CONSUL_SERVICE_NAME: dev-app
50     network_mode: host
51     restart: always
52     volumes:
53       - ../elastic_logs:/elastic_logs/
54   kibana:
55     environment:
56       ELASTICSEARCH_URL: http://localhost:9200
57       SERVER_NAME: http://localhost:5601
58       XPACK_MONITORING_UI_CONTAINER_ELASTICSEARCH_ENABLED: 'false'
59     image: docker.elastic.co/kibana/kibana:5.4.0
60     network_mode: host
61     restart: always
62   rabbit:
63     domainname: rabbit.local
64     hostname: rabbitmq
65     image: pdffiller/rabbitmq:latest
66     network_mode: host
67     ports:
68       - 4369:4369
69       - 5671:5671
70       - 5672:5672
71       - 15671:15671
72       - 15672:15672
73       - 25672:25672
74     restart: always
75     volumes:
76       - rabbitdb:/var/lib/rabbitmq
77   registrator:
78     command: consul://127.0.0.1:8500
79     depends_on:
80       - consul
81     image: gliderlabs/registrator:latest
82     network_mode: host
83     restart: always
84     volumes:
85       - /var/run/docker.sock:/tmp/docker.sock
86   version: '3'
87 volumes:
88   consuldb:
89     external: true
90   rabbitdb:
91     external: true
```



Structure

```
1 dev-docker-compose
2 └─ .generate.yaml
```


Structure

```
1 dev-docker-compose
2 |— .generate.yaml
3 |— Makefile
4 |— Makefile.dev.mk
```

Structure

```
1 dev-docker-compose
2 |— .generate.yaml
3 |— Makefile
4 |— Makefile.dev.mk
5 |— .dev_docker_compose
6 |   |— Dockerfile
7 |   |— loader.py
8 |   |— RELEASE_NOTES.md
```

Structure

```
1 dev-docker-compose
2 |— .generate.yaml
3 |— Makefile
4 |— Makefile.dev.mk
5 |— .dev_docker_compose
6 |   |— Dockerfile
7 |   |— loader.py
8 |   |— RELEASE_NOTES.md
9 |   |— yaml_files
10 |       |— consul.yaml
11 |       |— elasticsearch.yaml
```



Microsoft Student Partners



Structure

```
1  dev-docker-compose
2  |— .generate.yaml
3  |— Makefile
4  |— Makefile.dev.mk
5  |— .dev_docker_compose
6  |   |— Dockerfile
7  |   |— loader.py
8  |   |— RELEASE_NOTES.md
9  |   |— yaml_files
10 |       |— consul.yaml
11 |       |— elasticsearch.yaml
12 |— .consul_conf
13 |   |— dev_conf.json
14 |   |— devops_conf.json
```

Structure

```
1  dev-docker-compose
2  |— .generate.yaml
3  |— Makefile
4  |— Makefile.dev.mk
5  |— .dev_docker_compose
6  |   |— Dockerfile
7  |   |— loader.py
8  |   |— RELEASE_NOTES.md
9  |   |— yaml_files
10 |       |— consul.yaml
11 |       |— elasticsearch.yaml
12 |— .consul_conf
13 |   |— dev_conf.json
14 |   |— devops_conf.json
15 |— .elastic_logs
16 |   |— logs.json
```

Structure

```
1  dev-docker-compose
2  |— .generate.yaml
3  |— Makefile
4  |— Makefile.dev.mk
5  |— .dev_docker_compose
6  |   |— Dockerfile
7  |   |— loader.py
8  |   |— RELEASE_NOTES.md
9  |   └─ yaml_files
10 |       |— consul.yaml
11 |       └─ elasticsearch.yaml
12 |— .consul_conf
13 |   |— dev_conf.json
14 |   └─ devops_conf.json
15 |— .elastic_logs
16 |   └─ logs.json
17 └─ README.md
```




CIKLUM
EMPOWERING COLLABORATION



Microsoft



LOHIKA

Questions?

fb.com/maxymvlasov
@maxymvlasov



bit.ly/msp-docker



DataArt

DEVELOPEX

DEVELOPMENT EXCELLENCE

