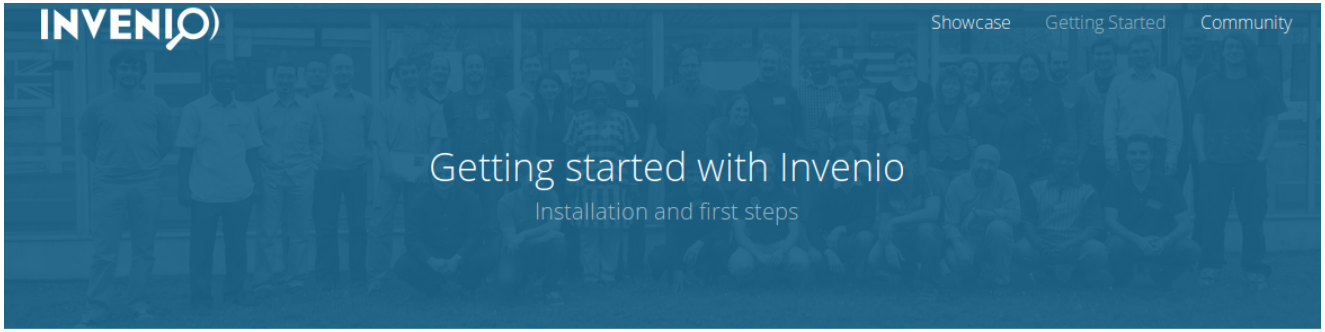
Practical exercises: Installing Invenio

Tibor Šimko

@tiborsimko

CERN-UNESCO School on Digital Libraries, Kumasi, Ghana · November 2016

Installing Invenio



Installation

Invenio 1.2

Install latest stable release in a virtual machine using vagrant:

```
$ git clone git@github.com:inveniosoftware/invenio -b maint-1.2
$ cd invenio
$ vagrant up && vagrant ssh web
web> source .inveniorc
web> /vagrant/scripts/create-instance.sh
web> /vagrant/scripts/populate-instance.sh
$ firefox http://192.168.50.10/record/1
```

Invenio 3.0

Install latest developer preview using docker:

```
$ git clone git@github.com:inveniosoftware/invenio
$ cd invenio
$ docker-compose build
$ docker-compose up -d
$ docker-compose run --rm web ./scripts/populate-instance.sh
$ firefox http://127.0.0.1/
$ firefox http://127.0.0.1/records/1
```

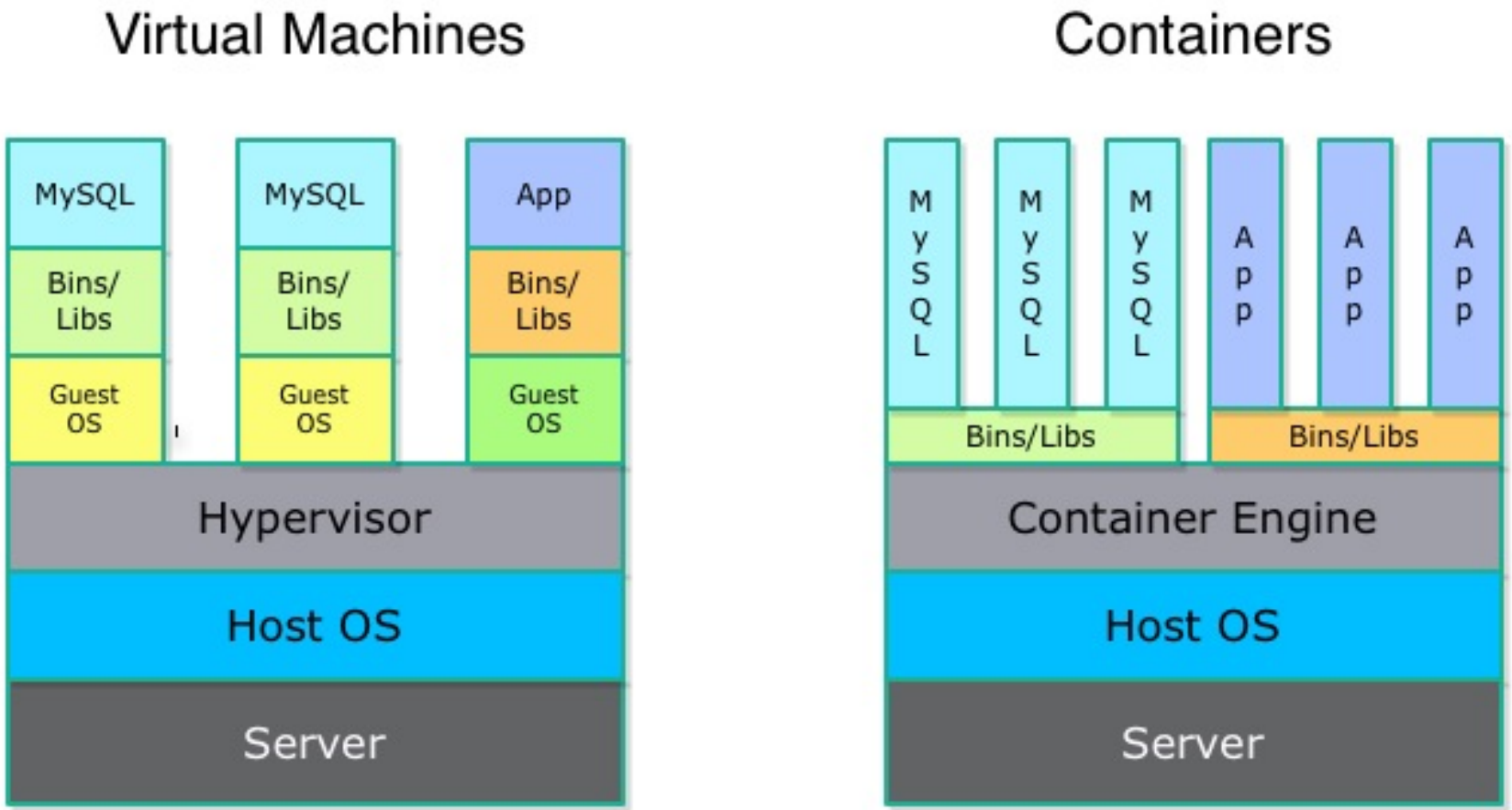
`http://inveniosoftware.org/gettingstarted`

Installing Invenio on your laptop

1. via scripts
   - “manual” way, good for learning
2. via Virtual Machines
   - “clean” way, emulates production hardware
3. via Docker
   - “modern” way, uses containers

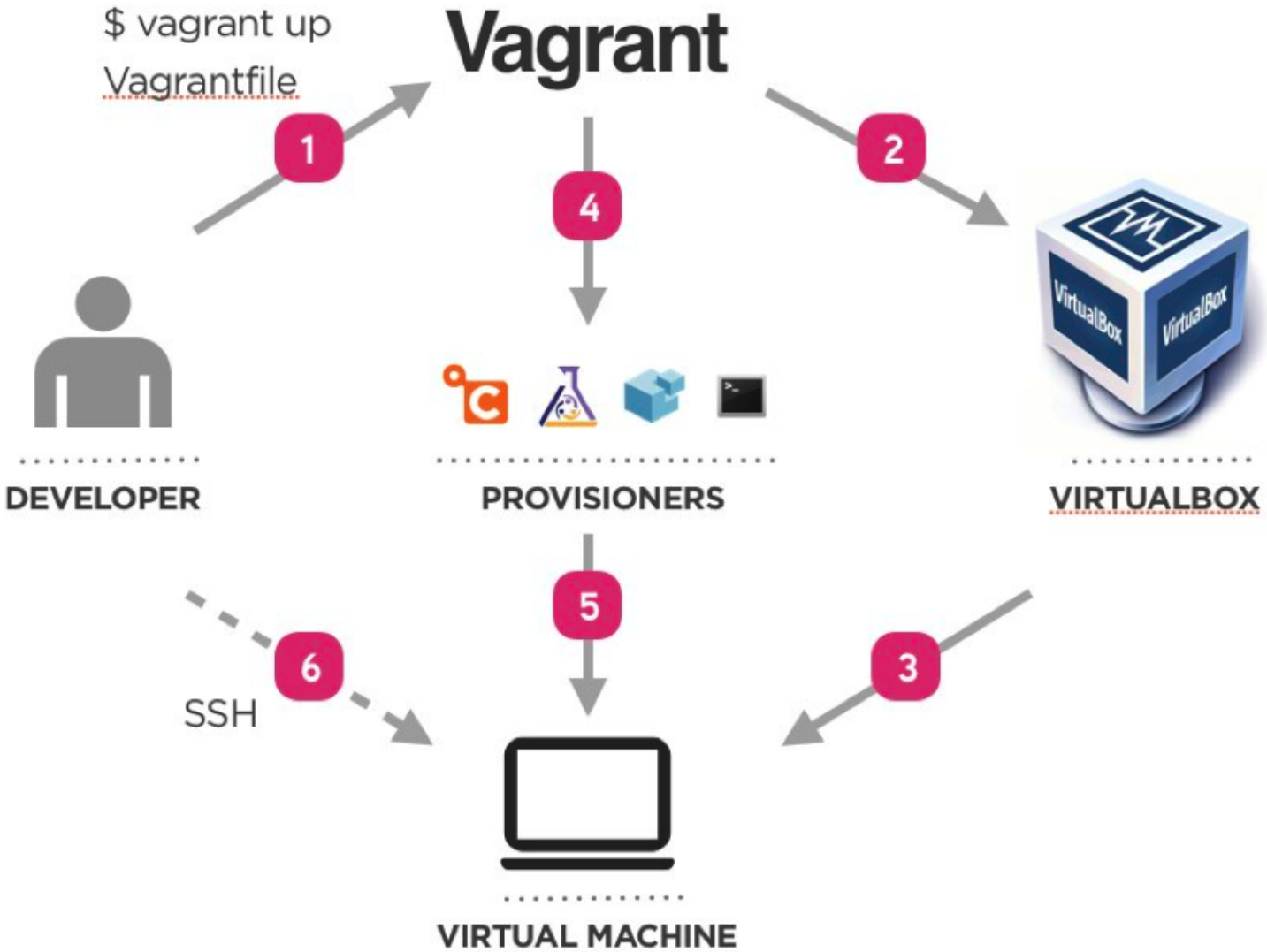
`http://invenio.readthedocs.io/en/latest/installation/installation-quick.html`

Virtual Machines vs Containers



http://qafe.com/what-is-docker-why-en-how-use-it/

VirtualBox and Vagrant

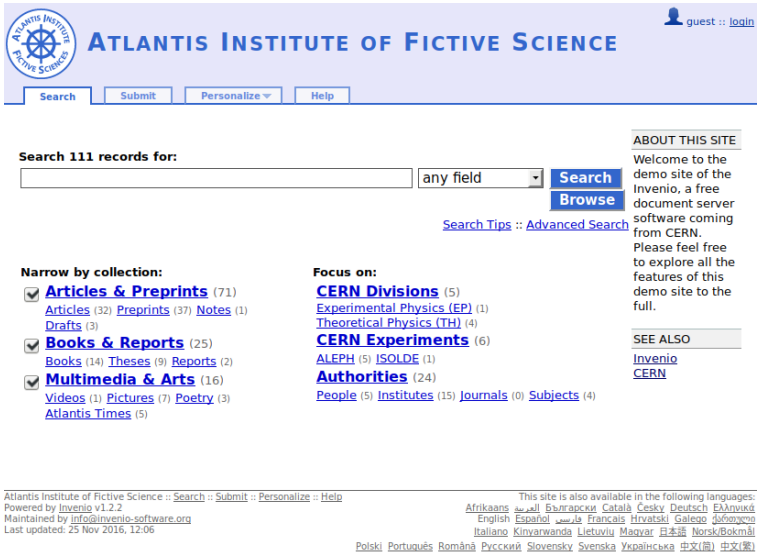


https://wiki.tankywoo.com/tool/vagrant.html

Example: Invenio Vagrantfile

```
OS = 'ubuntu/trusty64'

Vagrant.configure("2") do |config|

  if Vagrant.has_plugin?("vagrant-cachier")
    config.cache.scope = :box
  end

  config.vm.define "web" do |web|
    web.vm.box = OS
    web.vm.hostname = 'web'
    web.vm.provision "file", source: ".inveniorc", destination: ".inveniorc"
    web.vm.provision "shell", inline: "source .inveniorc && /vagrant/scripts/provision-web.sh", privileged: fal
    web.vm.network "forwarded_port", guest: 80, host: 80
    web.vm.network "forwarded_port", guest: 5000, host: 5000
    web.vm.network "private_network", ip: ENV.fetch('INVENIO_WEB_HOST','192.168.50.10')
    web.vm.provider :virtualbox do |vb|
      vb.customize ["modifyvm", :id, "--memory", "1024"]
      vb.customize ["modifyvm", :id, "--cpus", 2]
    end
  end

  config.vm.define "postgresql" do |postgresql|
    postgresql.vm.box = OS
    postgresql.vm.hostname = 'postgresql'
    postgresql.vm.provision "file", source: ".inveniorc", destination: ".inveniorc"
    postgresql.vm.provision "shell", inline: "source .inveniorc && /vagrant/scripts/provision-postgresql.sh", p
    postgresql.vm.network "private_network", ip: ENV.fetch('INVENIO_POSTGRESQL_HOST','192.168.50.11')
  end
  [...]
```

Task 1: Invenio v1.2.2 on Vagrant



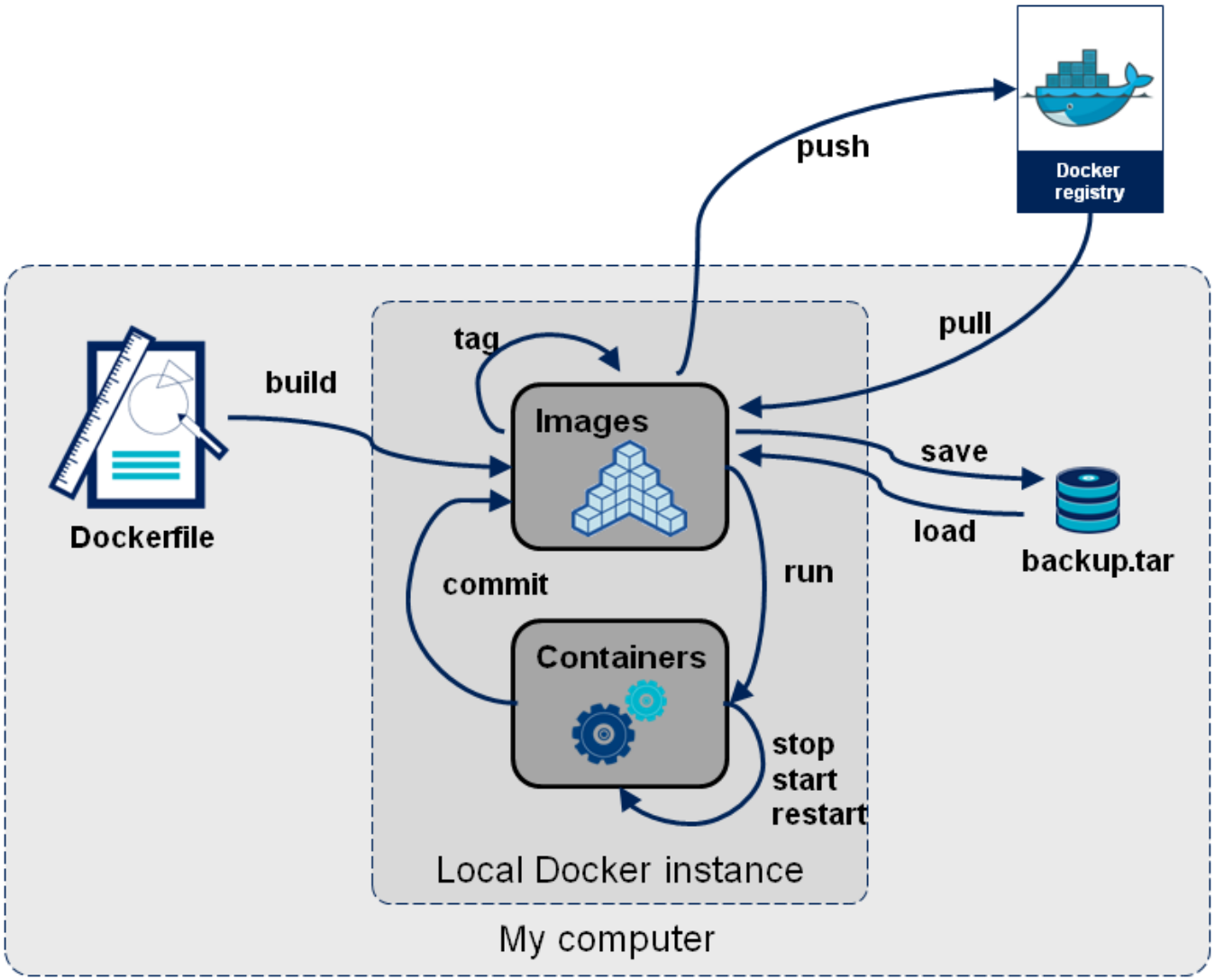
Install Invenio 1.2.2 locally via Vagrant.

Observe multiple machines emulating production setup.

Task 1: answer

```
$ mkdir src
$ cd src
$ git clone https://github.com/inveniosoftware/invenio
$ cd invenio
$ git checkout maint-1.1
$ vagrant up
$ vagrant ssh web -c 'source .inveniorc && /vagrant/scripts/create-instance.sh'
$ vagrant ssh web -c 'source .inveniorc && /vagrant/scripts/populate-instance.sh'
$ firefox http://192.168.50.10/
```

Docker



http://blog.octo.com/en/docker-registry-first-steps/

On your USB stick

```
data
    World_historical_and_predicted_populations_in_percentage.csv

docker
    invenio-1.2.2.img

    invenio_nginx.img                   postgres.img
    invenio_static.img                  rabbitmq.img
    invenio_web.img                     redis.img

    zenodo_es.img                       kibana.img
    zenodo_frontend.img
    zenodo_lb.img
    zenodo_static.img
    zenodo_statsd.img
    zenodo_web.img
    zenodo_worker.img

src
    invenio
    zenodo
```

Task 2: Load docker images

Load provided docker images from pre-saved files.

Task 2: answer

```
$ docker images

$ docker load < invenio-1.2.2.img

$ docker images
REPOSITORY           TAG     IMAGE ID       CREATED      SIZE
tiborsimko/invenio   1.2.2   12a42499b670   5 days ago   2.354 GB
```

Task 3: Docker basics

Download Python docker image. (optional)

Start a simple one-time container.

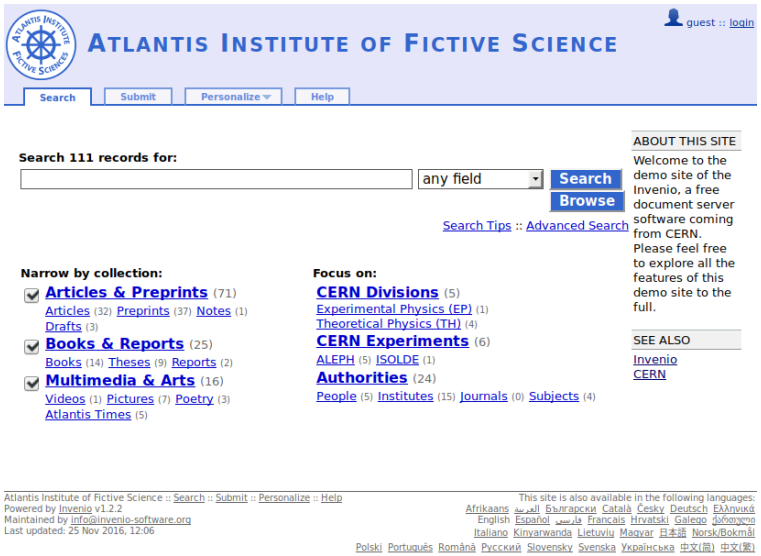
Add two numbers.

Observe images and running and “sleeping” containers.

Task 3: answer

```
$ docker pull python
Using default tag: latest
latest: Pulling from library/python
Digest: sha256:f142b7cd6ae3538ba2c661999003099ef1e1e3fb9d79732170d82280871f
Status: Downloaded newer image for python:latest

$ docker run -i -t --rm python /bin/bash
root@5962013475fa:/# python
Python 3.5.2 (default, Nov 17 2016, 22:42:56)
[GCC 4.9.2] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> 1+1
2
>>>
root@5962013475fa:/# exit

$ docker images

$ docker ps -a
```

Task 4: Invenio v1.2.2

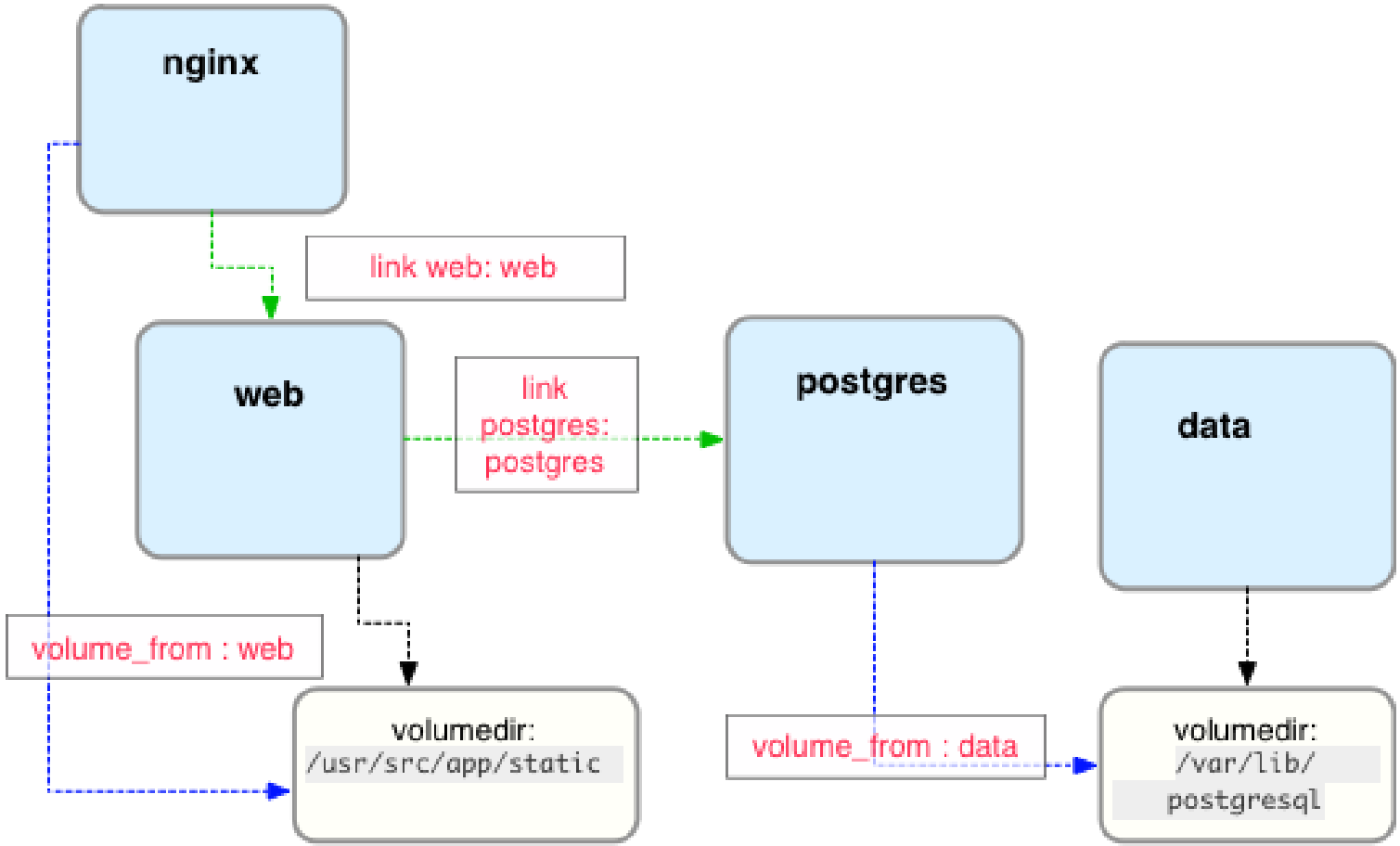


Install Invenio v1.2.2 demo site via provided simple docker environment.

Task 4: answer

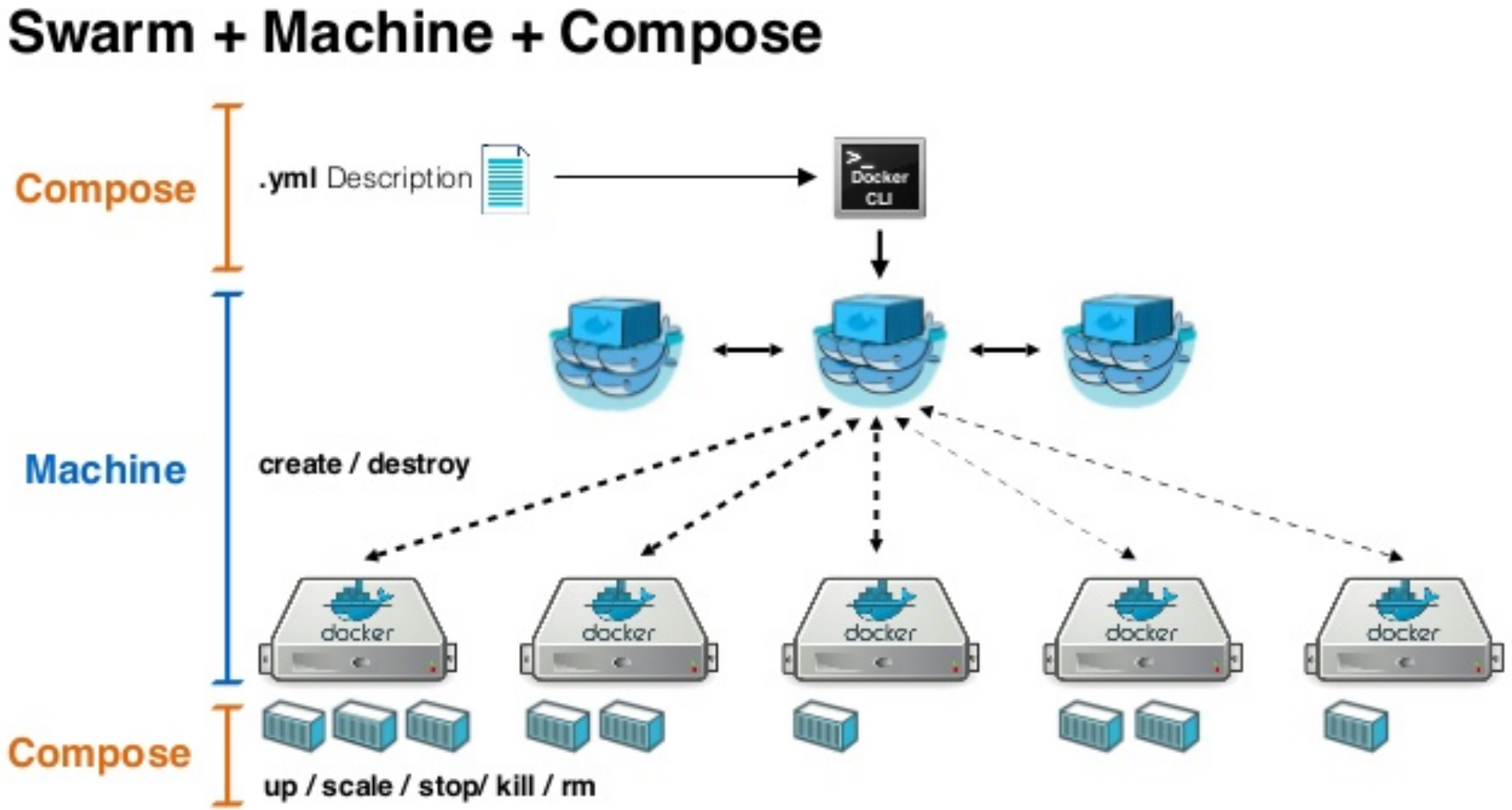
```
$ docker pull tiborsimko/invenio:1.2.2
$ docker run -i -t --rm -p 80:80 -p 443:443 \
       tiborsimko/invenio:1.2.2
$ firefox http://0.0.0.0/
```

Docker Compose



http://containertutorials.com/docker-compose/nginx-flask-postgresql.html

Docker Swarm

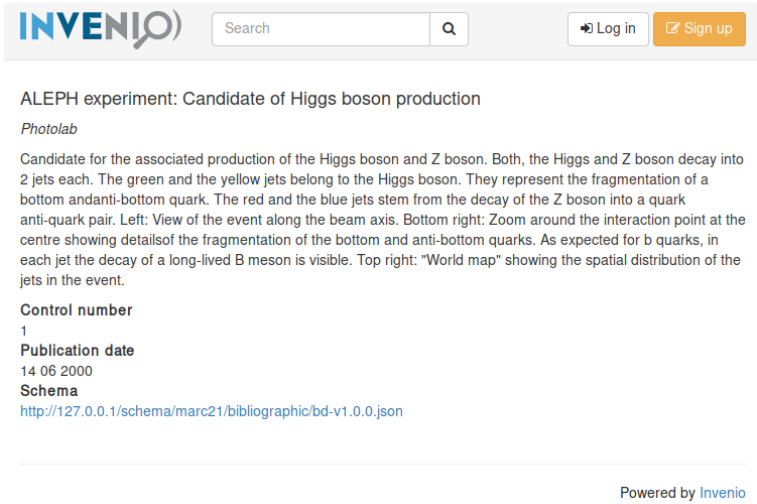


http://www.slideshare.net/Docker/docker-online-meetup-28-productionready-docker-swarm

Example: Invenio docker compose

```
web:
  restart: "always"
  build: .
  command: /bin/bash -c "invenio3 run -h 0.0.0.0"
  environment:
    - PATH=/home/invenio/.virtualenvs/invenio3/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
    - VIRTUALENVWRAPPER_PYTHON=/usr/local/bin/python
    - INVENIO_WEB_HOST=127.0.0.1
    - [...]
  volumes_from:
    - static
  links:
    - postgresql
    - redis
    - elasticsearch
    - rabbitmq
  ports:
    - "5000:5000"

postgresql:
  restart: "always"
  image: postgres
  environment:
    - POSTGRES_USER=invenio3
    - POSTGRES_DB=invenio3
    - POSTGRES_PASSWORD=dbpass123
  ports:
    - "25432:5432"
[...]
```

Task 5: Invenio v3.0 alpha



Install Invenio v3.0 alpha demo site via provided complex docker environment.

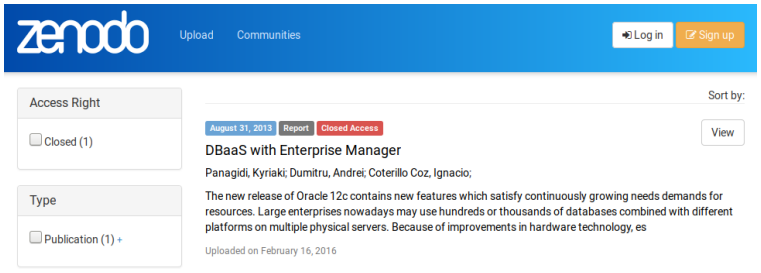
Load demo records.

Execute a simple task in the running container.

Task 5: answer

```
$ cd src/invenio
$ docker-compose build
$ docker-compose up -d
$ docker-compose run --rm web ./scripts/populate-instance.sh
$ firefox http://127.0.0.1/records/1

$ docker-compose run --rm web /bin/bash
$ docker exec -i -t invenio_web_1 /bin/bash
```

Task 6: Zenodo



Install Zenodo locally via provided complex docker environment.

Load demo records and launch indexing processes.

Observe a plethora of containers emulating full production setup.

Task 6: answer

```
$ cd src/zenodo
$ docker-compose build
$ docker-compose up
$ docker-compose run --rm web bash /code/zenodo/scripts/init.sh
$ docker-compose run --rm statsd bash /init.sh
$ docker-compose run --rm web zenodo fixtures loaddemorecords
$ docker-compose run --rm web zenodo migration recordsrun
$ docker-compose run --rm web zenodo index reindex --yes-i-know
$ docker-compose run --rm web zenodo index run -d
$ firefox https://0.0.0.0/
```