

Docker sur le poste du développeur

Docker

Docker est une plateforme permettant de lancer certaines applications dans des conteneurs logiciels.

— Wikipedia, [Docker \(logiciel\)](#)

Docker Compose

Docker Compose est un logiciel pour définir et exécuter des applications à partir de multiples conteneurs. Il est basé sur un fichier YAML qui permet de définir les services et les paramètres de leurs créations et ainsi de les démarrer par une commande unique.

— Wikipedia, Docker (logiciel) : [Outils associés](#)

Docker Hub

Catalogue d'images docker.

hub.docker.com

Ouvrir un terminal dans un container

Cas d'utilisation 1/2

Docker et docker-compose peuvent être utiles au quotidien pour les développeurs.

- Utiliser des outils sans les installer
- Tester un logiciel compliqué à installer
- Test containers
- Reproduire la CI en local
- Environnements de développement conteneurisés

Cas d'utilisation 2/2

- Lancer des logiciels dont on ne connaît pas la technologie
- Tester une nouvelle version d'un langage

- Être admin dans un container
- Infrastructure éphémère
- Bac à sable
- Simuler l'environnement cible en local

01 Utiliser des outils sans les installer

- Éviter de polluer son environnement
- base de données
- serveur d'authentification

RabbitMq — installation classique

Installation classique de RabbitMq

[rabbitmq manual installation] | [./assets/rabbitmq_manual_installation.gif](#)

RabbitMq — lancement avec docker

Asciidoctor

Asciidoctor est un format de markup. C'est aussi un outil qui permet de générer du contenu.

- html
- pdf
- slides
- ebook

Démo Asciidoctor

Ces slides ^[1]

`docker-compose.yml`

```
build-slides:
  image: "asciidoctor/docker-asciidoctor"
  volumes:
    - ./:/documents/
  command:
    - "asciidoctor-revealjs"
    - "-a"
    - "revealjsdir=https://cdnjs.cloudflare.com/ajax/libs/reveal.js/3.9.2"
    - "01-slides/index.adoc"
```

- "-o"
- "docs/index.html"

02 Tester un logiciel compliqué à installer

- Par exemple **Gitlab (nécessite beaucoup de mémoire)
 - Nextcloud
- Pratique pour jouer avec la configuration

Démo Nextcloud + PostgreSQL

```
services:
  nc:
    # http://localhost/login
    # admin / admin
    image: nextcloud:apache
    environment:
      - POSTGRES_HOST=db
      - POSTGRES_PASSWORD=nextcloud
      - POSTGRES_DB=nextcloud
      - POSTGRES_USER=nextcloud
    ports:
      - 80:80
    restart: always
    volumes:
      - .volumes/nc_data:/var/www/html
  db:
    image: postgres:alpine
    environment:
      - POSTGRES_PASSWORD=nextcloud
      - POSTGRES_DB=nextcloud
      - POSTGRES_USER=nextcloud
    restart: always
    volumes:
      - .volumes/db_data:/var/lib/postgresql/data
  expose:
    - 5432
```

<http://localhost:80>

03 Test Containers

- lance un conteneur juste le temps des tests
- bases de données

- Nginx
- rabbitMq

testcontainers.org

Spring Boot + TestContainers + Redis 1/2

```
@RunWith(SpringJUnit4ClassRunner.class)
@SpringBootTest(classes = DemoApplication.class, webEnvironment =
WebEnvironment.RANDOM_PORT, properties = {
    "spring.datasource.url=jdbc:tc:postgresql:11-alpine:///databasename",
})
@ActiveProfiles("test")
public abstract class AbstractIntegrationTest {

    static GenericContainer<?> redis = new
GenericContainer<>(DockerImageName.parse("redis:3-alpine"))
        .withExposedPorts(6379);

    @DynamicPropertySource
    static void redisProperties(DynamicPropertyRegistry registry) {
        redis.start();
        registry.add("spring.redis.host", redis::getHost);
        registry.add("spring.redis.port", redis::getFirstMappedPort);
    }
}
```

Spring Boot + TestContainers + Redis 1/2

```
public class DemoControllerTest extends AbstractIntegrationTest {

    @Autowired
    TestRestTemplate restTemplate;

    @Autowired
    DemoRepository demoRepository;

    @Test
    public void simpleTest() {
        String fooResource = "/foo";

        restTemplate.put(fooResource, "bar");

        assertThat(restTemplate.getForObject(fooResource, String.class)).as("value is set").isEqualTo("bar");
    }
}
```

```

@Test
public void simpleJPATest() {
    DemoEntity demoEntity = new DemoEntity();
    demoEntity.setValue("Some value");
    demoRepository.save(demoEntity);

    DemoEntity result = restTemplate.getForObject("/" + demoEntity.getId(),
    DemoEntity.class);

    assertThat(result.getValue()).as("value is set").isEqualTo("Some value");
}
}

```

Démo

<https://github.com/baldir-fr/bbl-docker-pour-le-developpeur/tree/main/03-test-containers/spring-boot>

04 Reproduire la CI en local

Lancer des scripts sur le même environnement que la CI

05 Environnements de développement conteneurisés

- Vs code remote
- Gitpod
- Code Server

Code Server

<https://github.com/coder/code-server>

Editeur de code pré-configuré sur le navigateur.

Démo

```

05-code-server:
  image: registry.gitlab.com/crafting-software/online-code/js
  user: code
  ports:
    - "8094:8080"
  environment:

```

- PASSWORD=testing

<http://localhost:8094>

06 Lancer des logiciels dont on ne connaît pas la technologie

- Par exemple springboot si on fait du JavaScript
- Keycloak
- ffmpeg + imagemagick

[Awesome Docker Compose starters](#)

Conversion de .mov en .gif

<https://toub.es/2017/09/11/high-quality-gif-with-ffmpeg-and-docker/>

```
alias video2gif='sudo docker run -v=`pwd`: /tmp/ffmpeg kafebob/video2gif'
```

```
video2gif rabbitmq_manual_installation.mov rabbitmq_manual_installation.gif
```

07 Tester une nouvelle version d'un langage

- (ex. java 11 java 17..)

Java 8

```
07-movie-rental-java-8:
  image: "maven:3-openjdk-8-slim"
  working_dir: "/java-app"
  volumes:
    - "./07-movie-rental-java-8:/java-app"
    - ".:/volumes/07-movie-rental-java-8-.m2:/root/.m2"
  command:
    - "mvn"
    # - "-Dmaven.compiler.source=8"
    - "clean"
    - "test"
```

Java 17

```
07-movie-rental-java-17:
  image: "maven:3-openjdk-17-slim"
  working_dir: "/java-app"
  volumes:
    - "./07-movie-rental-java-17:/java-app"
    - ".:/volumes/07-movie-rental-java-17-.m2:/root/.m2"
  command:
    - "mvn"
#    - "-Dmaven.compiler.source=17"
    - "clean"
    - "test"
```

08 Être admin dans un conteneur

- quand on n'est pas admin de sa machine sur Windows par exemple
- installer des paquets sur une image

09 Infrastructure éphémère

Lancer un serveur pour le temps d'un événement * Pour un événement donné, lancer un serveur *
Le supprimer à la fin de l'événement * Permet aux participants d'accéder à l'outil sans s'inscrire
par exemple (les données sont supprimées à la fin de l'événement)

10 Bac à sable

- Expérimenter sans risques
- Essayer des choses risquées dans un environnement safe
- apprendre les commandes linux sans tout casser
 - sous réserve de certaines précautions

Apprendre les commandes linux

[Busybox](#) : [Commandes disponibles](#)

```
docker run -it --rm busybox
```

[Alpine](#)

```
docker run -it --rm alpine
```

11 Simuler l'environnement cible en local

- ex depuis windows exécuter sur linux

Keycloak + Vuejs + Spring-boot

image

assets/11-docker-compose.png

☐ Merci

Des questions? ☐

[1] <https://github.com/baldir-fr/bbl-docker-pour-le-developpeur>