



POLYTECHNIQUE MONTRÉAL

LOG3000

PROCESSUS DU GÉNIE LOGICIEL

Travail Pratique #5

DÉPLOIEMENT BLEU-VERT DANS OPENSIFT

<i>Chargé</i>	<i>Courriel</i>
<i>Varun Shiri</i>	varun.shiri@polymtl.ca

Automne 2022

1. Mise en contexte théorique

De nos jours, le pipeline de production et de déploiement de logiciels devient de plus en plus complexe et difficile. Récemment, les développeurs de logiciels pouvaient cibler un utilisateur final spécifique, tel qu'un ordinateur de bureau Windows, des applications Web simples ou des applications mobiles. Bien que certains défis aient été relevés, la situation est devenue encore plus compliquée aujourd'hui avec l'avènement des clouds, appareils intelligents, le IoT et des appareils interconnectés. Ainsi, l'architecture des applications est en train de s'éloigner des applications monolithiques traditionnelles au profit des conteneurs et des micro-services.

Heureusement, une panoplie de stratégies de déploiement et de mise en production ont été proposées, comme vues en cours. Ce TP se concentrera sur la stratégie populaire de **déploiement bleu-vert**.

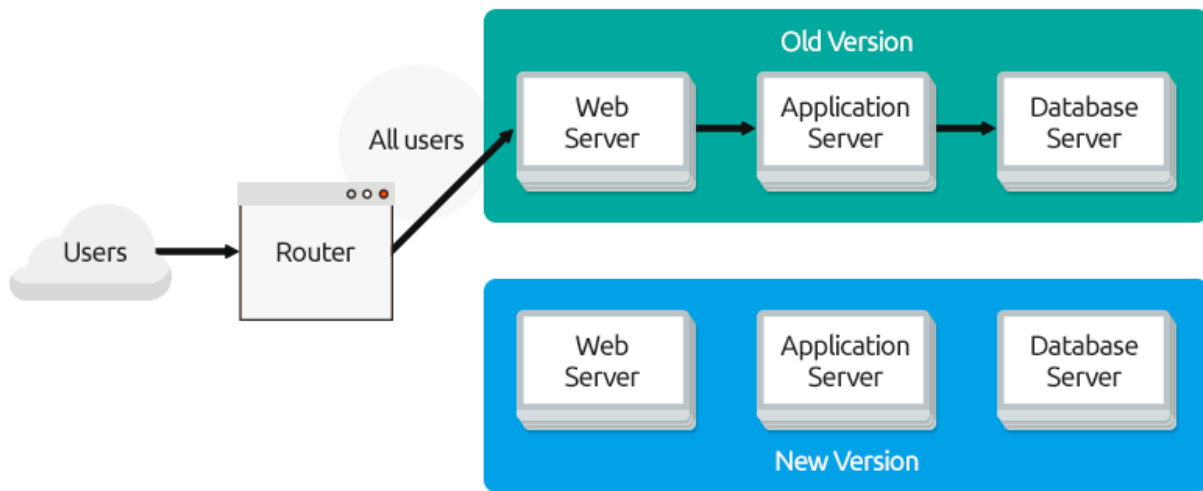


Figure 1. Le déploiement bleu-vert.

Les déploiements bleu-vert constituent un modèle qui élimine le temps d'arrêt pendant les déploiements de production en ayant deux environnements de production ("bleu" et "vert") entre lesquels le déploiement alterne.

Bref, le déploiement Bleu-Vert signifie avoir deux environnements (idéalement) identiques à la fois, l'un actif (vert), l'autre inactif (bleu). Les mises à jour sont poussées vers l'environnement inactif (bleu) et sont testées jusqu'à ce que les opérateurs soient contents. Ensuite, le nouveau trafic est dirigé vers l'environnement bleu, tandis que l'environnement vert devient inactif (finissant juste les sessions déjà ouvertes). En cas de problèmes, ce qui est possible même après des tests approfondis, le trafic est alors

rétabli (“roll-back”) dans l'environnement vert précédemment actif, comme illustré par la Figure 1.

2. Objectifs

L'objectif de ce TP est de vous aider à comprendre comment mettre en œuvre un déploiement bleu-vert. L'objectif auxiliaire est de se familiariser avec la plateforme OpenShift pour la gestion de plusieurs conteneurs Docker exécutant des microservices.

OpenShift est un outil de gestion pour l'infrastructure de cloud privé et virtuel développé par Red Hat®. OpenShift est une plateforme d'applications à conteneur qui permet à Docker et à Kubernetes de s'intégrer à l'entreprise. Quelle que soit l'architecture de votre application, OpenShift permet de construire, développer et déployer facilement et rapidement dans presque toutes les infrastructures, publiques ou privées. Que ce soit localement, dans un cloud public ou hébergé, vous disposez d'une plate-forme primée pour mettre votre prochaine grande idée en avant de vos concurrents.

3. Mise en contexte pratique

Le gouvernement du Québec a annoncé son intention d'étendre les transports en commun à l'ensemble de la province. Ainsi, il reliera toutes les grandes villes et banlieues avec différentes catégories de bus et de systèmes de métro. Actuellement, différentes villes exploitent leurs systèmes de transport différemment, ce qui empêche les voyageurs de planifier leur itinéraire correctement. Par exemple, une personne de Montréal qui souhaite visiter Sherbrooke et Québec et revenir à Montréal le même jour devrait pouvoir le faire avec un minimum de temps d'attente; connexion d'un système de transport à un autre.

Par ailleurs, le gouvernement du Québec a créé une agence appelée "agence de transport commune" (ATC), qui supervisera tous les besoins en matière de transport et de logistique de toutes les résidences du Québec.

ATC a donc demandé à votre expertise en tant qu'ingénieurs de DevOps de fournir une plateforme de déploiement sur laquelle ils peuvent tester de manière fiable différentes catégories d'applications. Par exemple, l'une de ces plateformes de déploiement devrait être la plateforme de déploiement bleu-vert.

Par conséquent, vous devez fournir des plateformes de travail que ATC peut utiliser pour déployer toutes ses applications au plus tard le 30 novembre 2022 à 23h55. Cette plateforme devrait exécuter l'application existante que l'ATC possédait et devrait pouvoir basculer vers une version plus récente une fois que les développeurs ont fini de créer l'application.


Astuce: ATC n'a besoin que de votre expertise pour créer des plateformes de déploiement opérationnelles en tant qu'ingénieurs DevOps. Donc, vous devriez penser longtemps à l'avance, quelle plateforme devrait bien fonctionner dans telle ou telle situation.

En premier lieu, ATC vous a donné accès à un système en ligne OpenShift¹, que vous devez utiliser pour créer les plateformes.

Cliquez le lien ci-dessous pour commencer le tutoriel:

[Getting started with OpenShift for Developers](#)

Remarque: La création de l'environnement peut prendre quelques minutes. Veuillez patienter.

Home

instruqt

Progress

Goal

The goal of this track is to learn how to use the OpenShift Container Platform to build and deploy an application that has a data backend and a web frontend.

What you need to know before you start

In order to get full benefit from taking this track you should...

- Be comfortable working at the command line in a terminal window.
- Have a general understanding about the purpose and use of [OpenShift](#)

Concepts and techniques you'll cover

¹ <https://www.redhat.com/en/technologies/cloud-computing/openshift>

Suivez les instructions étape par étape.

Remarque: Préfixez toutes vos commandes avec sudo afin de les faire fonctionner.



4. Travail à effectuer

4.1 Mise en œuvre pratique

Le TP5 commence avec un tutoriel² sur le fonctionnement d'OpenShift et sur le déploiement d'une application sur OpenShift, qui nous aideront à réaliser le TP5. Ce tutoriel vous donne un aperçu de la meilleure pratique de DevOps en matière de modèles de déploiement avancés comme le déploiement Bleu-Vert. Voici quelques liens [le post-scriptum 3 et 4] utiles pour vous aider à comprendre les implémentations pratiques du déploiement bleu-vert³, avec des cas d'utilisation⁴ similaires à ceux de ATC.

² <https://developers.redhat.com/courses/openshift/getting-started>

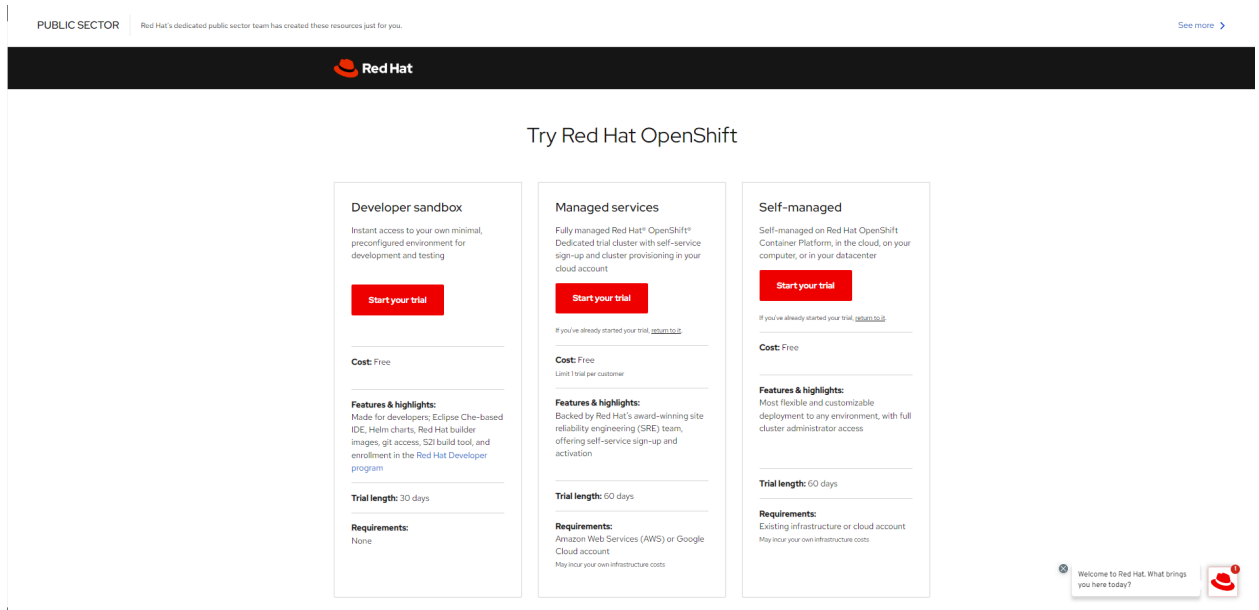
³

https://access.redhat.com/documentation/en-us/openshift_container_platform/3.7/html/upgrading_clusters/upgrading-blue-green-deployments

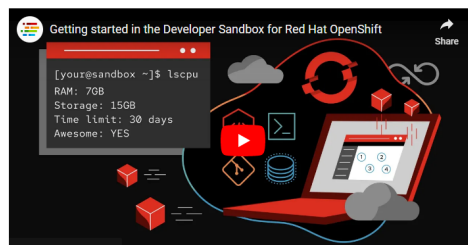
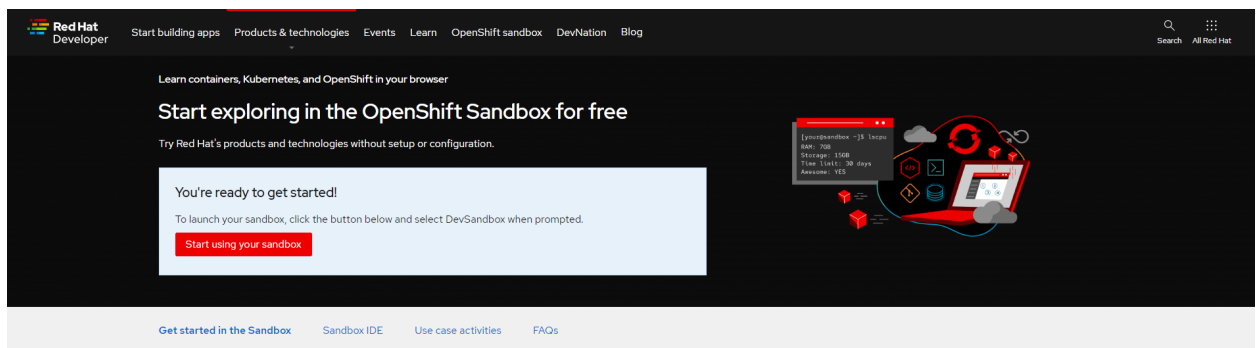
⁴ <https://opensource.com/article/17/5/colorful-deployments>

Finalement, l'architecture de notre application conteneurisée de l'ATC ressemblera à celle de la Figure 1 sur page 2. **Une fois le tutoriel fini, il faudra, dans le portail OpenShift en ligne, créer l'implémentation du déploiement bleu-vert de l'application pour l'ATC (voir question 8).** Vous pouvez utiliser OpenShift en ligne ([Try Red Hat OpenShift](#)). Suivez les étapes ci-dessous:

1. Sélectionnez l'option "Developer Sandbox".



2. Créez un compte Red Hat.
3. Configurez votre sandbox. Cette étape peut prendre quelques minutes.
4. Lancez le sandbox en appuyant "Start using your sandbox".



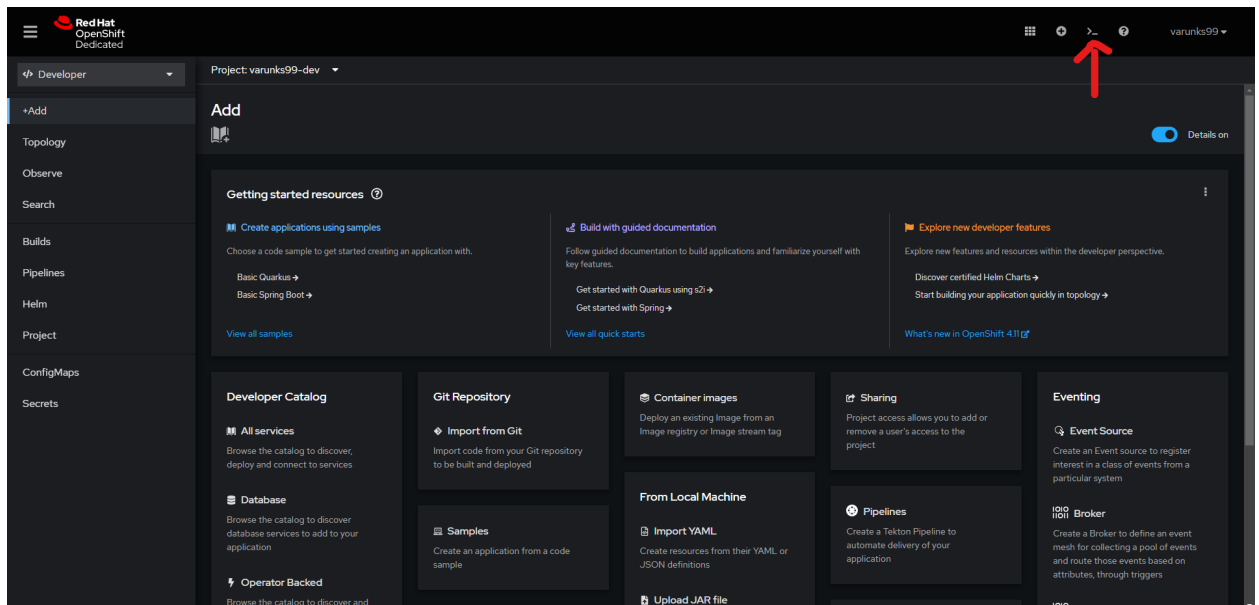
What is the OpenShift sandbox?

Get 30-days free access to a shared OpenShift and Kubernetes cluster.

- 1 Guided tutorials to experience and run sample applications
- 2 Modify and build on top of [pre-built sample applications](#)
- 3 Access the browser-based CodeReady Workspaces IDE
- 4 GitHub integration to your own source code

Start building with our [guided activities](#) or bring your own code.

5. Une fois dans la console, appuyer sur le bouton marqué ci-dessous pour lancer le terminal.



6. Suivez les instructions de la **question 8** pour effectuer le déploiement bleu-vert. Vous pouvez également effectuer toutes les étapes à l'aide de l'interface de la console. Consultez le [tutoriel](#) pour savoir comment faire.

5. Questions

Q1. Dans chaque stratégie de déploiement ci-dessous, donnez la stratégie de déploiement appropriée qu'elle représente et un exemple concret.

Cas 1.

State 0



State 1



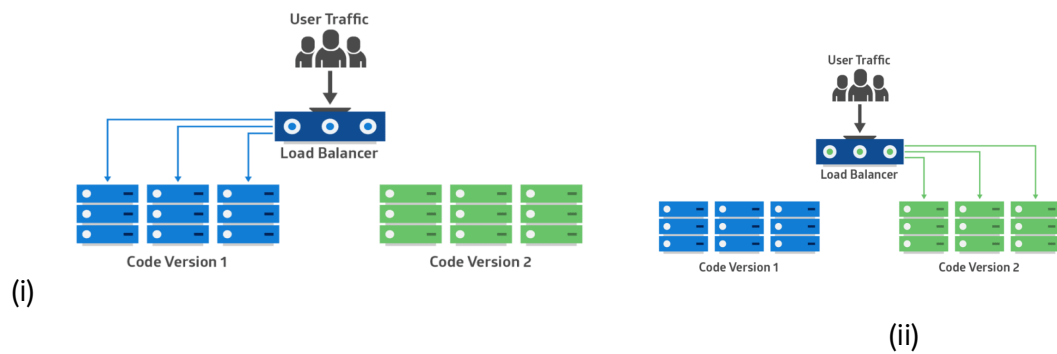
State 2



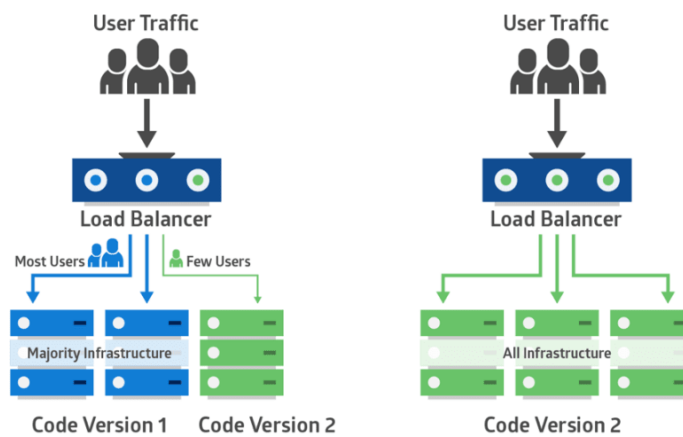
Final State



Cas 2.



Cas 3.



Q2. Quelle est la relation entre OpenShift et Kubernetes?

Q3. Dans vos propres mots, qu'est-ce que le Kubernetes et pourquoi est-il utile pour DevOps?

Q4. Décrivez quelques avantages de la mise à l'échelle d'une application.

Q5. Qu'est-ce qu'un pod? Expliquez comment ils fonctionnent.

Q6. Qu'est-ce que "l'auto-guérison d'application"?

Q7. Quel est le but du routage?

*Q8. Maintenant, vous devez implémenter le déploiement bleu-vert⁵ avec le code source disponible sur GitHub (<https://github.com/varunks99/blue-green-openshift>). Il déploiera l'application ci-dessous.



- Vous devez « fork » le répertoire et faire les modifications nécessaires.
- Vous devez expliquer chaque étape du déploiement. De plus, il est nécessaire d'ajouter des captures d'écran pour appuyer vos explications ainsi que le résultat du fonctionnement des versions bleu et vert.
- Vous devez créer votre propre branch "vert". Il faut changer la couleur du cercle en vert.
- La correction des problèmes/bogues rencontrés lors de l'implémentation et du déploiement fait partie du TP.

****Prenez des captures d'écran et incluez-les dans votre rapport (PDF).***

5.1 Question de rétroaction

Nous travaillons à l'amélioration continue des travaux pratiques de LOG3000. Cette question peut être répondue très brièvement.

1. Combien de temps avez-vous passé au travail pratique, en heures-personnes, en sachant que deux personnes travaillant pendant trois heures correspondent à six heures-personnes. Est-ce que l'effort demandé pour ce laboratoire est adéquat ?

⁵ http://redhatgov.io/workshops/openshift_101_dcmetroamap/lab9-bluegreen/

6. Livrable à remettre, procédure de remise et retard

Les fichiers à remettre sont les suivants :

- Le rapport complet en format PDF contenant vos réponses à toutes les questions.

La remise doit se faire sur Moodle avant la date suivante.

Groupe 01 et 02	30 novembre 2022 à 23 h 55
------------------------	-----------------------------------

Il y aura une pénalité pour les travaux remis en retard.