



INTRODUCTION AUX SYSTÈMES DISTRIBUÉS

INF36307 – SYSTÈMES DISTRIBUÉS

AGENDA

1

Étude de cas

Le site web de
l'UQAR

2

Résilience

Tolérance aux
pannes

3

Charge

Mise à l'échelle pour
gestion de grande
charge

4

Performance

Temps de réponse
et géo distribution

5

Extensibilité

Capacité au système
d'évoluer

ÉTUDE DE CAS

Tentons d'imaginer le site web de l'UQAR dans sa plus simple expression

- Une application monolithique
- Aucune base de données
- Un serveur unique



ÉTUDE DE CAS

- Quelles pourraient être les technologies utilisées pour ce cas simple?
 - Il y a 10 ans?
 - Aujourd'hui ?

PASSÉ (2010)



PRÉSENT (2020)



PROGRESSION

Introduction

Résilience

Charge

Performance

Extensibilité





ÉTUDE DE CAS

- Imaginons le modèle d'il y a 10 ans
 - Serveur unique
- Quel est l'impact d'un bris de disque dur?
- Quel est l'impact d'un bris de serveur?



Pourquoi distribué un système?

RÉSILIENCE

La redondance est une des applications d'un système distribué qui ajoute de la résilience aux pannes.

En entreprise, on parle régulièrement de “Disaster Recovery” et la majorité des systèmes sont conçus à la base avec une stratégie de gestion transparente des pannes matérielles ou réseau.

PROGRESSION

Introduction

Résilience

Charge

Performance

Extensibilité



IMPACT D'UNE LOURDE CHARGE SUR UN SYSTÈME

- Lors de la période d'inscription de site web de l'UQAR recoit une charge record
 - 10 000 fois la charge normale
- Quel est l'impact sur le système actuel?
 - Un serveur
 - Application monolithique (apache, PHP)



Fait vécu : Un startup qui passe à TVA en direct



MISE À L'ÉCHELLE

- Souvent appelé « scaling »
- S'applique selon 2 stratégies:
 - Verticale
 - Horizontale





MISE À L'ÉCHELLE VERTICALE

- Consiste à ajouter des ressources à un serveur
 - Mémoire
 - CPU
 - Espace disque
 - Bande passante
- Cette mise à l'échelle ne peut pas croître indéfiniment

Dans la plus part des cas, lors de la création d'un nouveau système, les ressources sont configurées pour que toute la charge puisse être traitée par un seul serveur.

Une deuxième instance est ajoutée sur un autre serveur en redondance pour assuré la résilience.

Lorsque la charge augmente, la stratégie de mise à l'échelle est révue.

MISE À L'ÉCHELLE HORIZONTALE

- Consiste à ajouter de nouvelles copies d'un serveur
- Utilise des notions de distribution des charges ou « load balancing »
- Est normalement plus rapide à appliquer que la mise à l'échelle verticale
- Permet la mise à l'échelle automatique ou « auto-scaling »
 - Adaptation de la flotte de serveur en fonction de la charge
- Ajoute de la complexité pour le partage des ressources
 - User session, Data replication
- Utilise des notions de virtualisation ou d'hébergement par conteneurs

Netflix sur Amazon Web Service

Startup: PHP & sticky session



PROGRESSION

Introduction

Résilience

Charge

Performance

Extensibilité



IMPACT DE LA DISTANCE SUR LA PERFORMANCE

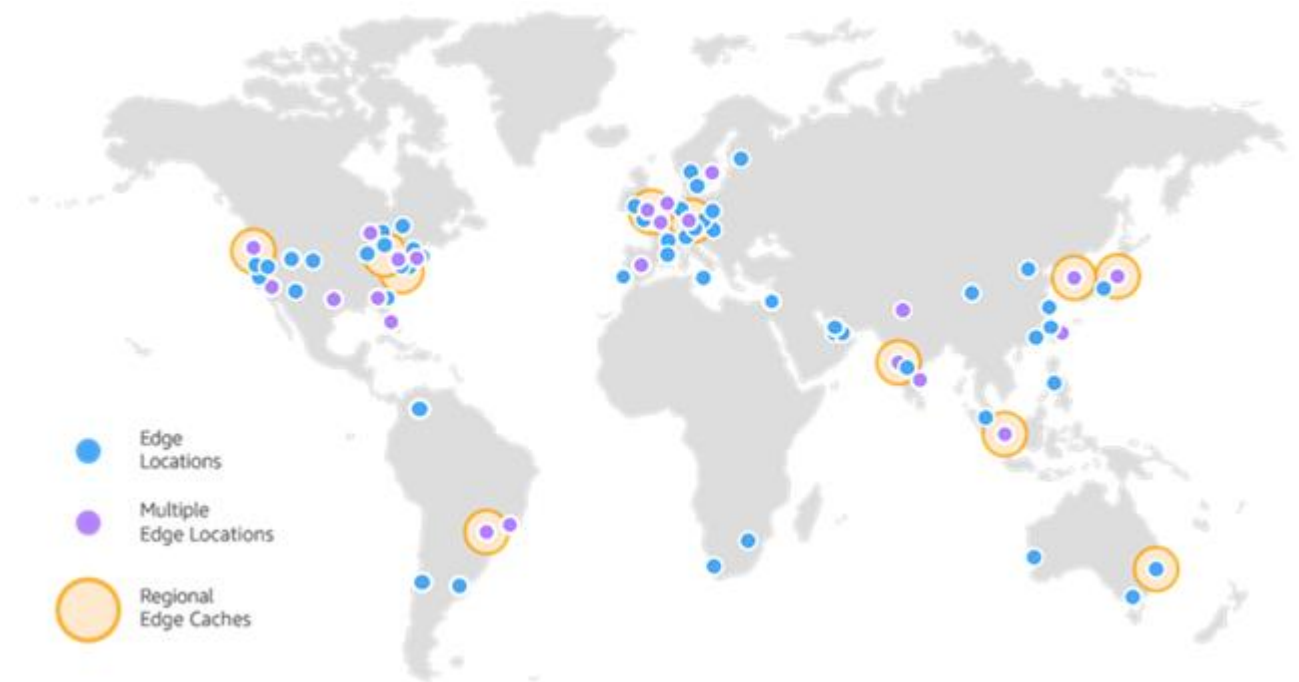
- Quel est l'expérience utilisateur d'un étudiant africain qui désire consulter ses notes de cours depuis son pays d'origine sur le site de l'UQAR?
 - Distance Québec – Afrique : 7000 km
 - Vitesse de la lumière : 300 km/ms
 - Temps théorique > 23 ms
 - Mais dans les faits c'est souvent > 250 ms
- Comment peut-on améliorer le système actuel pour réduire le temps de réponses?





CONTENT DELIVERY NETWORK (CDN)

- Est un système distribué qui vise à héberger les données statiques le plus près possible de l'utilisateur pour diminuer la latence



Source: AWS CloudFront point of presence

DIVISER POUR MIEUX RÉGNER

- Souvent appelé « divide & conquer »
- Est une approche parfois utilisée pour obtenir de la performance
 - Réduction du temps d'exécution
- Fortement utilisée chez Google (ex: MapReduce)
- Elles consiste à découper une tâche en plusieurs morceaux qui seront exécuté en parallèle sur un système distribué



PROGRESSION

Introduction

Résilience

Charge

Performance

Extensibilité



ARCHITECTURE ORIENTÉE SERVICE (SOA)

- Est une approche de la conception d'un système
 - Via un amalgame de services
 - Faiblement couplé entre eux
- Ces services sont souvent appelés “service web” ou “Microservice”
- Ils implémentent une fonction simple et réutilisable
- Le système réalise ses fonctionnalités en appelant plusieurs services
- Le protocole d'échange entre les service est souvent REST ou SOAP
 - On commence à voir gRPC faire son apparition

Dessiner l'Architecture Orienté Service du site web de l'UQAR

The diagram illustrates a network architecture. At the top, a horizontal bar is divided into three segments: dark grey, orange, and light grey. Below this, a large orange rectangle represents the 'Public internet'. A purple rounded rectangle labeled 'Navigateur' is positioned on the left side of this rectangle. To the left of the orange rectangle, a black rounded rectangle labeled 'Load Balancer' is shown. Below the load balancer is a purple square labeled 'Moodle'. A horizontal line separates the Moodle layer from the MySQL layer below. At the bottom left, there is an orange cylinder icon labeled 'Mysql'.

Navigateur

Public internet

Load
Balancer

Moodle

Mysql



Pourquoi distribué un système?

EXTENSIBILITÉ

Décomposer un système en plusieurs sous composants indépendants facilite l'évolution d'un système et permet à plusieurs équipes de collaborer à son développement

Web Service : Data Layer



DEVOIR

ÊTRE EN MESURE D'UTILISER DOCKER SUR VOTRE ORDINATEUR PERSONNEL

```
docker run hello-world
```