POLYTECH GRENOBLE

Netflix Zuul

Veille Technologique

Hugo Amodru-Favin

Contexte

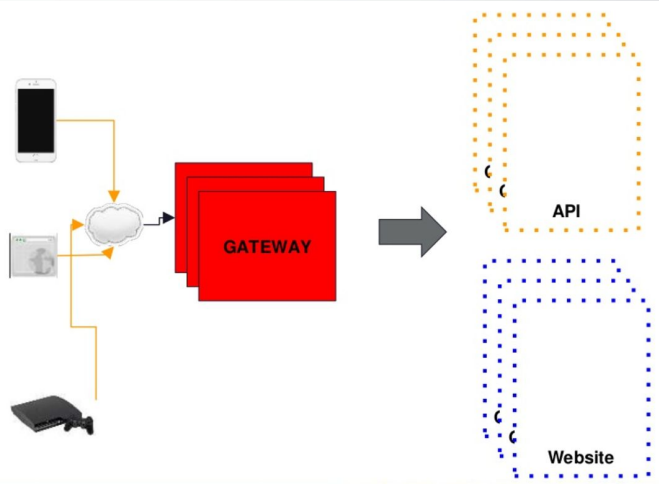
Le statut de Netflix

- 100 Millions d'abonnés
- Utilisable sur plus de 1000 appareils
- 50,000 requêtes par secondes
- Présent dans 190 pays
- 42,000 Millions d'heures de streaming en 2015

Gateway

Qu’est-ce qu’un Gateway ?

- Fait le lien entre Internet et les services dans le Cloud
- Essentiellement utilisé pour gérer un trafic
- Nécessaire au fonctionnement du service



- Faire des checks, des authentifications
- Faire un suivi de requêtes
- Favorise fluidité
- Peut servir de plateforme d’expérimentations

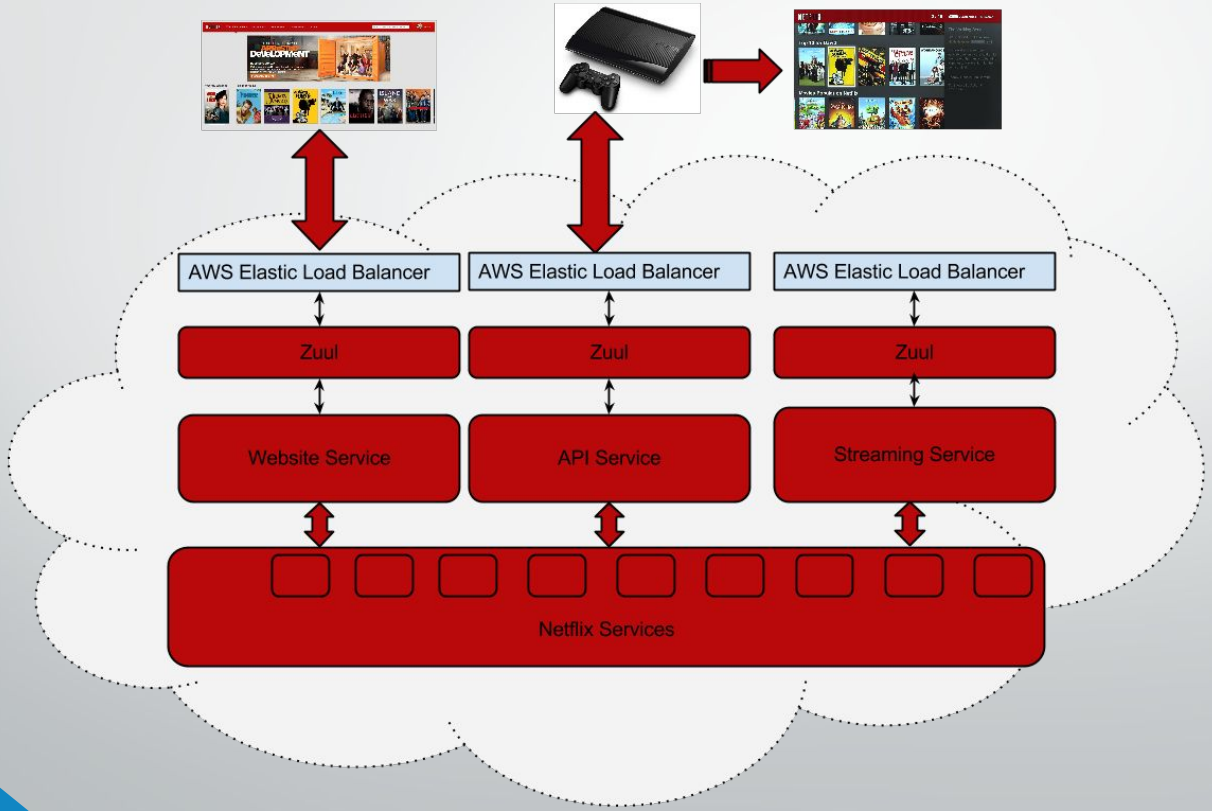
Zuul

Pourquoi Zuul ?

- Des technologies existent déjà: AWS, Nginx, Mashery, etc.

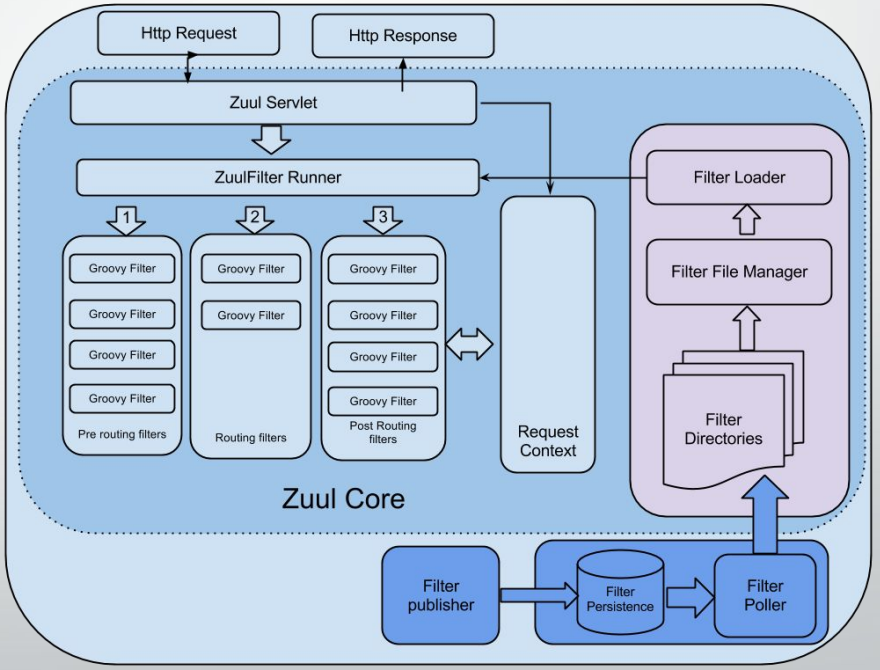
- Personnalisable aux besoins précis du service et de l'API
  - Load Balancing
  - Logique de routing dynamique
  - Vue interne
  - Ancré dans un écosystème de MicroService
  - Protection du service
  - Outil d'assurance qualité

Architecture Cloud de Netflix



Zuul

- Créé en 2012
- Langage Groovy
- Micro Service
- Système de filtres



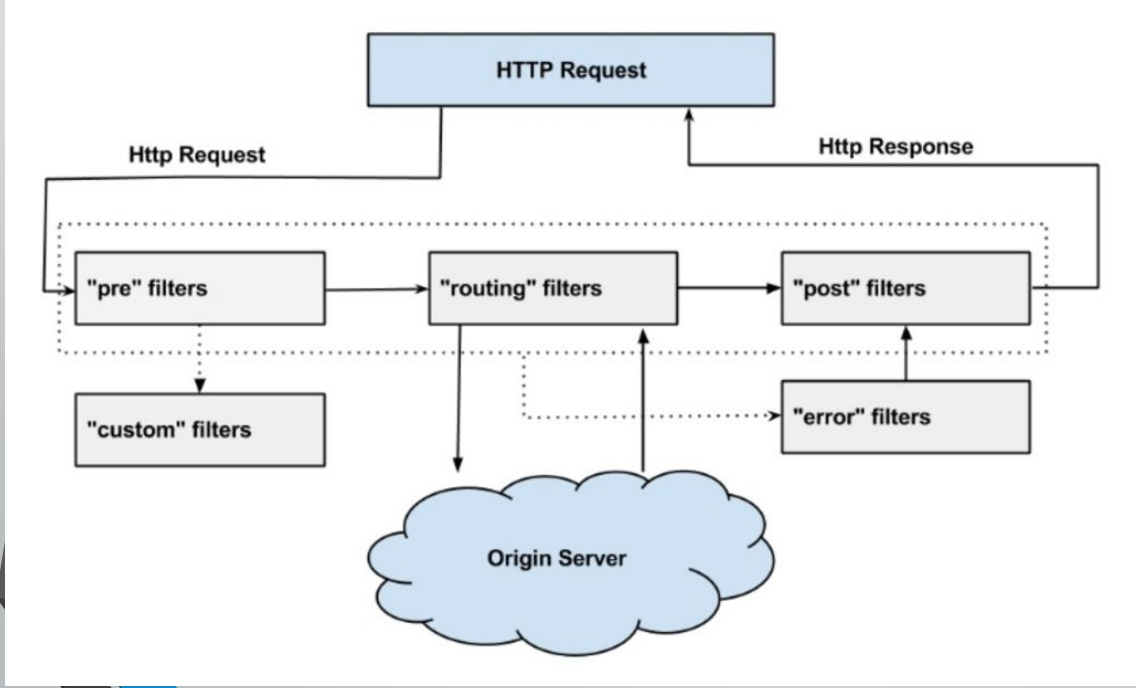
Filtres

- Types
- Ordre d’exécution
- Critères
- Action

```
class DeviceDelayFilter extends ZuulFilter {
   def static Random rand = new Random()
    @Override
    String filterType() {
     return 'pre'
    }
   @Override
   int filterOrder() {
     return 5
   }
   @Override
   boolean shouldFilter() {
       return  RequestContext.getRequest().getParameter("deviceType")?equals("BrokenDevice"):false
   }
   @Override
   Object run() {
          sleep(rand.nextInt(20000)) //Sleep for a random number of seconds between [0-20]
   }
}
```

- Ne communiquent pas entre eux directement
- Partagent leur état
- Compilés dynamiquement

Fonctionnement et Utilités




```
class DeviceDelayFilter extends ZuulFilter {
   def static Random rand = new Random()
    @Override
    String filterType() {
     return 'pre'
    }
   @Override
   int filterOrder() {
     return 5
   }
   @Override
   boolean shouldFilter() {
      return  RequestContext.getRequest().getParameter("deviceType")?equals("BrokenDev
   }
   @Override
   Object run() {
         sleep(rand.nextInt(20000)) //Sleep for a random number of seconds between [0-20]
   }
}
```

Succès ?

- Facile à utiliser
- Résultats immédiats
- Adopté à grande échelle
- Clients heureux

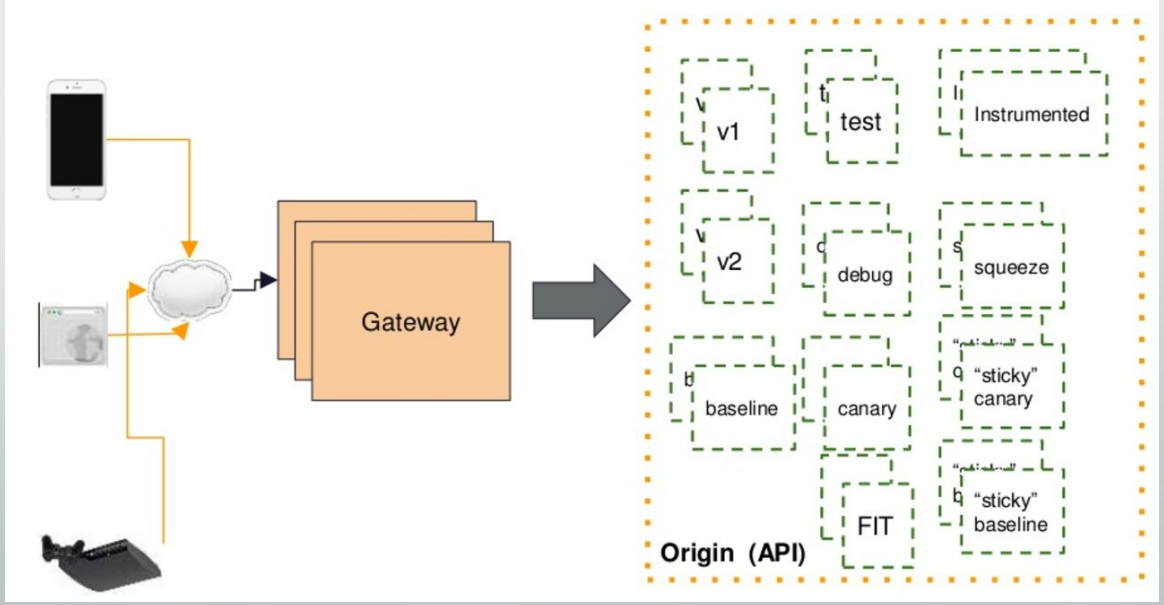
- Business Logic
- Système moins élastique
- Clarté

Stratégie

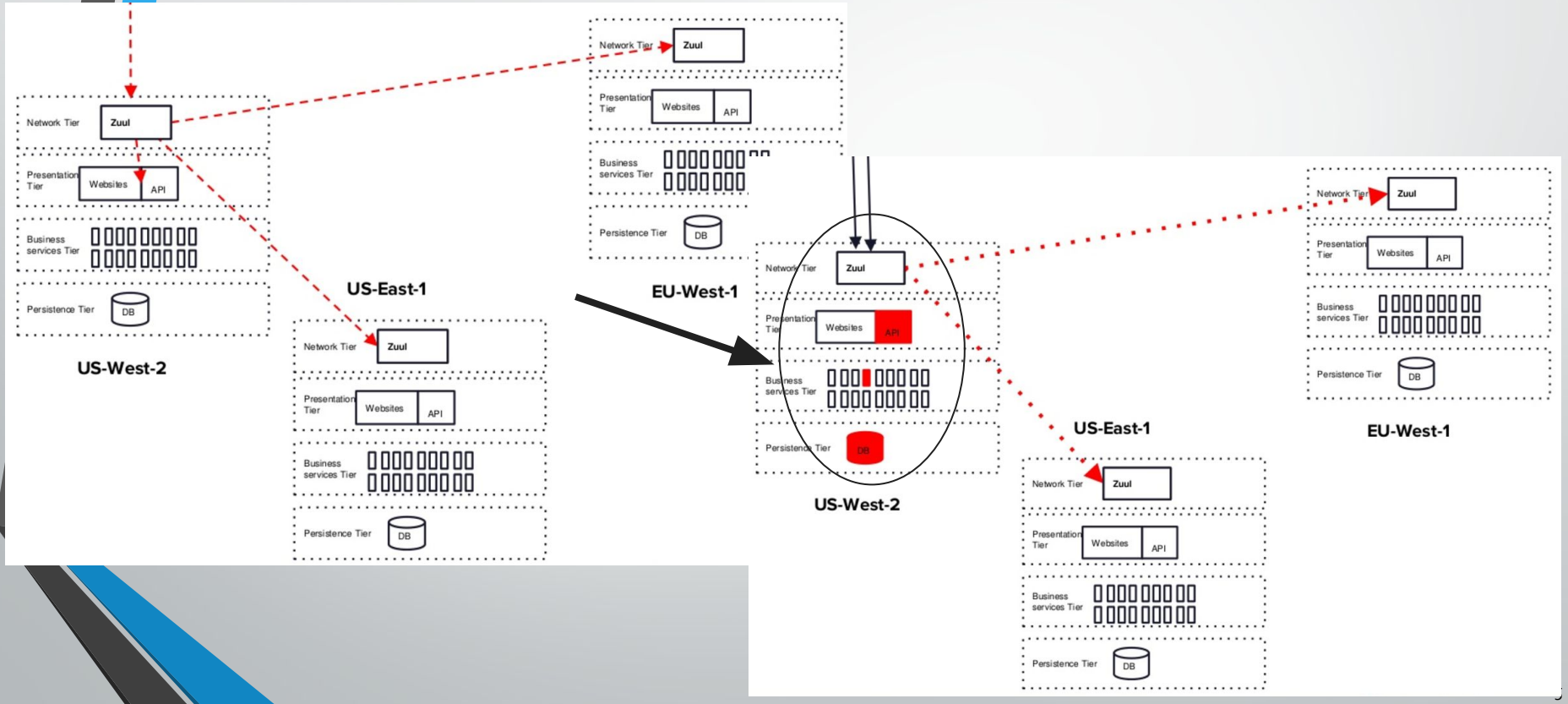
- Routing créatif et dynamique
- Mise en forme du trafic
- Vitesse de réaction
- Vue interne
- Focus sur la livraison, éviter les erreurs
- Stress Testing

Routing créatif

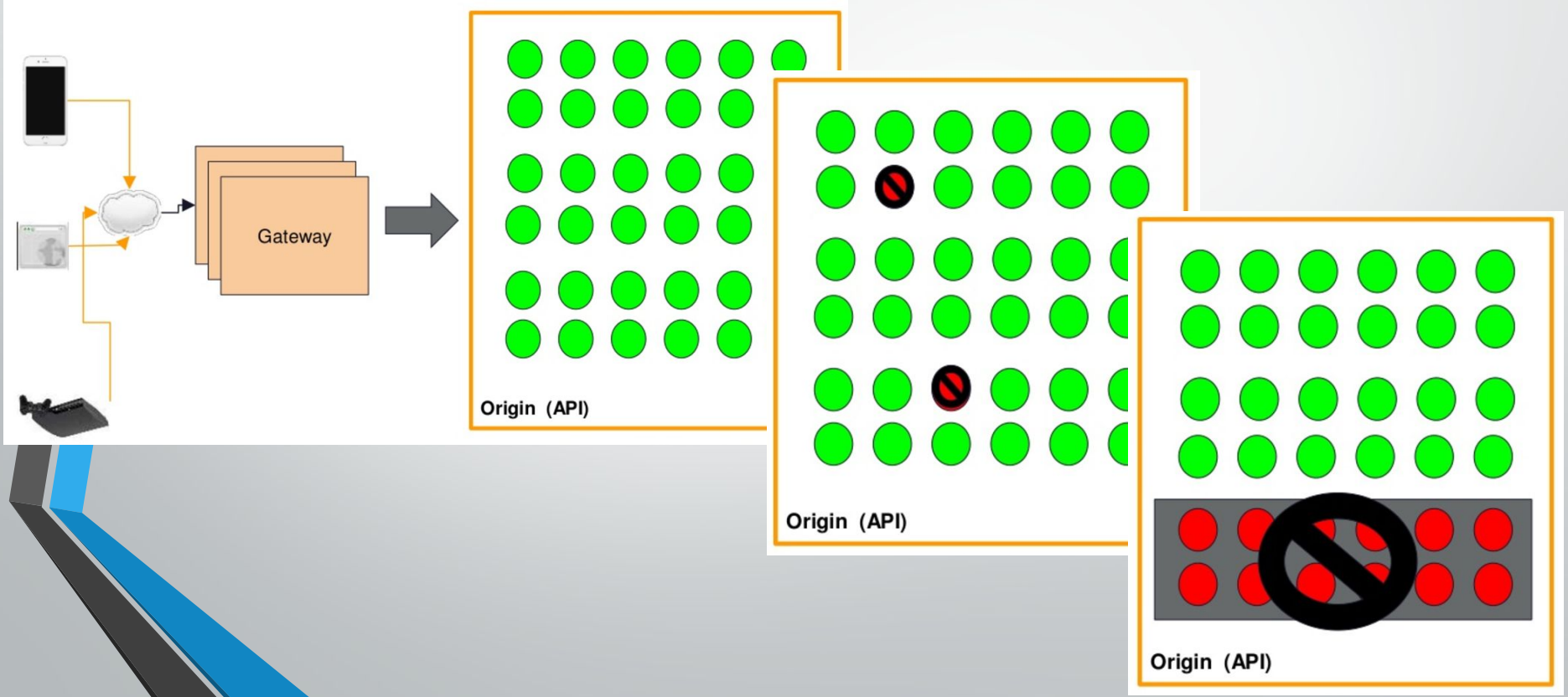
- Différentes versions implémentées en même temps



Mise en forme du trafic



Load Balancing intelligent



Zuul 2.0

Zuul 2.0

- Pas encore accessible en Open Source
- Non Bloquant, Asynchrone
  - Pas de threads, moins gourmand lors des connexions
  - Plus complexe à débugger
- Connexions persistantes
- Tous types de requêtes (HTTP/2, Websocket, etc.)

Bilan

Sources

- https://spring.io/blog/2016/12/01/springone-platform-2016-replay-netflix-s-edge-gateway-using-zuul
- https://github.com/Netflix/zuul