

# Architecture logicielle de Gmail

## Sommaire :

1. Présentation de Gmail
2. Architecture logicielle de Gmail
3. Historique d'architecture de Gmail
4. Conclusion

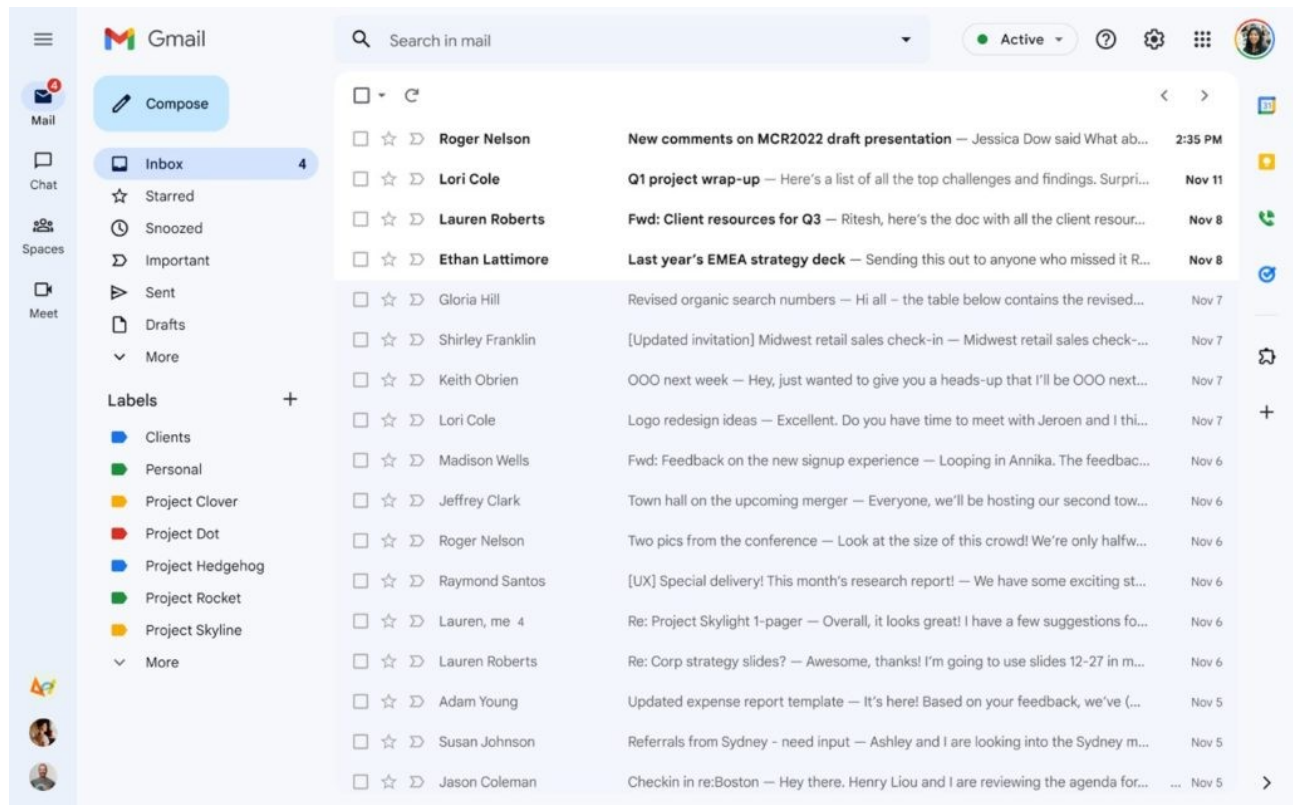


## 1. Présentation de Gmail

Gmail est un service de messagerie électronique créé en 2004 par Google. Il est aujourd'hui l'un des services de messagerie les plus utilisés par des milliards d'utilisateurs à travers le monde. Gmail a été créé par Google pour apporter une solution de messagerie électronique propre à Google et montrer sa supériorité en créant une plateforme beaucoup plus réussie et répondant aux besoins de la population face à ces concurrents qui avaient proposé une solution bien avant mais qui avait beaucoup de manques comme Microsoft ou Yahoo.

Le succès de Gmail s'explique déjà car c'est gratuit pour les particuliers même si on peut souscrire à un abonnement pour avoir plus de stockage en cloud même si les 15 GO offerts par Google sont largement suffisants, l'interface de Gmail est très facile à utiliser donc permet de gérer facilement ses courriers électroniques. L'intégration de Gmail dans l'écosystème Google est un facteur de succès car il faut obligatoirement avoir une adresse Gmail pour accéder aux autres services de Google et permet une harmonie avec les différentes plateformes comme Google Docs (éditeur de texte en ligne), Meet (organiser des visio en ligne), etc. Les outils utilisables sur Gmail permettent à l'utilisateur de faire des recherches précises donc cela offre un confort non négligeable. Tous ce que j'ai cité jusqu'à maintenant ce sont des éléments de confort pour l'utilisateur mais le fait que Gmail ait des milliards d'utilisateurs est surtout dû au fait que Gmail est très sécurisé grâce à l'authentification et pleins d'autres outils mis en place par Gmail pour éviter le phishing. Pour moi le plus gros point fort de Gmail c'est que c'est accessible depuis partout via un client de messagerie, une application mobile ou avec un navigateur web.

L'interface de Google sur Ordinateur :



## 2. Architecture logicielle de Gmail

Tous d'abord nous devons définir ce que c'est une architecture logicielle : l'architecture logicielle décrit d'une manière symbolique et schématique les différents éléments d'un ou de plusieurs systèmes informatiques, leurs interrelations et leurs interactions. (wiki).

### Architecture de Microservices

L'architecture de microservices décompose une application en une collection de services indépendants, chacun gérant une fonction spécifique. Chaque microservice peut être développé, déployé et mis à l'échelle indépendamment des autres.

Fonctionnement dans Gmail :

- Microservices : Gmail est composé de multiples microservices, chacun responsable d'une fonctionnalité spécifique, comme la gestion des emails, la recherche, le filtrage anti-spam, la gestion des contacts, etc.
- Communication Inter-Services : Les microservices communiquent entre eux via des API.

Elle utilise une structure composée de plusieurs couches :

## **Couche externe**

Cette couche est celle qui interagit directement avec les utilisateurs, pour leur parler dans leur langue. Il est responsable de la gestion de l'interface utilisateur graphique et de la présentation des courriels.

## **Couche Noyau**

Cette couche effectue tous les traitements nécessaires pour le fonctionnement de Gmail. Elle assure la logique d'application et le traitement des données.

## **Couche de Stockage**

Cette couche interagit avec les bases de données où sont stockés les courriels et les documents joints. Elle assure la persistance et la récupération des données.

En utilisant ses deux architectures cela permet à Gmail d'être fiable et de bien fonctionner.

## **3. L'historique de l'architecture de Gmail**

L'architecture de Gmail a évolué significativement depuis son lancement en 2004. Voici un aperçu de son évolution au fil des années :

Phase initiale (2004-2006) :

### **1. Architecture Monolithique :**

- À ses débuts comme la plupart des entreprises pour une nouvelle application, Google a opté pour une architecture monolithique qui consiste à mettre tout dans un bloc car c'est beaucoup plus simple car au début l'application on met uniquement en place les bases et surtout pour l'époque où on n'a pas autant d'architecture qu'aujourd'hui.

### **2. Utilisation intensive d'AJAX :**

- Gmail a été l'un des premiers à utiliser la technologie AJAX (Asynchronous JavaScript and XML) pour offrir une expérience utilisateur riche et interactive

directement dans le navigateur web car avant cela il avait presque que du JAVA et HTML qui été utilisé donc l'expérience utilisateur n'été pas sympa car cela nécessité beaucoup de rafraîchissement de pages.

Une évolution vers une architecture distribuée (2006-2010) :

#### 1. Les microservices:

- L'augmentation de la taille et de la complexité de son système a poussé Google a adopté une architecture à base de microservices. Il a décomposé l'application en plusieurs parties, tels que le service de gestion des courriels, le service de recherche, le service anti-spam, et ainsi de suite.

#### 2. Le stockage distribué:

- En réaction au grand nombre de petites pièces de données générées par les habitudes d'utilisation de Gmail, Google a créé de robustes systèmes de stockage distribué pour gérer vastes quantités dans les systèmes exobytes générés

Stabilité et amélioration continue (2010 à aujourd'hui) :

#### 1. Intégration avec l'infrastructure Cloud de Google :

- Gmail fait partie intégrante de l'écosystème cloud de Google, utilisant des services et des infrastructures de calcul, de stockage et de mise en réseau distribués pour assurer la scalabilité et la disponibilité.

#### 2. Adoption de Kubernetes :

- Google utilise Kubernetes pour gérer et orchestrer ses conteneurs Docker. Bien que les détails spécifiques à Gmail ne soient pas divulgués, l'utilisation de Kubernetes facilite la gestion de microservices à grande échelle.

#### 3. Focus sur la sécurité et la confidentialité :

- Gmail a renforcé ses mesures de sécurité au fil du temps, en mettant l'accent sur la protection des données utilisateur et en introduisant des fonctionnalités comme le chiffrement des e-mails de bout en bout car aujourd'hui Gmail est utilisé par des

milliards de personnes donc Gmail se doit d'être sécurisé car les données des utilisateurs et strictement personnel donc les utilisateurs doivent être protégés.

#### 4. Optimisation pour la performance :

- Aujourd'hui avec nos smartphones et appareils on peut tous avoir à la minute donc pareil pour les courriers électroniques pour les avoir quasi instantanément, Google continue à optimiser Gmail pour plus de puissance et vitesse et réactivité de l'interface.

#### **4. Conclusion**

Depuis son lancement en 2004, l'architecture de Gmail a subi des transformations importantes de l'architecture monolithique à une architecture distribuée et sophistiquée faisant appel aux microservices. Grâce à cela, l'application a fourni des services à des millions d'utilisateurs, tout en maintenant les performances à un niveau élevé et en renforçant la sécurité. Cela est possible grâce à la structure en couches et les microservices, qui assurent le niveau approprié d'organisation et de flexibilité :

Structure en couches : Divise les responsabilités en présentation, logique métier et gestion des données afin de faciliter la maintenance et l'évolution du système.

Microservices Permet la modularité et la scalabilité, chaque fonctionnalité étant implémentée en tant que service indépendant pour fournir suffisamment de flexibilité et de résilience.

Cette combinaison d'architectures permet à Gmail de fournir un service de messagerie robuste, performant et évolutif, capable de répondre aux besoins actuels et futurs des utilisateurs tout en intégrant des technologies avancées de stockage, de traitement et de sécurité des données.

Sources : wikipédia et cours