



# Implémentation d'architectures IA sur AWS



# Aperçu de la présentation

1. Introduction au Cloud AWS et au Serverless
2. Présentation de Necko
3. Présentation de l'IA sur AWS
4. Démonstration - RAG en IaC
5. Conclusion et contacts

# AWS Cloud

# Qu'est ce que le Cloud ?

Le cloud computing est la mise à disposition de ressources informatiques à la demande via Internet, avec une tarification en fonction de votre utilisation.

source: <https://aws.amazon.com/fr/what-is-cloud-computing/>



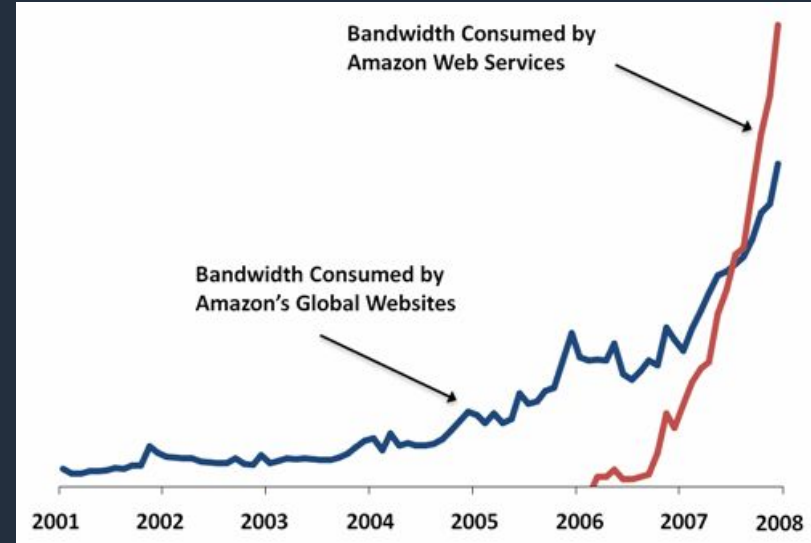
# Origine de AWS

Amazon débute avec la vente de livres, puis standardise son infrastructure face à la diversification.

Dès 2003, ses ingénieurs développent des services réutilisables, aboutissant à S3, SQS et EC2.

AWS est lancé en 2006 et devient, dès 2010, un leader du cloud.

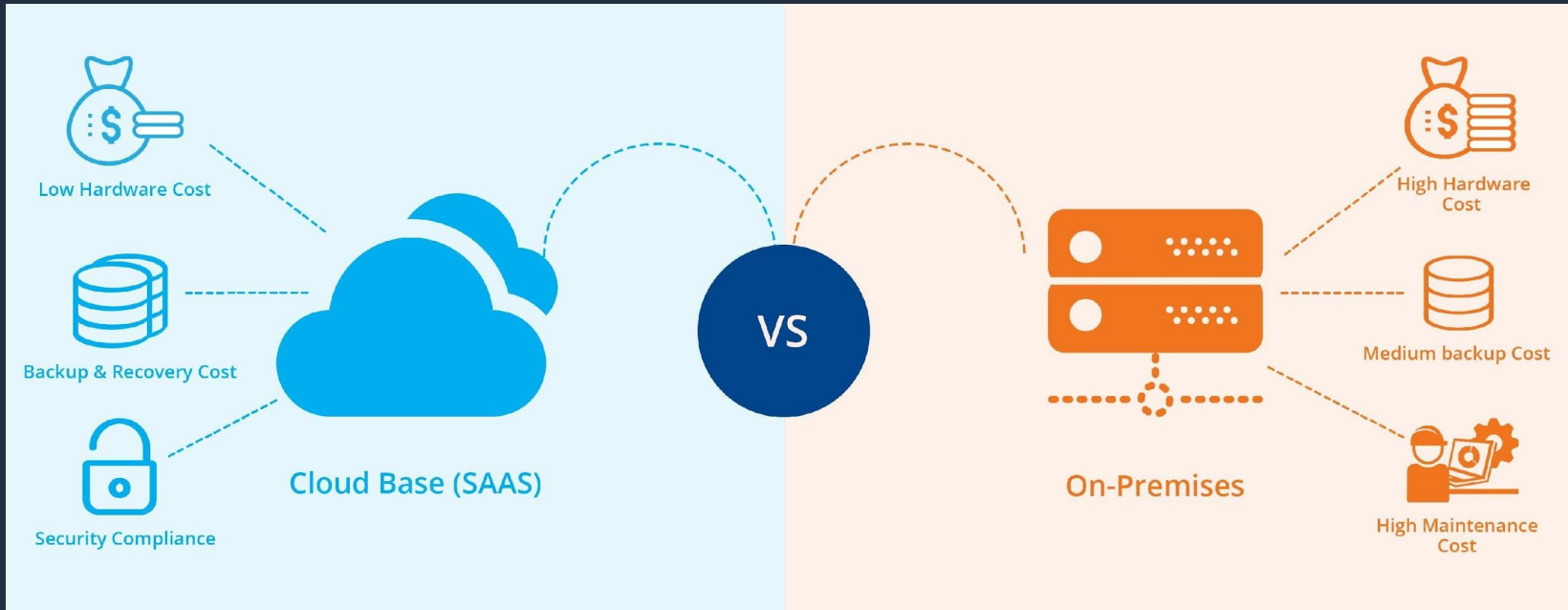
Depuis 2015, AWS génère plus de bénéfices qu'Amazon Commerce, avec ~20% du CA et 50% des profits.



Source:

<https://aws.amazon.com/blogs/aws/lots-of-bits/>

# Cloud vs On-Premises



# Amazon Web Services

- Pas de frais initiaux
- Paiement à l'utilisation
- Accélération du Time to Market et Agilité
- Scaling automatique
- Infrastructure self-service



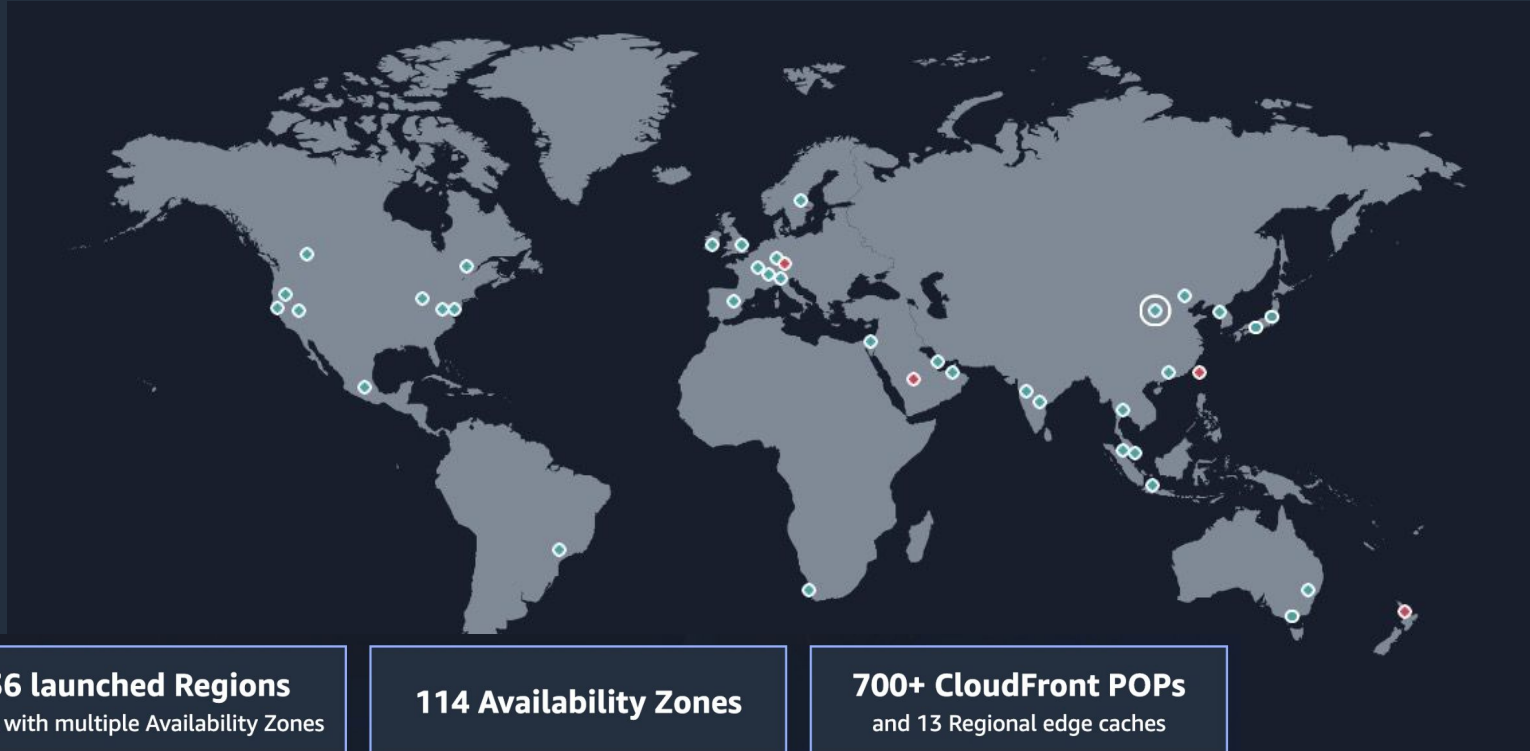
# AWS leader depuis 15 ans

Figure 1: Magic Quadrant for Strategic Cloud Platform Services



Gartner.

# AWS Infrastructure Globale



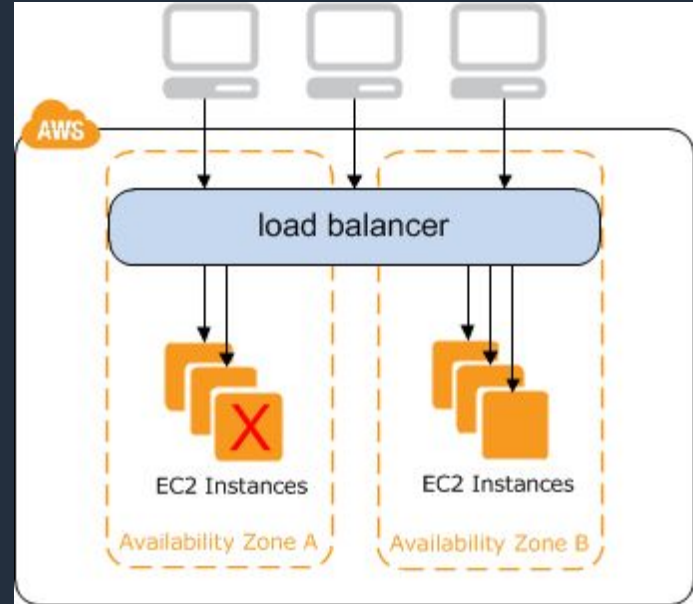
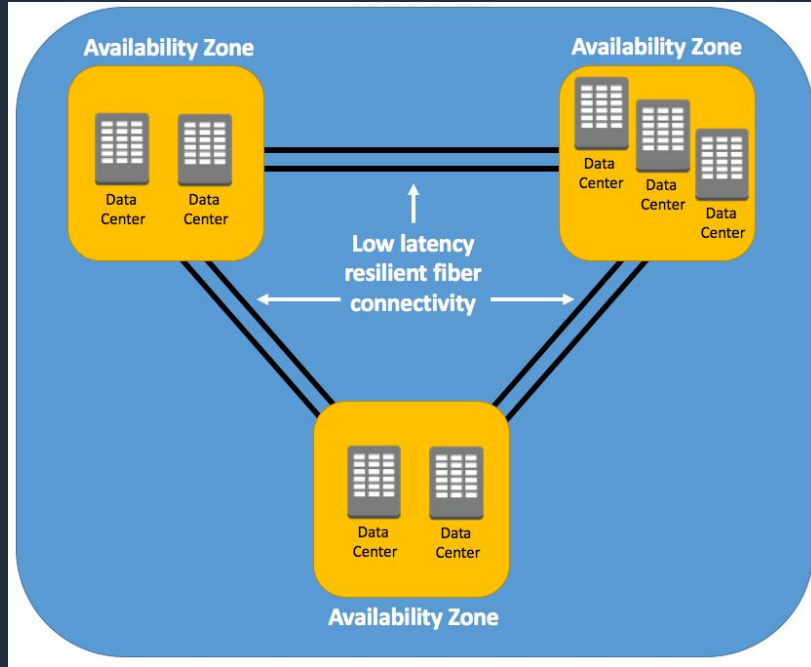
**36 launched Regions**  
each with multiple Availability Zones

**114 Availability Zones**

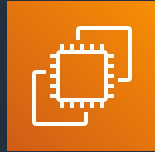
**700+ CloudFront POPs**  
and 13 Regional edge caches

Source : <https://aws.amazon.com/about-aws/global-infrastructure/>

# Région divisée en Availability Zone



# Serveurs virtuels dans le cloud



AWS Elastic  
Cloud Compute  
(EC2)

- Stockage modulable
- Disponibilité
- Démarrage rapide
- Instance différents
- Payez à l'utilisation

# La tendance dans les entreprises



Emergence de services managés sur le Cloud

Déléguons toute la partie infrastructure

---



Se concentrer sur la spécificité de l'entreprise

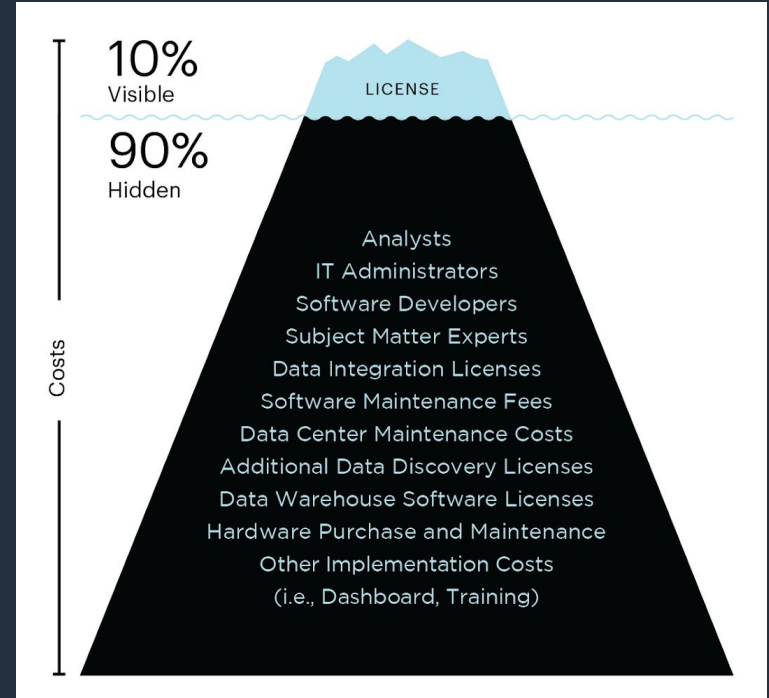
Ne faire que de la logique business

---

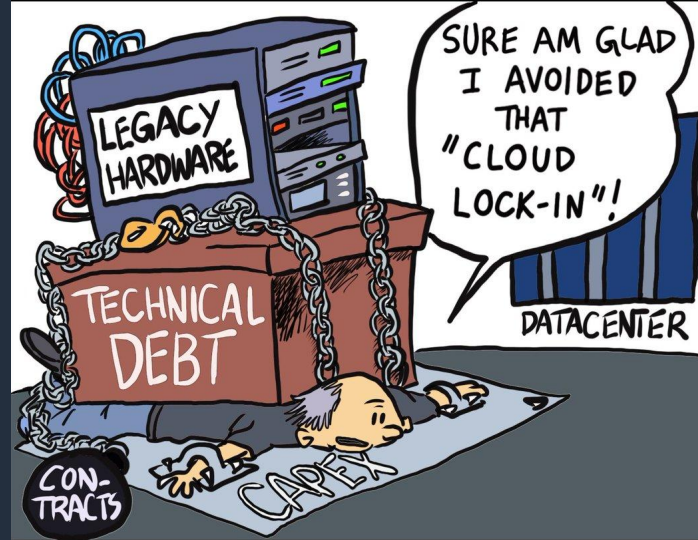


# Serveurs = problèmes

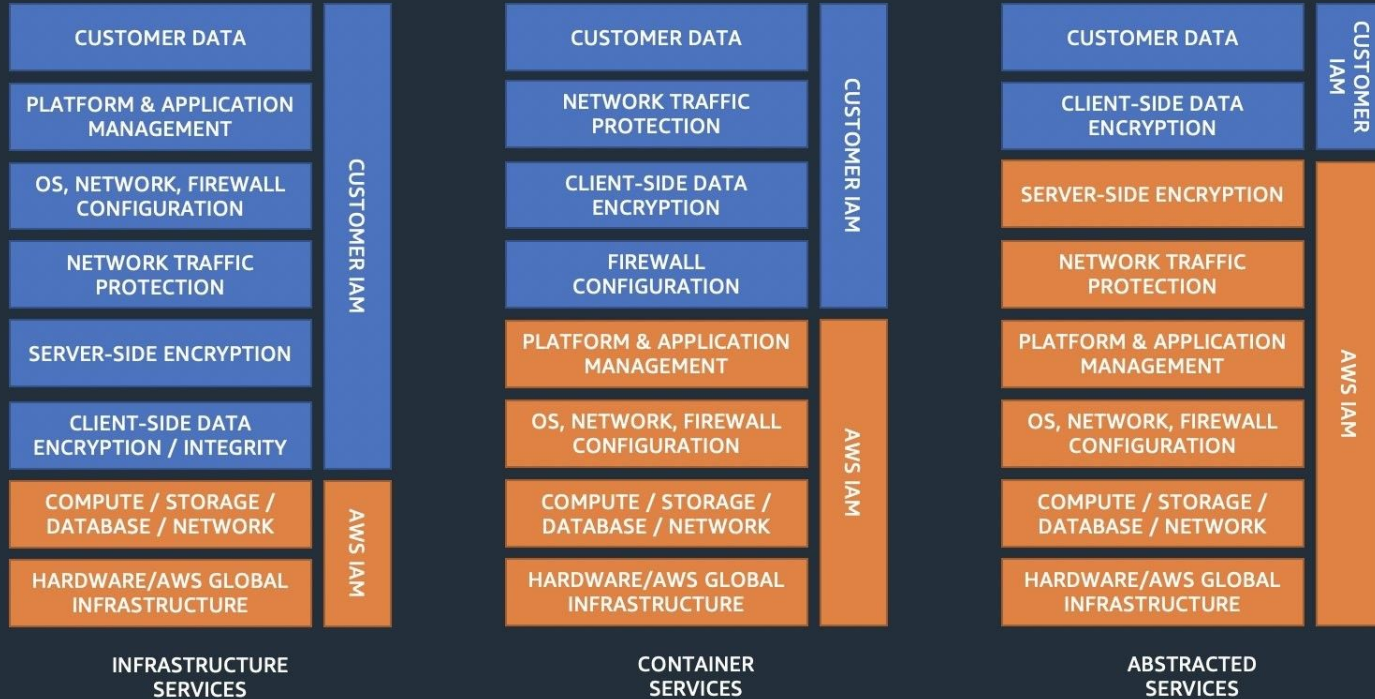
- Coûts cachés
- Ralentit l'agilité
- Maintenir infrastructure != maintenir logiciel



# Serveurs = problèmes



# De on-premise vers serverless

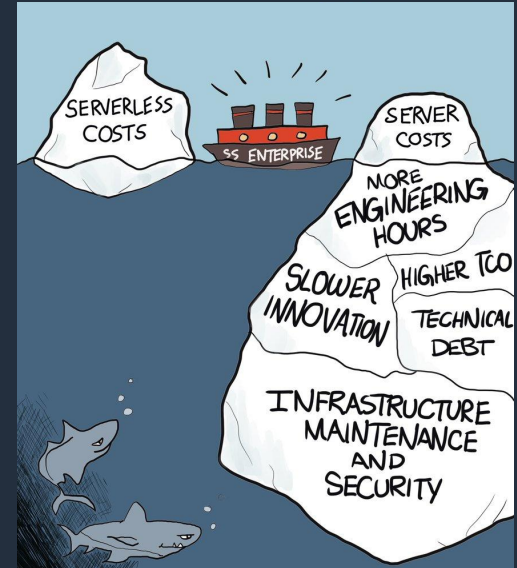


Source:

<https://aws.amazon.com/blogs/industries/fsi-service-spotlight-amazon-elastic-container-service-ecs-with-aws-fargate/>

# Managed service = Serverless

- Coût à l'utilisation
- Garantie de fiabilité
- Alerting: une alerte = erreur dans la logique
- Petit services



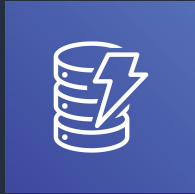
# Gestion des identités et des accès



AWS IAM

- Gestion accès AWS
  - Utilisateur
  - Service
- Peut être fédérée avec d'autres systèmes (Microsoft Active Directory)

# Database pour documents



Amazon  
DynamoDB

- NoSQL
- Interface JSON
- Accès en ms

# Database relationnelle serverless



Amazon  
Aurora  
Serverless V2

- SQL
- Scale to zéro
- Accès en ms
- Petite soeur: Aurora DSQL

# S3: Simple Storage Service



Amazon S3

- Stockage de fichiers
- Max 5 Pb/fichier
- Scale (jusqu'à 55000 requêtes/s)



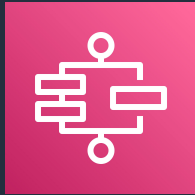
# Appel de fonction



AWS Lambda

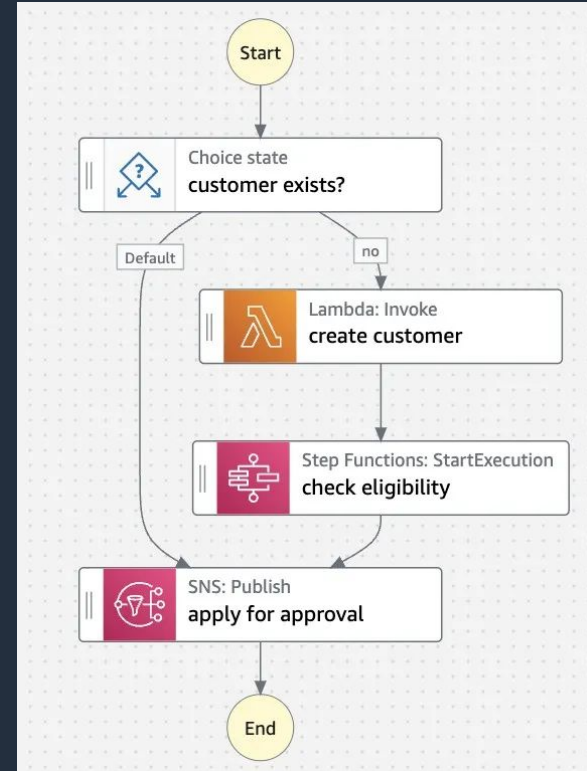
- Déclenché par un évènement
  - Cron
  - par action sur S3, Dynamo...
- Scaling automatique
- Exécute du code
- Coût = € x secondes x mémoire alloué

# Orchestration de services



AWS Step  
Functions

- Machine à états
- outil de visualisation de workflow



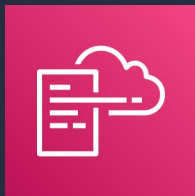
# Infrastructure as Code

**Objectif:** Décrire l'infrastructure sous forme de code

**Avantages:**

- Vue détaillée de l'implémentation
- Facile à redéployer la même infrastructure
- Permet de versionner le code (Git)
- Déploiement automatique
- Tests et vérifications automatique

# Créer et gérer des ressources à l'aide de modèles



AWS  
CloudFormation

- Template au format Yaml
- Rollback en cas de problèmes
- Suivi du *drift*
- Permet de faire un *diff*
- *Managé par AWS*

# AWS Cloud Development Kit

```
const table = new Table(this, id: 'table', props: {
  partitionKey: {
    type: AttributeType.STRING,
    name: 'id'
  }
});
const role = new Role(this, id: 'role', props: {
  assumedBy: new ServicePrincipal(ServicePrincipals.LAMBDA)
});
table.grantReadData(role);
```

- Génère du CloudFormation
- Simplifie les liens entre les ressource
- Debuggable
- Permet de créer des packages réutilisables

# Présentation de Necko



Nous accompagnons les entreprises dans leur voyage vers le cloud

Nous leur offrons:

- une expertise pointue sur les services et les produits AWS (mais aussi GCP et Azure)
- du développement software
- des services de consultance

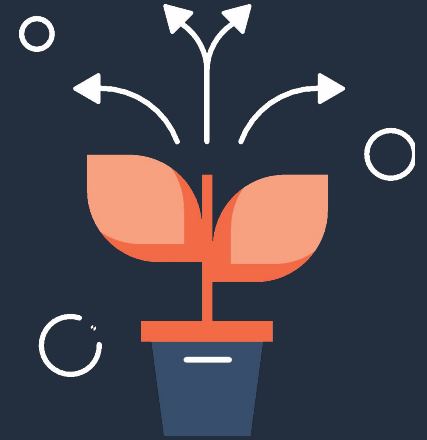
# Notre offre de services



Foundation



Migration



Optimisation



# En quelques mots

- Startup fondée en 2017
- Basée à Mons
- Partenaire cloud AWS depuis 2017, premier partenaire wallon du cloud AWS
- AWS Advanced Partner, unique en Wallonie
- AWS WAR Partner
- Expertise spécifique AWS



- AWS Lambda Delivery
- AWS CloudFormation Delivery
- Well-Architected Partner Program



Prix Mercure 2018  
Ville de Mons  
Jeune Entreprise



Inno pépites 2018  
LME  
Catégorie entreprise IT



AWS Service Delivery  
AWS Lambda  
AWS CloudFormation

# WAR: Well Architected Framework



# L'équipe

- 18 collaborateurs (pour le moment... On recrute 😁)
- Profils architectes et développeurs
- Bientôt 100 certifications AWS



# L'équipe

- 18 collaborateurs (pour le moment... On recrute 😁)
- Profils architectes et développeurs
- Bientôt 100 certifications AWS



# Business Case

- Studio radio online
- Transfert d'un logiciel existant basé sur PC vers une nouvelle solution basée sur le cloud computing sur AWS.
- Solution SaaS complète et performante
- Connexion sécurisée, de n'importe où, à tout moment permettant une diffusion en direct avec une faible latence
- Équipe sur le projet: 1 Cloud Developer, 1 Solution Architect

# Nos clients

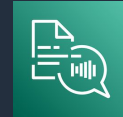


# L'IA sur AWS

# Vue d'ensemble

## Services AWS

- APIs Serverless
  - Rekognition
  - Transcribe
  - Polly
  - Textract
  - ...
- Machine learning
  - Sagemaker
- IA Générative
  - AWS Bedrock





# Vue d'ensemble

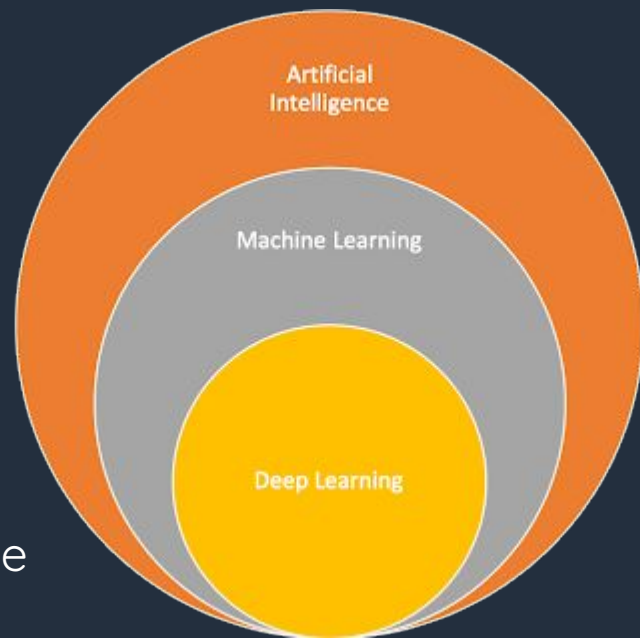
## Amazon Bedrock

- Focus sur l'IA générative
  - Techniques exploitant les modèles génératifs
  - Tour d'horizon de Bedrock
- Démonstration d'une solution complète
  - Utilisation de l'infrastructure as Code
  - Déploiement d'un RAG



# Petits rappels

- Intelligence artificielle
  - Domaine d'étude
- Machine Learning
  - Sous domaine de l'IA
  - Algorithmes d'apprentissage automatique
- Deep Learning
  - Sous domaine du machine learning
  - Architectures basées sur les réseaux de neurones



# Petits rappels

- Large Language Models (LLMs)
  - Modèle probabiliste sur le langage naturel
- Reinforcement learning (RL)
  - Technique d'apprentissage basée sur récompenses/pénalités

Chatbot = LLM + RL (HF)





































# Pourquoi choisir AWS pour l'IA ?

- Services gérés
  - Utilisation à la demande
  - Idéal pour l'inférence
- Serveurs dédiés (Inferentia, A100, H100, ...)
  - Idéal pour entraîner des modèles
- Intégration avec le reste de votre infrastructure
  - Et l'Infrastructure as Code !

# Techniques exploitant les modèles génératifs

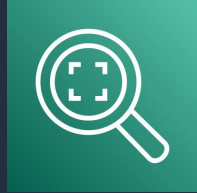
- Fine-tuning
  - Spécialisation d'un modèle sur un dataset plus petit
- Retrieval Augmented Generation (RAG)
  - Chatbot combiné à une source de données
- Agents
  - Chatbot pouvant exécuter des commandes
- Prompt Engineering
  - Trouver le meilleur prompt pour un problème précis

# Techniques exploitant les modèles génératifs

	Continued pre-training	Fine-tuning	RAG	Agent
Data quantity	  	 	 	
Data structure				API
Pickup changes				 
Accuracy				 
Data liveness		 	 	 
Cost	   	  	 	

Source: AWS re:Invent 2024 - The art of transforming foundation models into domain experts (DEV301)

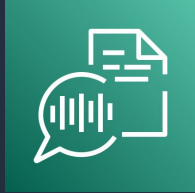
# Vision par ordinateur



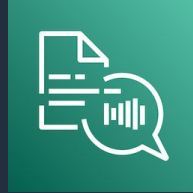
Rekognition

- API Serverless
- Reconnaissance d'images
  - Détection de visages
  - Reconnaissance d'objets
  - Détection de célébrités
  - ...

# Services paroles et textes



Transcribe  
Speech-to-text



Polly  
Text-to-speech



Textract  
Extract text from  
images



# Plateforme pour le machine learning



Sagemaker

- Gère le cycle de vie des modèles :
  - Entraînement
  - Validation
  - Déploiement
- Environnements pour les Data Scientists :
  - Machine dédiées
  - Notebooks en ligne
  - Dépendances préinstallées

# IA Générative



AWS Bedrock

- Catalogue de modèles
  - Claude (Anthropic)
  - Mistral (Mistral AI)
  - Llama (Meta)
  - Stable Diffusion (Stability AI)
  - DeepSeek-R1
- API Serverless
- Moderation (Guardrails)
- Outils de Prompt Engineering
- Outils d'évaluation

# IA Générative



AWS Bedrock

Mais aussi :

- Compréhension d'images / vidéos
- Génération d'images / vidéos
  - Stable Diffusion
  - Amazon Nova
  - Amazon Titan

# Catalogue de modèles

Amazon Bedrock

▼ Getting started

Overview

Providers

▼ Foundation models

Model catalog [New](#)

Marketplace deployments [New](#)

Custom models (fine-tuning, dist...)

Imported models

Prompt Routers [Preview](#)

▼ Playgrounds

Chat / Text

Image / Video

▼ Builder tools

Agents

Flows

Knowledge Bases

Prompt Management

▼ Safeguards

Guardrails

Watermark detection

▼ Inference and Assessment

Provisioned Throughput

Batch inference

Cross-region inference

Evaluations

Amazon Bedrock > Model catalog

Model catalog (186)

Discover Bedrock serverless or Marketplace models that best fit your use case. To get started using a serverless model, request access. For Marketplace models, subscribe and deploy.

▼ Filters

▼ Model collection

☐ Bedrock Marketplace (137) ⓘ

☐ Serverless (49)

▼ Providers

☐ AI21 Labs (5)

☐ Amazon (13)

☐ Anthropic (10)

☐ Arcee AI (6)

☐ BRIA AI (3)

☐ Camb.ai (1)

☐ Cohere (6)

☐ DeepSeek (7)

☐ EvolutionaryScale, PBC (1)

☐ Gretel (1)

[Show 10 more](#)

▼ Modality

☐ Audio (1)

☐ Embedding (5)

☐ Image (4)

☐ Multimodal (1)

☐ Text (125)

☐ Text Vision (10)

☐ Video (1)

▼ Serverless model access status

☐ Access granted (29)

☐ Available to request (20)

▼ Spotlight

AI

Claude 3.7 Sonnet

Access granted

Text generation, Code generation, Rich text formatting, Agentic computer use

Serverless

AI

Nova Pro

Access granted

Agents, Chat optimized, Code generation, Complex reasoning analysis, Conversation, Mat...

Serverless

AI

DeepSeek-R1

Available to request

Fine Tuning, Distillation, Code generation

Serverless

Filter for a model

Most popular ▼

< 1 2 3 4 5 6 7 >

AI

Claude 3.7 Sonnet

Access granted

Text generation, Code generation, Rich text formatting, Agentic computer use

Serverless

AI

Claude 3.5 Haiku

Access granted

Agents, Chat optimized, Code generation, Complex reasoning analysis, Conversation, Mat...

Serverless

AI

Claude 3.5 Sonnet v2

Access granted

Agents, Chat optimized, Code generation, Complex reasoning analysis, Conversation,...

Serverless

AI

Claude 3.5 Sonnet

Access granted

Agents, Chat optimized, Code generation, Complex reasoning analysis, Conversation,...

Serverless

AI

Claude 3 Opus

Access granted

Image to text & code, multilingual conversation, complex reasoning & analysis

Serverless

AI

Claude 3 Haiku

Access granted

Image to text, conversation, chat optimized




Serverless

# Chat

ModeChat

Compare mode

ConfigurationsInfo

 **Claude 3.5 Sonnet** v1    
EU Anthropic Claude 3.5 Sonnet

▼ System prompts

Add system prompts

▼ Randomness and diversity

Temperature1

Top P0.999


Top K250

▼ Length

Maximum length2000




Input: 56Output: 615Latency: 19102 ms

Quelles sont les dernières innovations dans le domaine des modèles de langues ? Plus particulièrement les modèles entraînés par renforcement pour produire des chatbots tels que ChatGPT.

 Voici quelques-unes des innovations récentes les plus importantes dans le domaine des modèles de langage et des chatbots basés sur l'apprentissage par renforcement :

1. Modèles de plus en plus grands et performants : Des modèles comme GPT-4 atteignent des performances impressionnantes sur un large éventail de tâches, avec des capacités de raisonnement et de génération de texte très avancées.
2. Apprentissage par renforcement à partir de retours humains (RLHF) : Cette technique, utilisée notamment pour ChatGPT, permet d'affiner les modèles en fonction des préférences humaines, améliorant ainsi leur alignement avec nos attentes.
3. Techniques d'instruction fine (InstructGPT) : Permet d'améliorer la capacité des modèles à suivre des instructions précises et à générer des réponses plus pertinentes.
4. Modèles multimodaux : Intégration de capacités de traitement d'images, de vidéos ou de sons en plus du texte (ex: GPT-4 avec analyse d'images).
5. Amélioration de la mémoire à long terme : Développement de techniques permettant aux modèles de mieux retenir les informations sur de longues conversations.
6. Réduction des biais et amélioration de la sécurité : Efforts continus pour réduire les biais inhérents aux données d'entraînement et rendre les modèles plus sûrs et éthiques.
7. Optimisation de l'efficacité computationnelle : Recherches sur des architectures plus légères et efficaces pour réduire les coûts et l'empreinte carbone.
8. Personnalisation et adaptation : Développement de techniques permettant d'adapter rapidement les modèles à des domaines ou utilisateurs spécifiques.

Write a prompt. Press Shift + Enter to add a new line. Press Enter to generate a response.

Run

# Prompt Engineering

Amazon Bedrock

▼ Getting started

Overview

Providers

▼ Foundation models

Model catalog [New](#)

Marketplace deployments [New](#)

Custom models (fine-tuning, dist...)

▼ Playgrounds

Chat / Text

Image / Video

▼ Builder tools

Agents

Flows

Knowledge Bases

Prompt Management

▼ Safeguards

Guardrails

▼ Inference and Assessment

Provisioned Throughput

Batch inference

Cross-region inference

Evaluations

User guide [?](#)

Bedrock Service Terms [?](#)

Compare variants Info

Exit comparison

▶ Run all

Original prompt

Save ▶ Run

Models ▼ Claude 3.5 Son... v1 ⓘ EU Anthropic Claude 3.5 Sonnet

≡ ×

Prompt

▶ System instructions - Optional

▶ Tools Configuration - Optional

This is my first text prompt. Please summarize the {{topic}} .

Add message

▼ Test variables (1)

The variables you define in the prompt are listed here. You can add values to test the prompt output in the test window. These values are temporary and aren't saved to your prompt.

Variable name ▼ Variable value - Optional ▼

topic transformer architecture.

Variant\_1

Replace original prompt ▶ Run

Models ▼ Claude 3.5 Son... v1 ⓘ EU Anthropic Claude 3.5 Sonnet

≡ ×

Prompt

▶ System instructions - Optional

▶ Tools Configuration - Optional

You are an expert in deep learning. Your task will be to explain specific deep learning concepts.

Please summarize the {{topic}}

Add message

▼ Test variables (1)

The variables you define in the prompt are listed here. You can add values to test the prompt output in the test window. These values are temporary and aren't saved to your prompt.

Variable name ▼ Variable value - Optional ▼

topic transformer architecture.

# Prompt Engineering



This is my first text prompt. Please summarize the transformer architecture. .



Certainly! I'd be happy to summarize the Transformer architecture for you.

The Transformer is a deep learning model introduced by Vaswani et al. in their 2017 paper "Attention Is All You Need." It has become a foundational architecture for many state-of-the-art natural language processing models. Here are the key points:

1. Core Mechanism: The Transformer relies entirely on self-attention mechanisms, dispensing with recurrence and convolutions used in previous sequence models.
2. Architecture:
  - Encoder-Decoder Structure: It consists of an encoder that processes the input and a decoder that produces the output.
  - Multiple Layers: Both encoder and decoder are composed of multiple identical layers.
3. Key Components:
  - Multi-Head Attention: Allows the model to focus on different parts of the input sequence for different purposes.
  - Feed-Forward Neural Networks: Applied to each position separately and identically.
  - Layer Normalization and Residual Connections: Used for stabilizing the learning process.



You are an expert in deep learning. Your task will be to explain specific deep learning concepts.

Please summarize the transformer architecture.



Certainly! The transformer architecture, introduced in the 2017 paper "Attention Is All You Need" by Vaswani et al., has become a cornerstone of modern natural language processing and has been adapted for various other tasks. Here's a summary of its key components and features:

1. Self-Attention Mechanism:
  - The core innovation of transformers
  - Allows the model to weigh the importance of different parts of the input sequence when processing each element
  - Replaces recurrent and convolutional operations found in earlier architectures
2. Encoder-Decoder Structure:
  - Encoder processes the input sequence
  - Decoder generates the output sequence
  - Both consist of multiple identical layers
3. Multi-Head Attention:
  - Allows the model to attend to information from different representation subspaces
  - Improves the model's ability to capture various aspects of the input

# Evaluation de modèles

## Metric summary

Evaluate overall performance using metrics (average score across all conversations). Closer to 1 is a higher score, closer to zero is a lower score. For example, closer to 1 for Correctness means more correct answers. You can define custom criteria to highlight any metrics that fall above or below a threshold.

[Define metric criteria](#)

## Quality metrics

These metrics assess the effectiveness of retrieving relevant information. For example, closer to 1 for Context relevance means more contextually relevant information on average than if the score was closer to zero. Click on the metric name for more info.

Context relevance

0.54

Context coverage

0.83

## Generation metrics breakdown [Info](#)

See metrics below to track and understand how Knowledge Base arrived at the output. Click on the chart for more details.

### Metrics

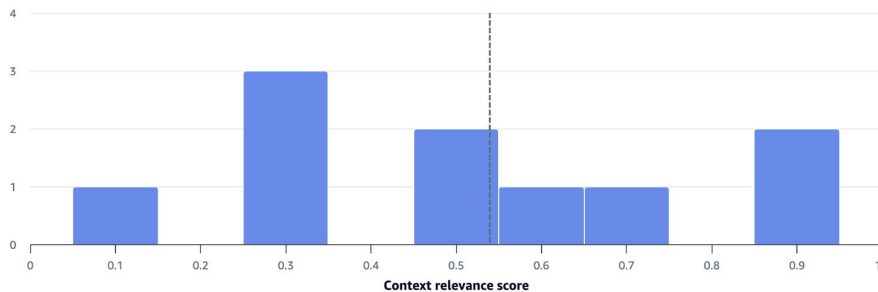
#### Quality metrics:

Context relevance  
Context coverage

### Context relevance

Measures how contextually relevant the retrieved texts are to the questions.

Number of example conversations



---- Avg score: 0.539

■ Value

Total: 10 prompts



# Démonstration d'un RAG

# Retrieval Augmented Generation (RAG)

**Principe:** Augmenter les capacités d'un modèle chatbot en lui donnant accès à une base de connaissances

**Avantages:**

- Permet d'aller au delà de la limite de contexte d'un modèle
- Diminue les problèmes d'*hallucinations* du modèle
- Facilite la recherche dans une grande collection de documents

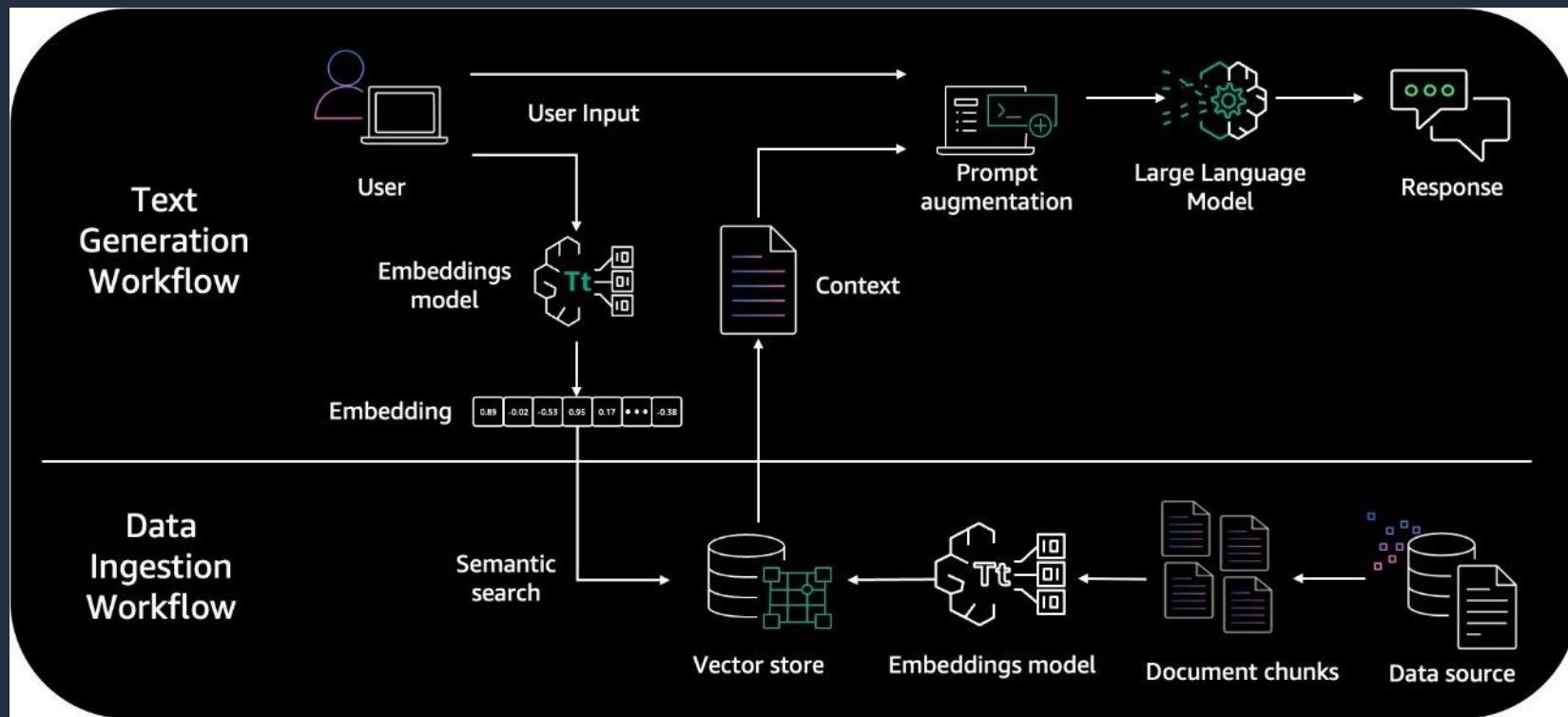
# Démonstration d'un RAG

**Objectif:** Créer un chatbot assistant sur des cours universitaires

**Fonctionnalités:**

- Recherche approfondie dans des slides et/ou syllabus
- Génération d'explications basées sur le contenu
- Déployé en Infrastructure as Code

# Architecture



Source:

<https://aws.amazon.com/blogs/machine-learning/evaluate-the-reliability-of-retrieval-augmented-generation-applications-using-amazon-bedrock/>

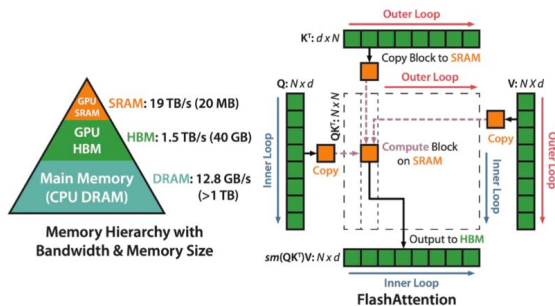
# Démonstration d'un RAG

**Query:** Quels algorithmes/techniques exploitent le hardware?

## Main Idea: Hardware-aware Algorithms

IO-awareness:

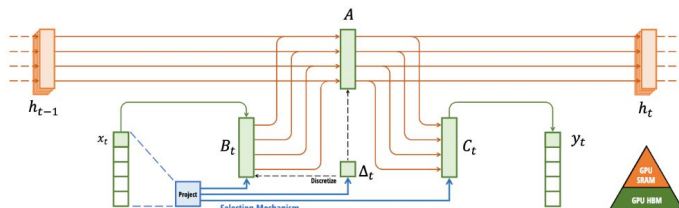
reducing reads/writes to GPU memory yields significant speedup



FlashAttention: **fast** and **memory-efficient** attention algorithm, with **no approximation**

State-space expansion:

expand recurrent states in SRAM only to avoid memory cost



Mamba: selective state-space model that **matches Transformers** on language model, with **fast inference** and **up to 1M context**

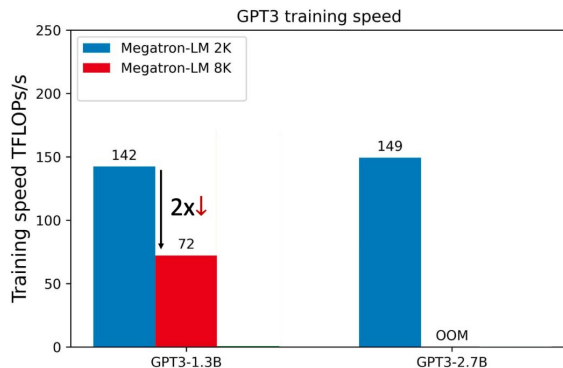
# Démonstration d'un RAG

**Query:** Comment la taille de contexte influence les performances?

## Efficiency is the Bottleneck for Modeling Long Sequences with Attention

Context length: how many other elements in the sequence does the current element interact with.

Increasing context length slows down (or stops) training



How to efficiently scale models to longer sequences?

# Démonstration d'un RAG

**Query:** Y a t-il des études au sujet de la propagande politique utilisant l'IA générative ?

- AI generated political propaganda is here, and it works. (Tomz et al., 2024) – and this study was done using propaganda generated from GPT-3! What about GPT-4/4o??

# Démonstration d'un RAG

**Query:** Quelles sont les différentes méthodes de raisonnement?

## What is Reasoning?

Using *facts* and *logic* to arrive at an answer

**Deductive Reasoning:** Use logic to go from premise to firm conclusion

Premise: All mammals have kidneys  
Premise: All whales are mammals  
Conclusion: All whales have kidneys

**Inductive Reasoning:** From observation, predict a likely conclusion

Observation: When we see a creature with wings, it is usually a bird  
Observation: We see a creature with wings.  
Conclusion: The creature is likely to be a bird

**Abductive Reasoning:** From observation, predict the most likely explanation

Observation: The car cannot start and there is a puddle of liquid under the engine.  
Likely Explanation: The car has a leak in the radiator



Voyons comment  
ça fonctionne!

# Environnement de travail




# Formations



# Expertise

# cdk <https://github.com/aws/aws-cdk> 11

Messages Add canvas Files +


 **Release notes from aws-cdk** Wednesday, March 5th

v2.182.0

Features

assertions: added getResourcesId method to Template (#33521) (a96b0f1)

autoscaling: add new HealthChecks for multiple health check types, including EBS and VPC\_LATTICE types (<a href="https://github.com/aws/aws-cdk/issues/31286"... (edited)

 **GitHub**

[Release v2.182.0 · aws/aws-cdk](#)

Features

assertions: added getResourcesId method to Template (#33521) (a96b0f1)

autoscaling: add new HealthChecks for multiple health check types, including EBS and VPC\_LATTICE types (#31286) (b3ed... (50 kB) ▾

New Release v2.182.0

**v2.182.0**

Features

- **assertions:** added getResourcesId method to Template (#33521) (a96b0f1),...



# Conclusion et contacts



# Envie de travailler avec nous ? Tu nous rejoins ?



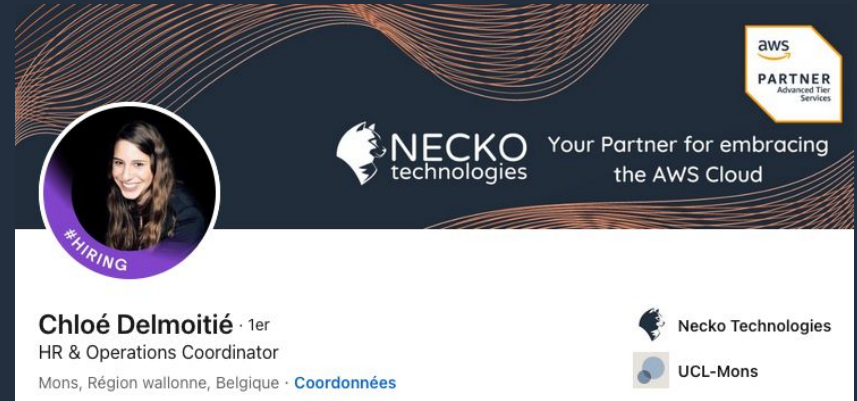
# Nous contacter




Necko Technologies




Necko Technologies





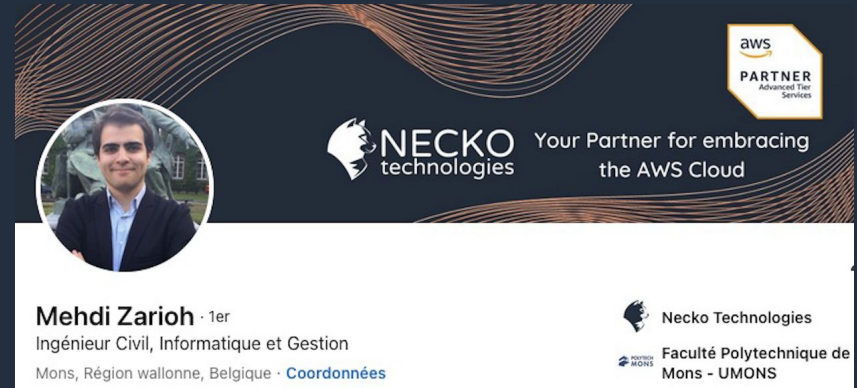
aws  
PARTNER  
Advanced Tier  
Services

 **NECKO**  
technologies Your Partner for embracing  
the AWS Cloud





**Chloé Delmoitié** · 1er  
HR & Operations Coordinator  
Mons, Région wallonne, Belgique · [Coordonnées](#)

 Necko Technologies  
 UCL-Mons





aws  
PARTNER  
Advanced Tier  
Services

 **NECKO**  
technologies Your Partner for embracing  
the AWS Cloud



**Mehdi Zarion** · 1er  
Ingénieur Civil, Informatique et Gestion  
Mons, Région wallonne, Belgique · [Coordonnées](#)

 Necko Technologies  
 Faculté Polytechnique de  
Mons - UMONS

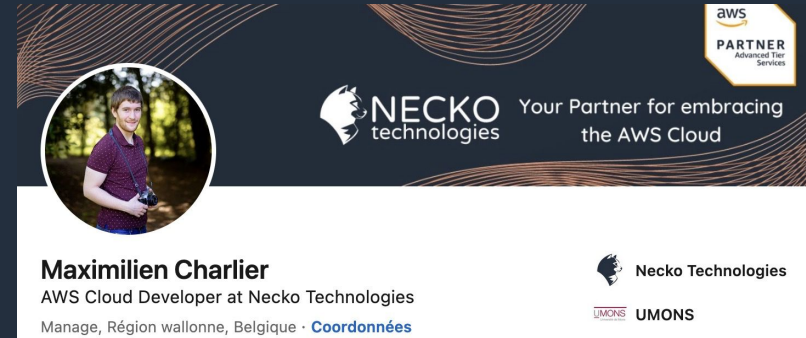
# Nous contacter




Necko Technologies



Necko Technologies

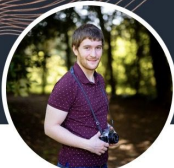


aws  
PARTNER  
Advanced Tier  
Services




NECKO  
technologies


Your Partner for embracing  
the AWS Cloud



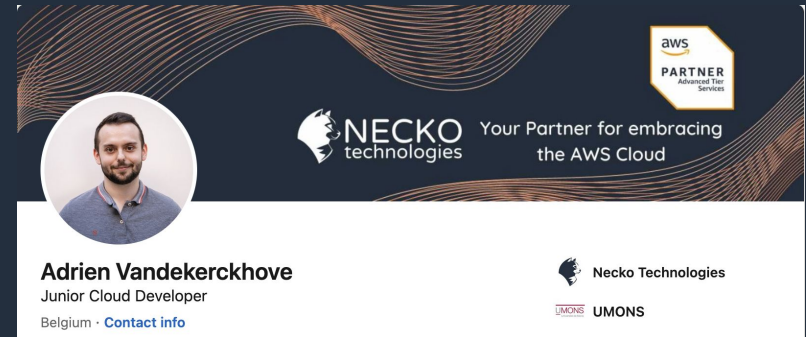
**Maximilien Charlier**  
AWS Cloud Developer at Necko Technologies  
Manage, Région wallonne, Belgique · [Coordonnées](#)




Necko Technologies



UMONS




aws  
PARTNER  
Advanced Tier  
Services




NECKO  
technologies


Your Partner for embracing  
the AWS Cloud



**Adrien Vandekerckhove**  
Junior Cloud Developer  
Belgium · [Contact info](#)



Necko Technologies



UMONS



# Merci pour votre attention!

Posez vos questions  
maintenant ou n'hésitez  
pas à nous contacter !

[www.necko.tech](http://www.necko.tech)



[contact@necko.tech](mailto:contact@necko.tech)

