Power Saturday

14 et 15 juin 2019, Paris

#SQLSatParis

3 communautés pour partager, échanger et apprendre

Club Power BI

GUSS

aOS

Power BI, Data, IA, Power Platform, Office 365, SharePoint, etc.

Power Saturday

SQL 2019 Big Data

Arian PAPILLON / Julien PIERRE

Merci à nos sponsors

Gold

Microsoft

Exakis Nelite
MagellanPartners

Silver

AZEO
talents & technology

Bronze

KPI consulting

Conseil IT

http:// PowerSaturday.com

Julien PIERRE

Microsoft Most Valuable Professional
Data Platform

infeeny | econocom

Microsoft

(…prochainement)

Microsoft CERTIFIED Solutions Expert Data Platform

Microsoft CERTIFIED Trainer

Arian PAPILLON

Microsoft Most Valuable Professional
Data Platform

Datafly

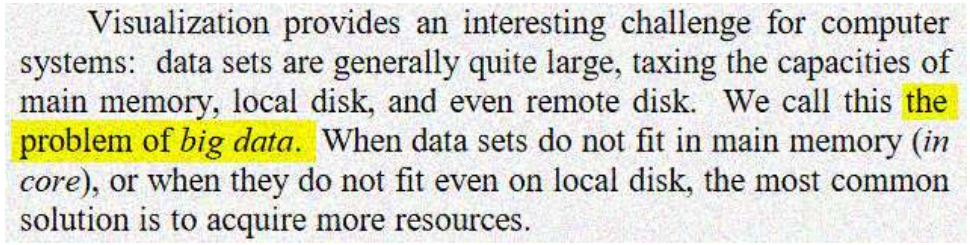
- www.datafly.fr
- blog.datafly.pro
- www.mssql.fr
- www.youtube.com/datafly

Microsoft CERTIFIED Solutions Expert Data Platform

Microsoft CERTIFIED Solutions Expert Data Management and Analytics

Microsoft CERTIFIED Trainer

Big Data

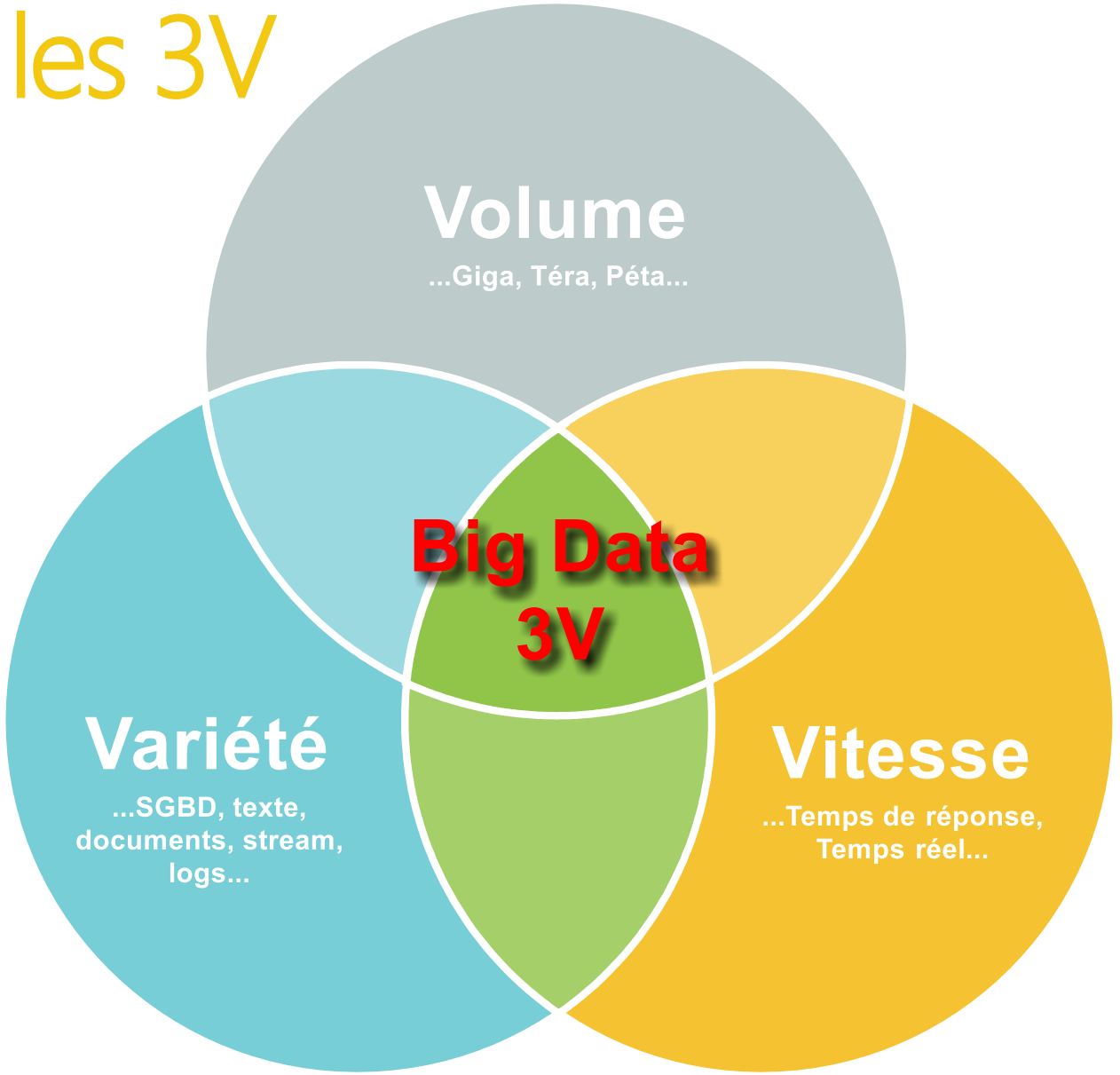


Visualization provides an interesting challenge for computer systems: data sets are generally quite large, taxing the capacities of main memory, local disk, and even remote disk. We call this the problem of *big data*. When data sets do not fit in main memory (*in core*), or when they do not fit even on local disk, the most common solution is to acquire more resources.

La première fois qu'on parle de « Big Data » (1997)

- Explosion des volumes de données numériques
  - Volumes massifs, bases de données géantes
  - Web, e-commerce, logs, IOT, …
- Au delà des capacités des SGBDR
  - Problématiques spécifiques de stockage, recherche, d'analyse et de visualisation
  - Formats de données multiples (structuré/non structuré)
  - Solutions de bases de données distribuées, parallélisation massive
- Différent de la BI traditionnelle

Big Data : les 3V

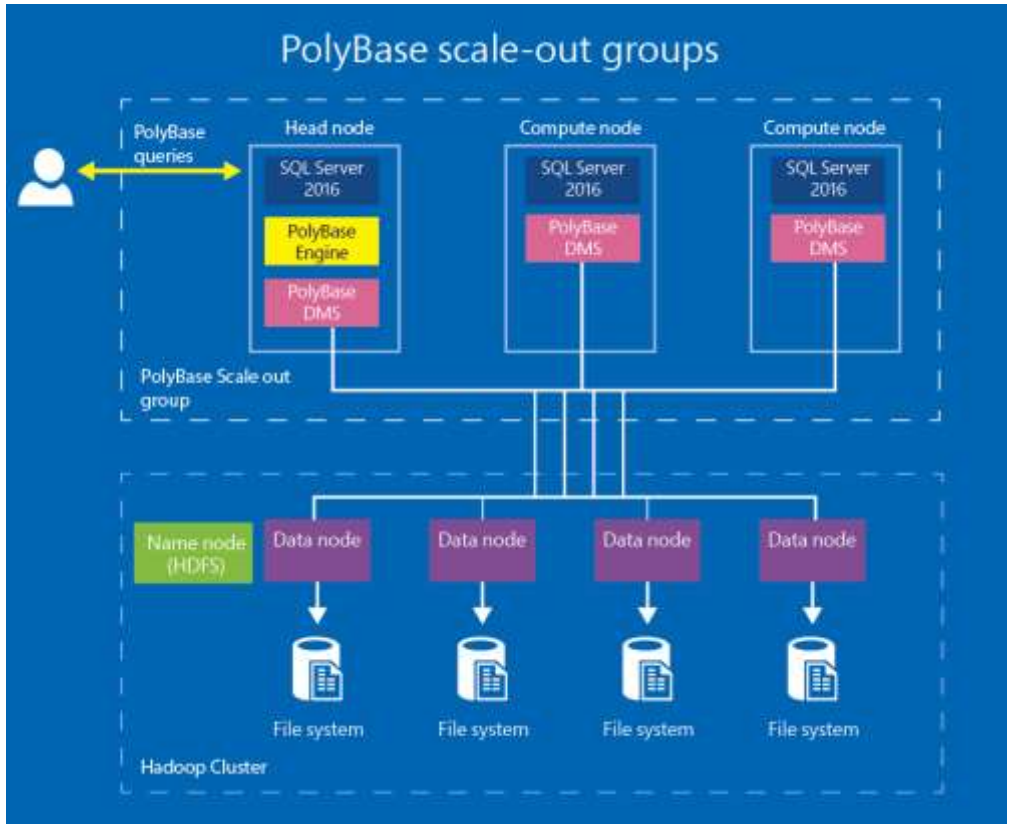


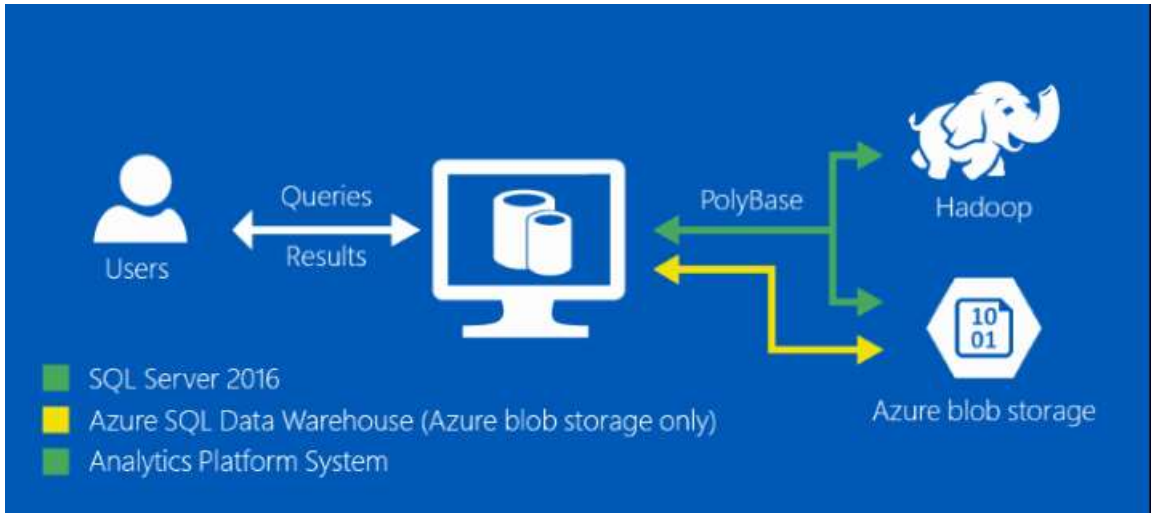
Le challenge du big data

- Une unique instance SQL Server ne suffit pas (et n'a pas été conçue pour) pour l'analyse de petabytes de données, structurées ou non structurées.
- Il faut donc y ajouter des fonctionnalités :
  - Distribution et parallélisation du stockage et du calcul
  - Virtualisation de données : accès et ingestion de données hétérogènes, structurées ou non
  - Capacités et langages pour l'interrogation et le machine learning

SQL Server 2016

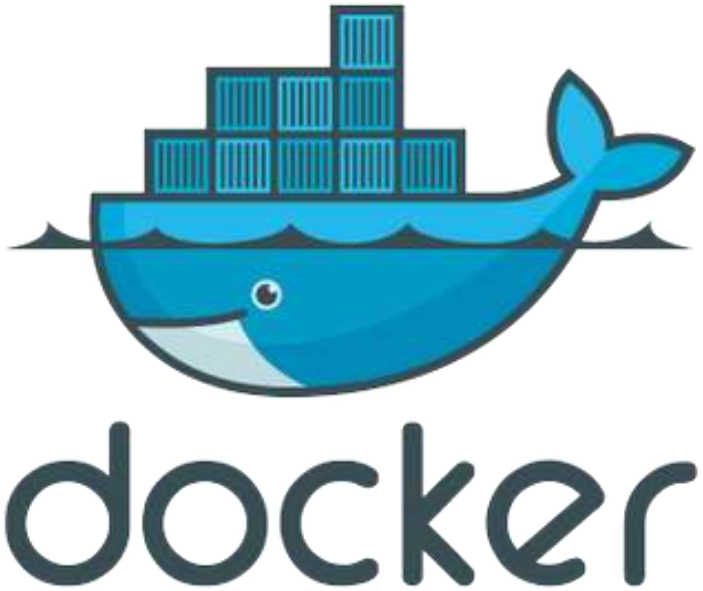
Polybase -> Hadoop, R Server





SQL Server 2017

SQL Server on Linux, containers Docker, Python



SQL Server 2019 (CTP)

Intégration Spark et HDFS « in the box »

SQL Server Big Data Cluster avec Docker et Kubernetes

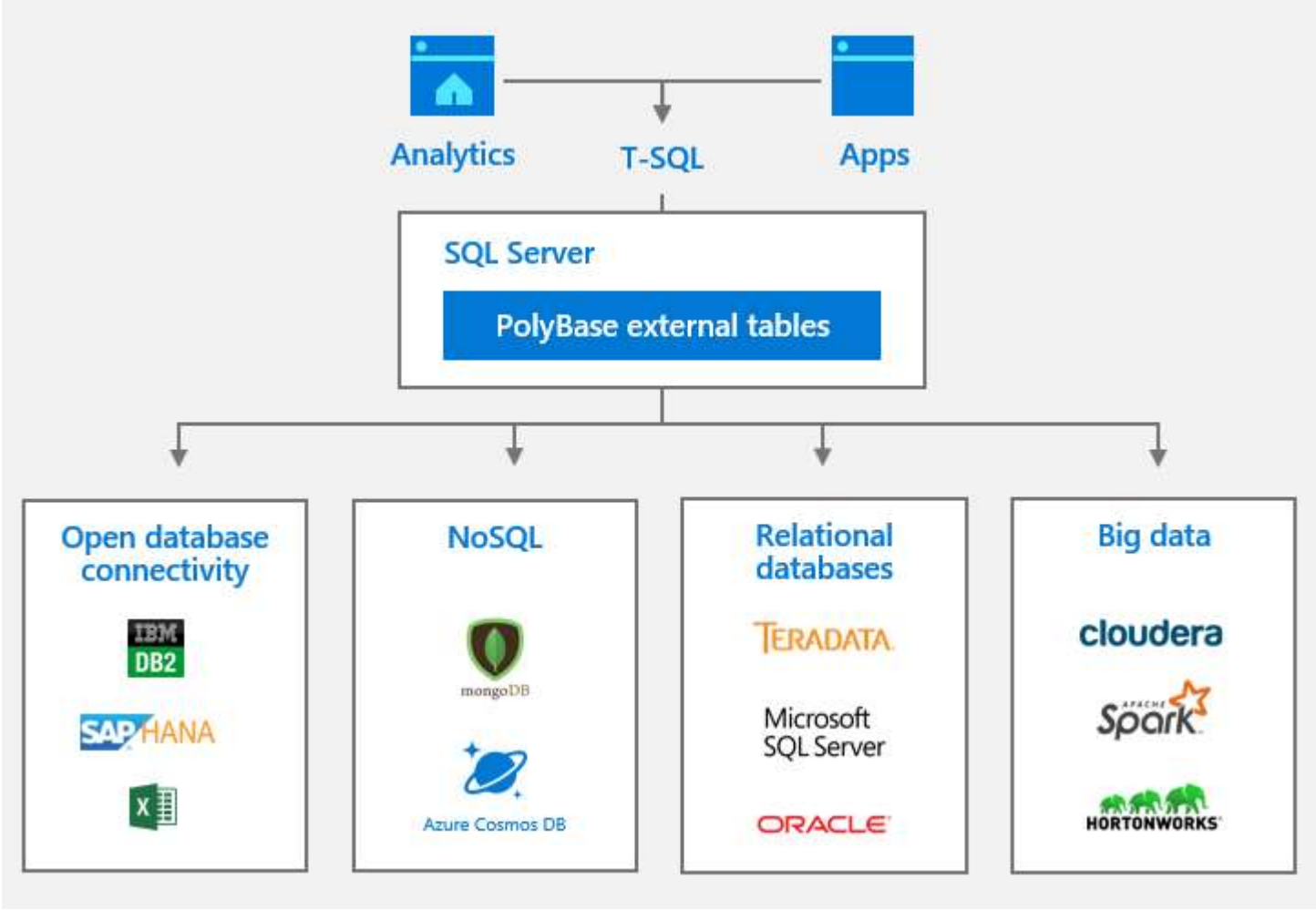
Polybase : nouveaux connecteurs Oracle, Teradata, MongoDB

Machine Learning Services : R, Python, Java

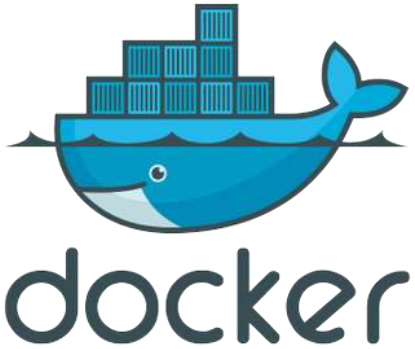
Nouvel outil Azure Data Studio

Polybase

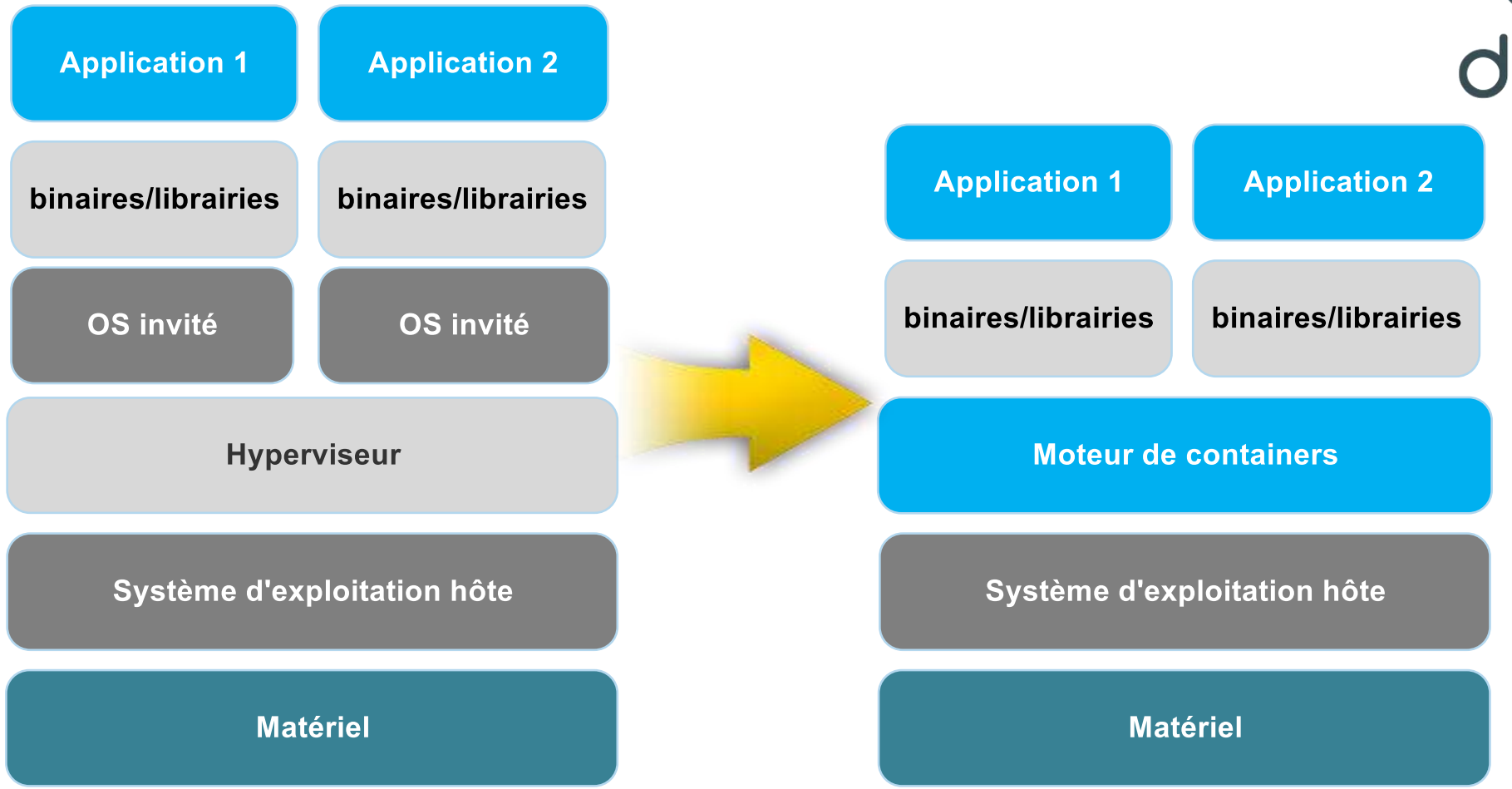
- Virtualisation de données
  - Alternative à l'ETL
  - Création de tables externes



Conteneurs : docker



- Solution de virtualisation d'applications
  - Déploiement d'images de programmes et de leur environnement d'exécution,
  - Environnements isolés, plus légers que des machines virtuelles : partagent le même kernel
- Peut être utilisé pour étendre des systèmes distribués
- Docker est une plate-forme open source de conteneurs


Application 1
Application 2
binaires/librairies
binaires/librairies
OS invité
OS invité
Hyperviseur
Système d'exploitation hôte
Matériel
Machines virtuelles
Application 1
Application 2
binaires/librairies
binaires/librairies
Moteur de containers
Système d'exploitation hôte
Matériel
Containers

Que peut faire un container tout seul ?

Le container n'a pas connaissance de ce qui se passe sur la machine hôte

Dans ces conditions, comment assurer la haute disponibilité et la scalabilité d'un service ?

Orchestration des containers

Création de services applicatifs sur différents conteneurs

Planification de l'exécution des conteneurs dans un cluster

Garantir l'intégrité des conteneurs

Assurer le monitoring

Kubernetes

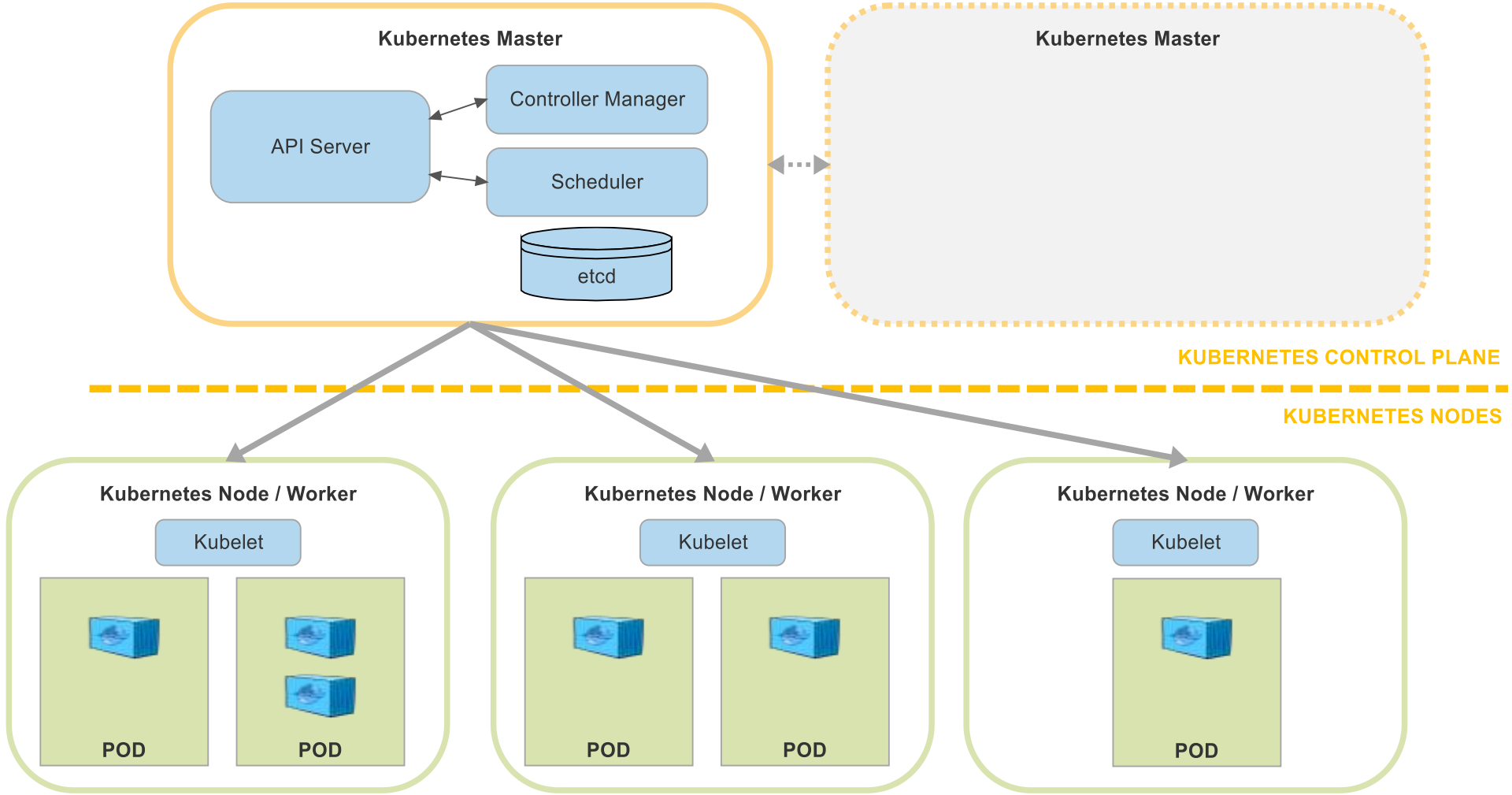
Kubernetes (de κυβερνήτης en grec pour « timonier » ou « pilote »)
Le « pilote » du porte-container

Kubernetes (k8s)

- Kubernetes est une plate-forme open source d'orchestration de conteneurs
  - Les conteneurs sont indépendants et isolés les uns des autres : un orchestrateur est nécessaire pour construire une architecture distribuée avec de multiples conteneurs.
- Permet de construire des clusters de serveurs avec des groupes de conteneurs organisés en unité logique
  - Déploiement et extensibilité
  - Haute disponibilité
  - Calcul distribué
  - Supervision

kubernetes

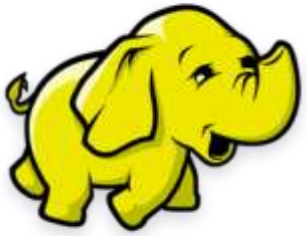
Architecture Kubernetes (k8s)

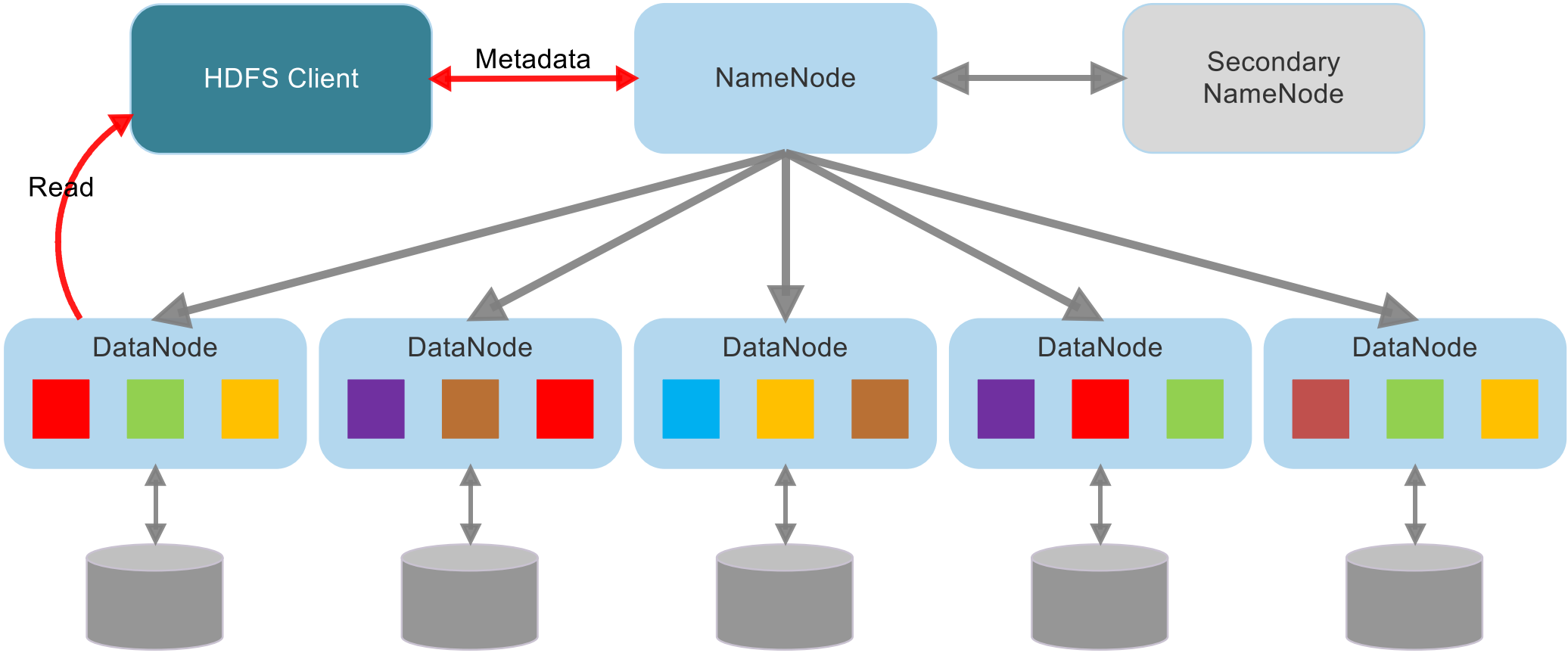


Stockage : HDFS

- Système de fichiers distribué : Hadoop Distributed File System
- Conçu pour stocker de très gros volumes sur un grand nombre de serveurs

HDFS



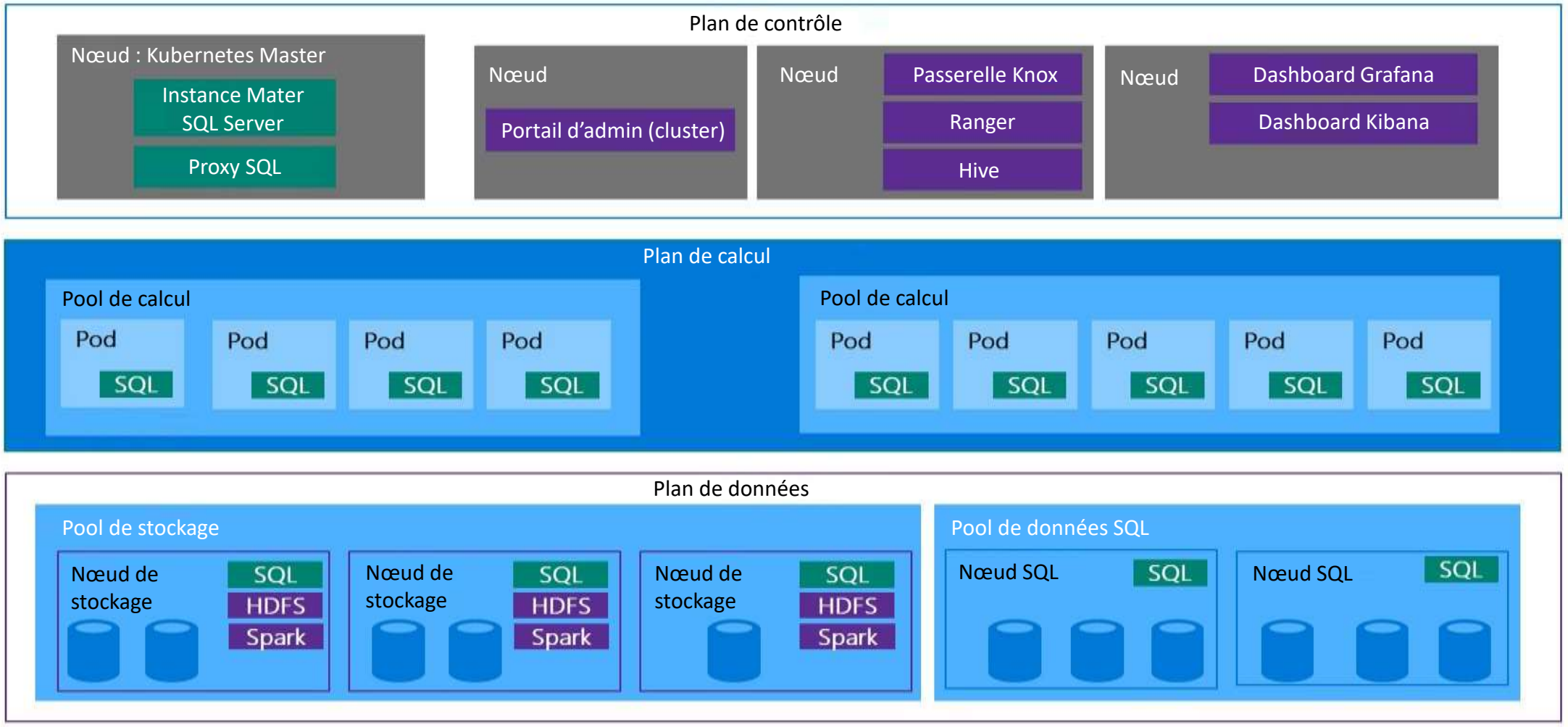

HDFS Client
Metadata
NameNode
Secondary NameNode
Read
DataNode
DataNode
DataNode
DataNode
DataNode

Spark

- Framework open source de calcul distribué
- Plus rapide que map-reduce : travaille en parallèle
- Spark jobs pour le machine learning
- Langage natif Scala, mais aussi Python et Java

```scala
// Every record of this DataFrame contains the label and
// features represented by a vector.
val df = sqlContext.createDataFrame(data).toDF("label", "features")

// Set parameters for the algorithm.
// Here, we limit the number of iterations to 10.
val lr = new LogisticRegression().setMaxIter(10)

// Fit the model to the data.
val model = lr.fit(df)

// Inspect the model: get the feature weights.
val weights = model.weights

// Given a dataset, predict each point's label, and show the results.
model.transform(df).show()
```

Architecture : Cluster SQL Server Big Data avec Kubernetes



Plan de contrôle

- Nœud maitre Kubernetes : nœud dédié à la gestion et au contrôle du cluster kubernetes
- Apache Knox Gateway : passerelle, point d'accès aux API REST du cluster
- Apache Livy : utilisé pour soumettre des jobs à Spark
- Apache Hive : stockage distribué pour les métadonnées de spark
- Grafana : tableau de bord de performances
- Kibana : visualisation des logs
- SQL Server Controller Service : manage le cluster sql server, déployé avec l'utilitaire mssqlctl
- SQL Server Master Instance : instance maître SQL Server, point d'accès TDS

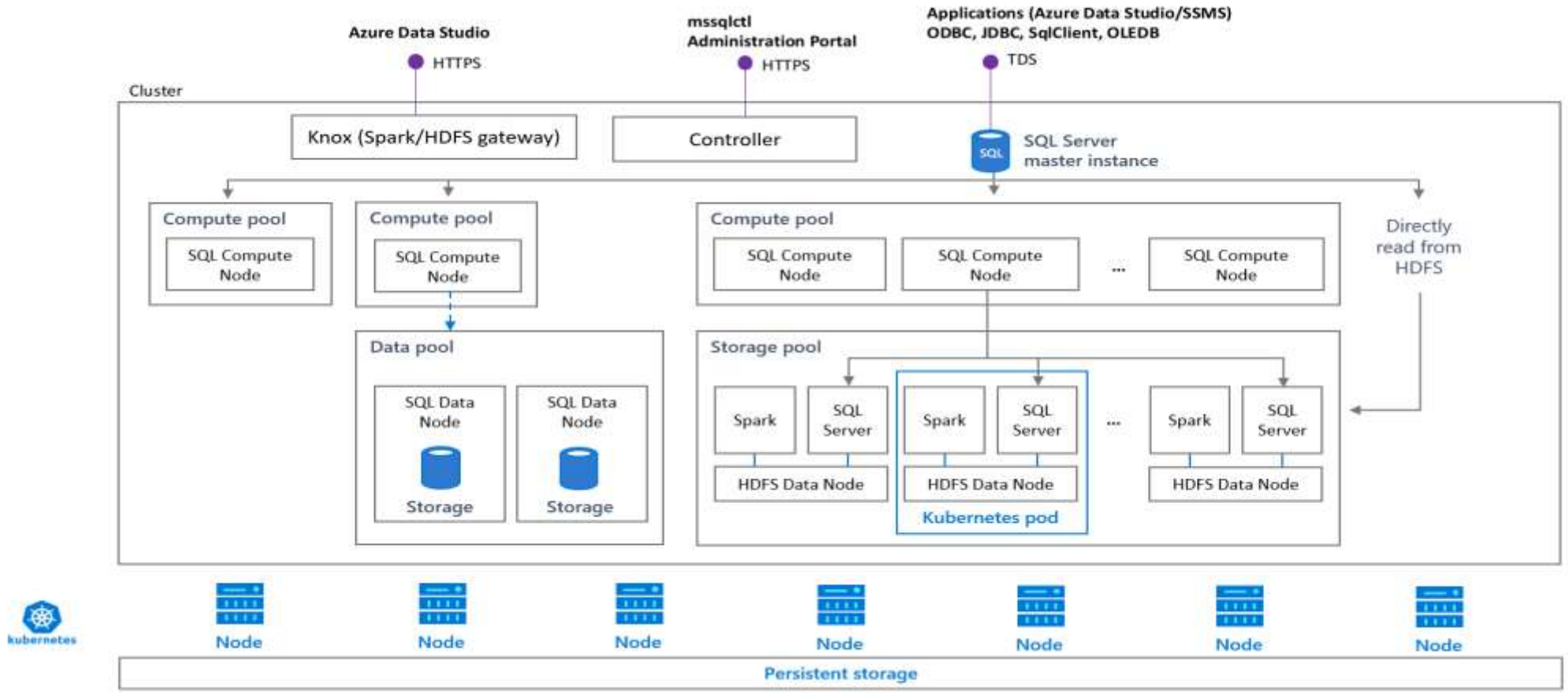
Plan de calcul

- Compute pool
  - Un ou plusieurs pods (jusqu'à 4 en CTP 3.0) utilisés pour distribuer les traitements, sous la direction de l'instance maître
  - Fait appel aux collecteurs Polybase pour les traitements distribués
- App pool
  - Un ensemble de pods, points d'accès au système pour des applications
  - SSIS, jobs, Machine Learning, etc…

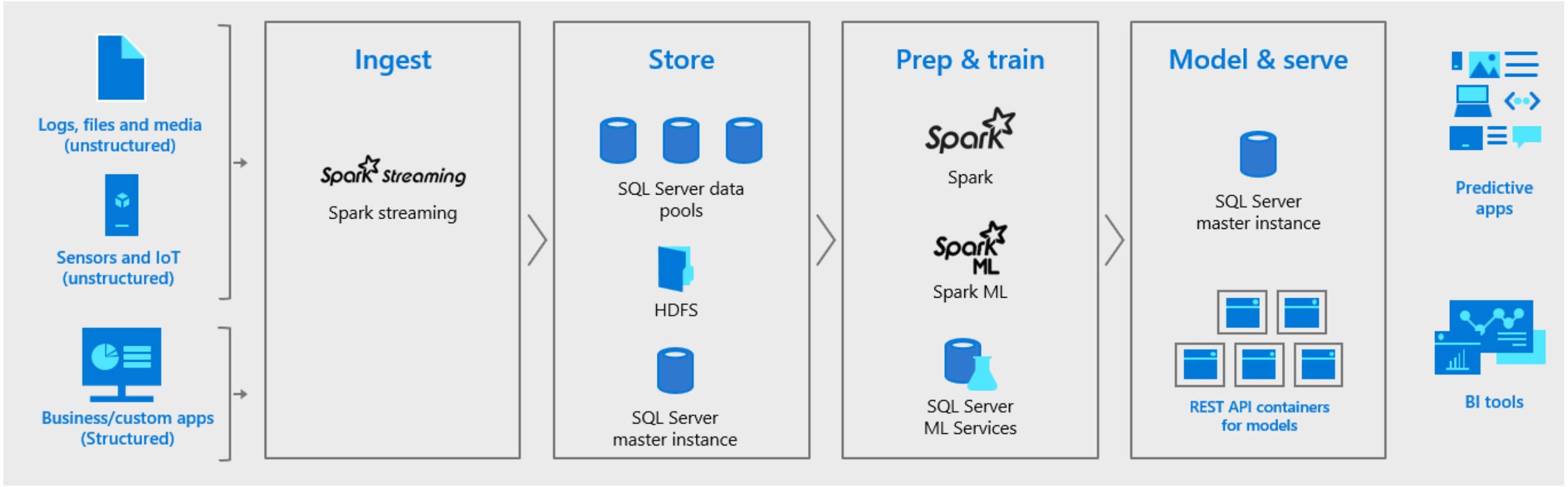
Plan de données

- Data pool
  - Une ou plusieurs instances qui fournissent le stockage persistant au cluster SQL
  - Les données sont distribuées sur plusieurs partitions entre les pods du data pool
- Storage pool
  - Offre le stockage HDFS pour stocker des données de sources diverses

Architecture : Cluster SQL Server Big Data avec Kubernetes



SQL Server : Plate-forme d'analyse de données



Comment on le met en place ?

Azure via Azure Kubernet Services

On premise

AKS

Prérequis

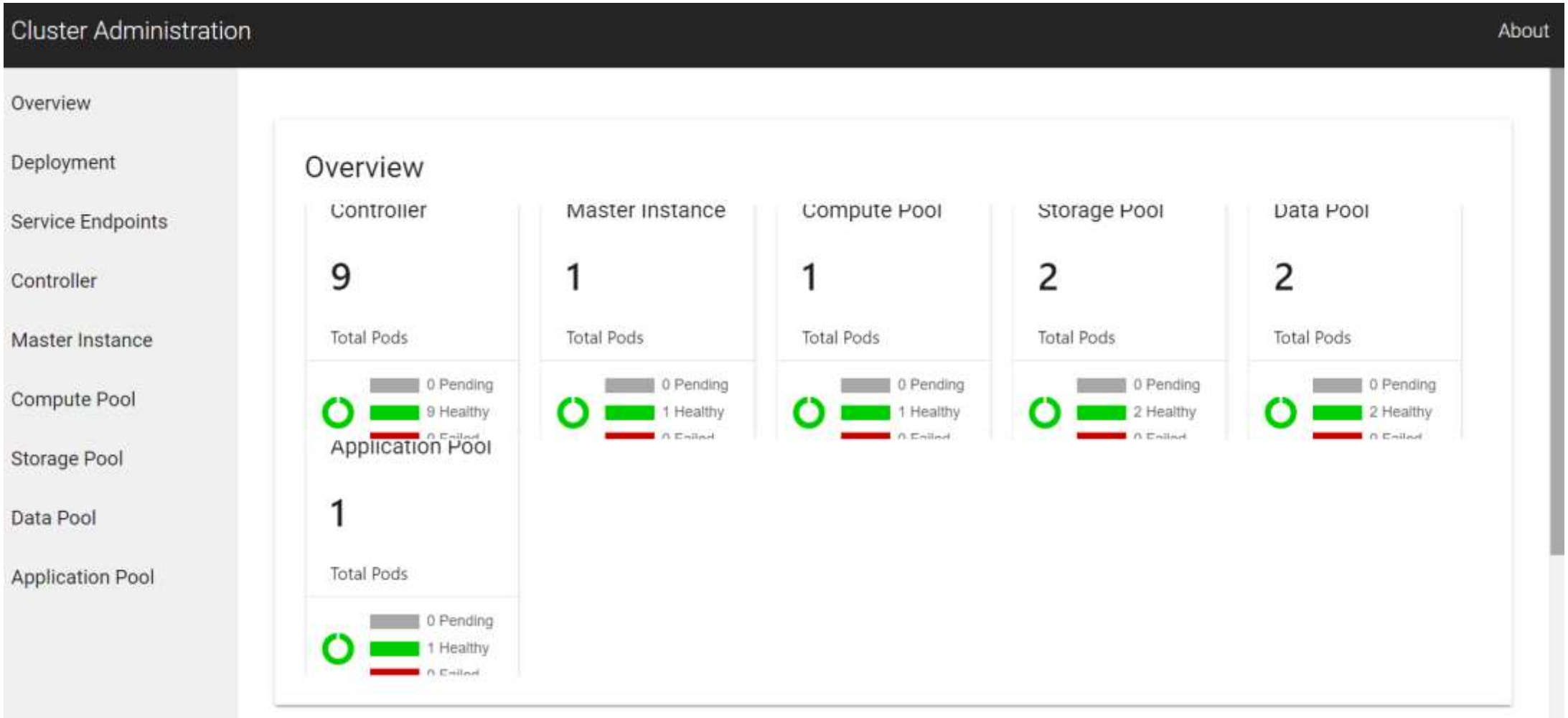
Un abonnement Azure.

Outils de Big data:

Mssqlctl (utilitaire en ligne de commande Python pour installer les clusters)

Kubectl (Kubectl est une interface en ligne de commande qui permet d'exécuter des commandes sur des clusters Kubernetes)

SQL Server 2019 (Preview) : Portail d'admin du cluster



REX: AKS

- Installation sur Azure :
  - Nécessite les variables d'environnement
  - Installation des pré requis Big Data
  - Soucis de stabilité jusqu'au version 3.0
  - Import des données ou des bases plus complexe que la version On prem

REX : Installation (de démo) on premise

- 3 machines, 6 cœurs, 32 GB RAM, 100 Go disque
- Linux Ubuntu 16.04 (ou 18.04)
  - Installer les pré-requis kubernetes et docker
  - Créer et configurer le cluster et le master kubernetes avec kubeadm et kubectl
  - Joindre les autres nœuds avec kubectl join
  - Provisionner/configurer le stockage
  - Installer mssqlctl et python
  - Déployer le cluster avec mssqlctl cluster create

REX

- Solution intéressante pour la partie BIG DATA:
  - Solution de map Reduce (Cluster Spark intégré)
  - Connection vers de nouvelles sources de données grâce a PolyBase (MySQL / Oracle / Sql Server)
  - Amélioration du système de stockage (HDFS)
  - Solution distribuée On Prem / Azure

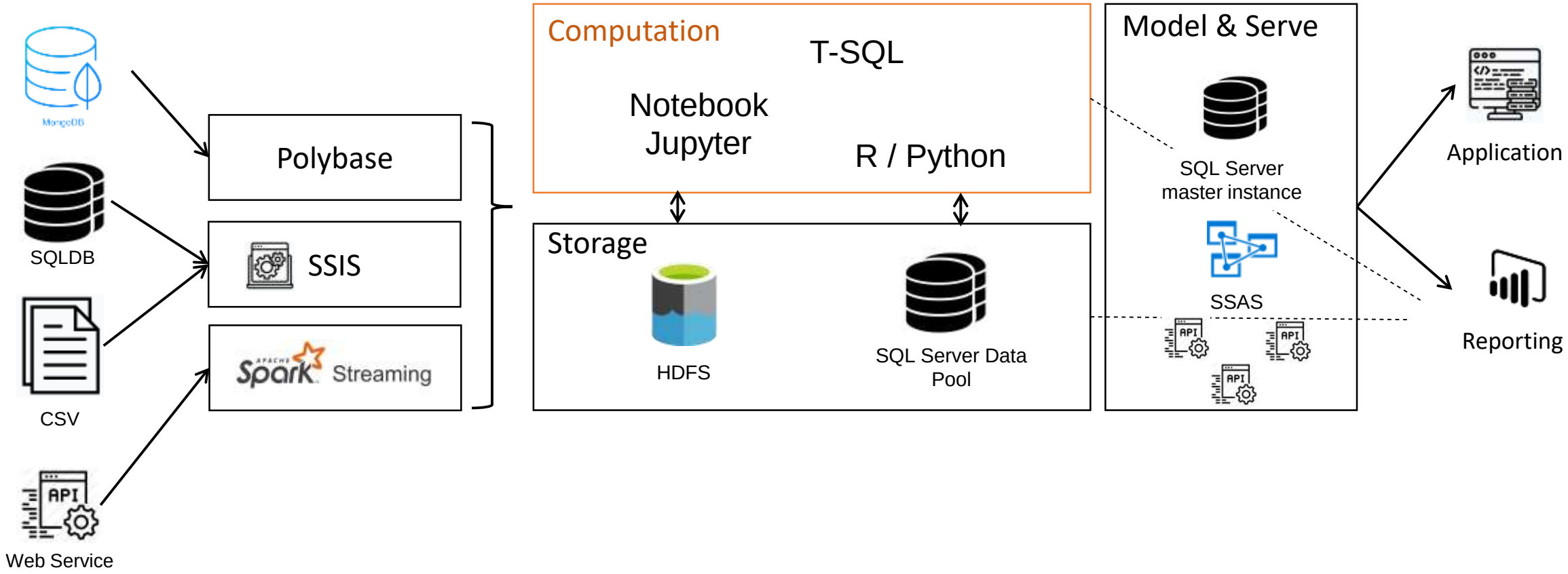
Les autres alternatives ?

- Créer un cluster Spark sur installation standard ?
  - Depuis la version 2019 il est possible d'installer des composants Machine Learning directement sur une instance
  - Il est possible d'utiliser le notebook Jupyter sans avoir une installation en cluster :
    - Objectif : réaliser les Map reduce / et les algorithmes d'entrainement directement depuis le serveur avant de les ingérer en base.
    - Interroger les résultats dans la table externe (Polybase)

Du coup j'utilise Quoi et Quand ?

- Petite volumétrie connexion vers plusieurs bases ?
  - Polybase SQL Server
  - Nécessité de réaliser du MAP Reduce ou de l'Analytics ?
    - Spark
      - Volumétrie restreinte ?
      - Oui ?
        - » Utilisation de Spark On Prem
      - Grosse volumétrie ?
        - » Installation d'un cluster BIG Data

Architecture Data Lake



démo

Des Questions ?

Merci!