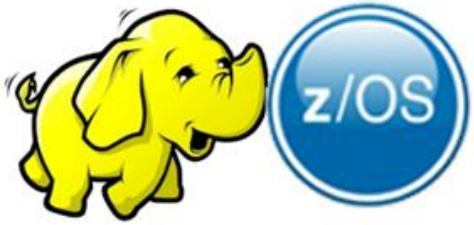
Marc AMADOU – Technical Sales Analytics on System z
amadoum@fr.ibm.com

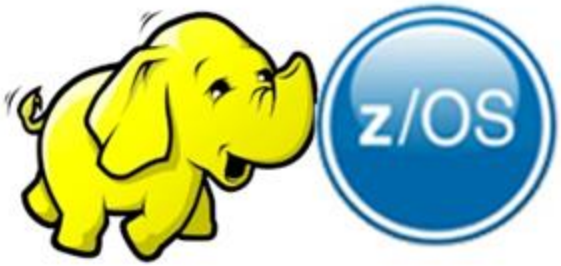
18 Mars 2015

Big data et le z



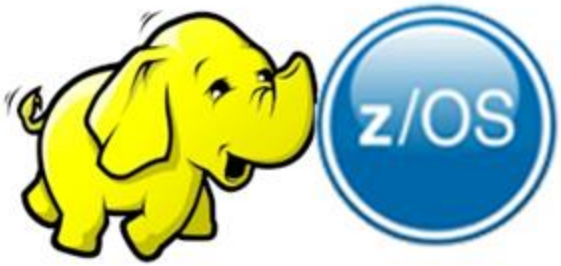
Agenda

- Contexte
- Cas d'utilisation
- DB2 z/OS et Hadoop
- Connecteurs z pour Hadoop



Agenda

- **Contexte**
- Cas d'utilisation
- DB2 z/OS et Hadoop
- Connecteurs z pour Hadoop



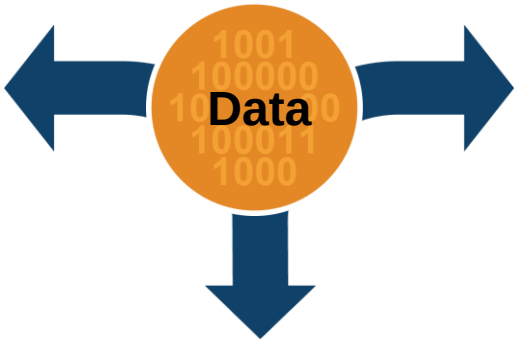
Contexte Big Data

Systems of Record

z/OS est la technologie prédominante pour sécuriser, stocker et traiter les données transactionnelles critiques, de manière fiable

Systems of Engagement

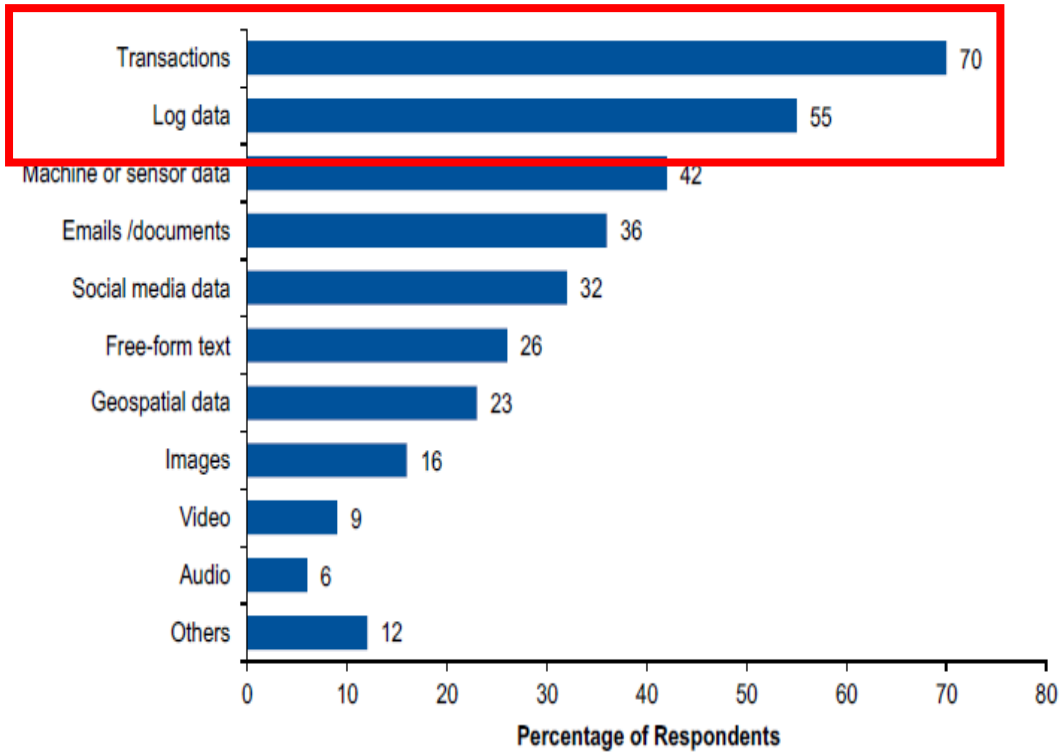
Hadoop fournit une solution rentable pour gérer les données non structurées – logs, données issues des réseaux sociaux, etc...



La **valeur métier** provient de plus en plus de la **fusion de ces sources multiples et hétérogènes**

Business Value

Les initiatives Big Data impliquent surtout des données transactionnelles & de logs



N =465 (multiple responses allowed)

Source: Gartner (September 2013)

- Alors que les données non structurées ont le vent en poupe, la majorité des projets à forte valeur ajoutée implique des données transactionnelles (1)
- Une grande partie de ces données transactionnelles sont sur z
- Hadoop est mieux adapté aux types de données structurées et semi-structurées

1. 70% of 465 survey respondents cite transactional data as a primary target for big data initiatives - Gartner research note “Survey Analysis - Big Data Adoption in 2013 Shows Substance Behind the Hype“ Sept 12 2013 Analyst(s): Lisa Kart, Nick Heudecker, Frank Buytendijk

Big Data – Cas d'utilisation

Exploration des Big Data

Trouver, visualiser, comprendre toutes les big data pour améliorer la prise de décision

Améliorer la vue 360° du Client

Etendre les vues existantes des clients (MDM, CRM, etc) en incorporant des sources d'information internes et externes

Extension de la Sécurité

Diminuer le risque, détecter la fraude et monitorer la cyber sécurité en temps réel

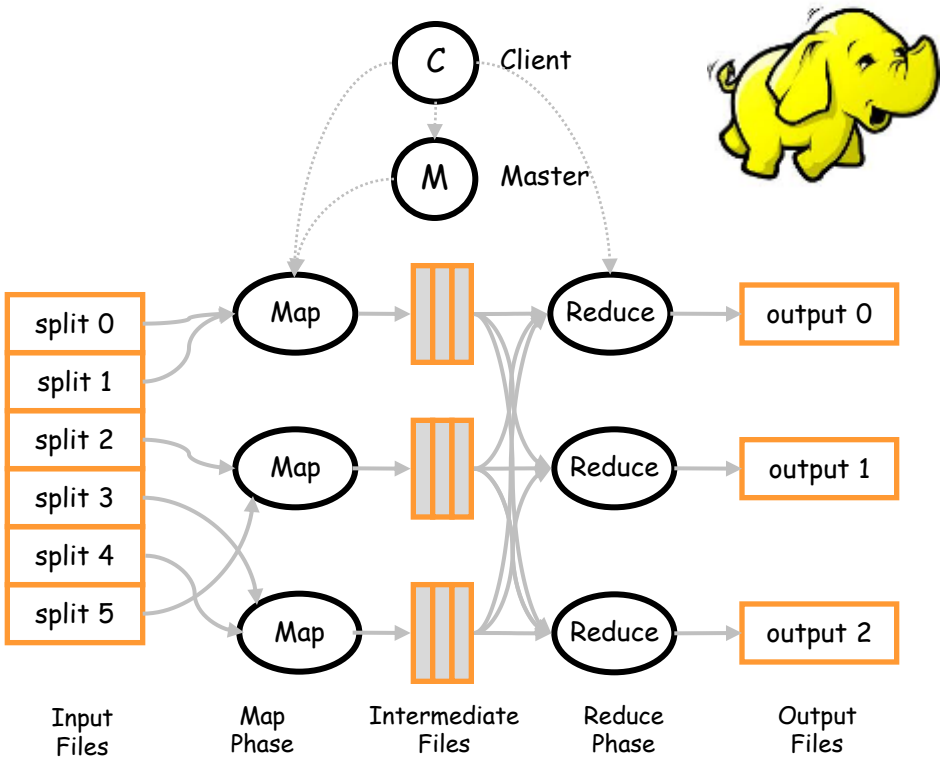
Opérations d'Analyse

Analyser une variété de données réparties pour améliorer les résultats business

Data Warehouse

Intégrer les big data et les capacités des data warehouse pour augmenter l'efficacité opérationnelle

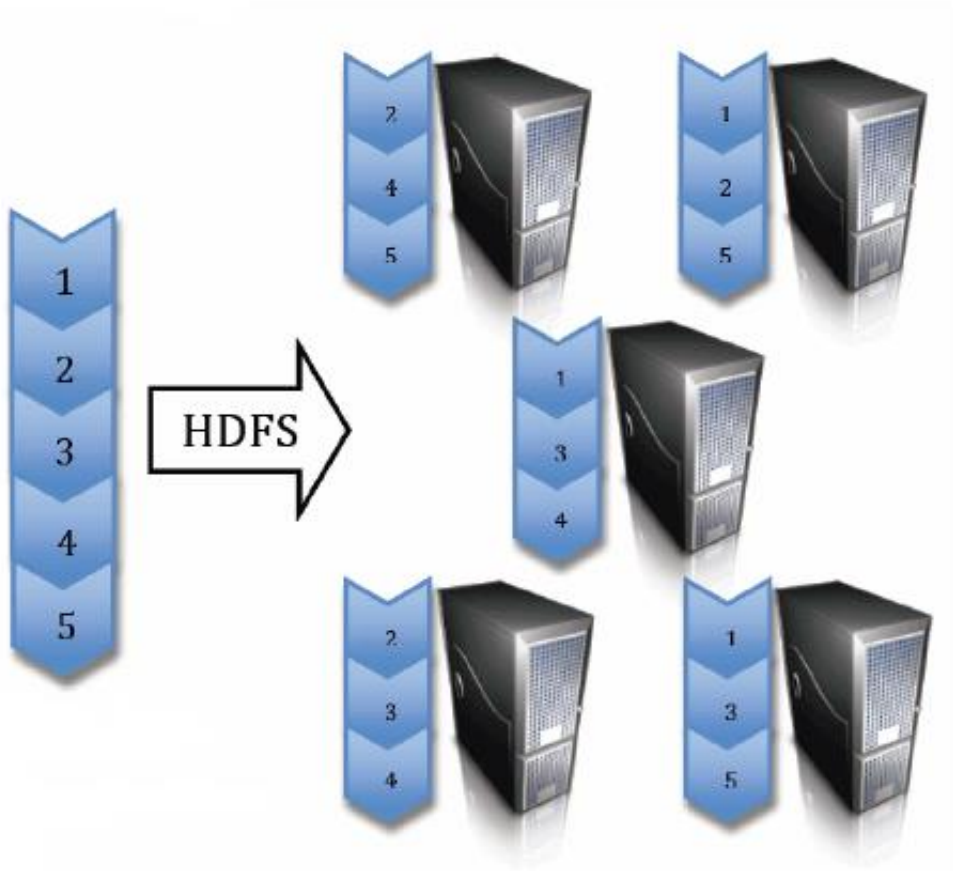
Hadoop attire presque toute l'attention pour les nouveaux Workloads



- Inspiré par les publications de Google et Yahoo!
- Framework conçu la création d'applications traitant des gros volumes de données (pétaoctets)
- Minimise les mouvements de données
- Plus rentable que les Data Warehouses traditionnels
- Gains phénoménaux en termes de performance pour certaines applications
- Environnement agile pour du prototypage ou du développement d'application
- Plusieurs distributions commerciales: Cloudera CDH, Hortonworks HDP, IBM BigInsights, MapR

Hadoop Distributed File System (HDFS)

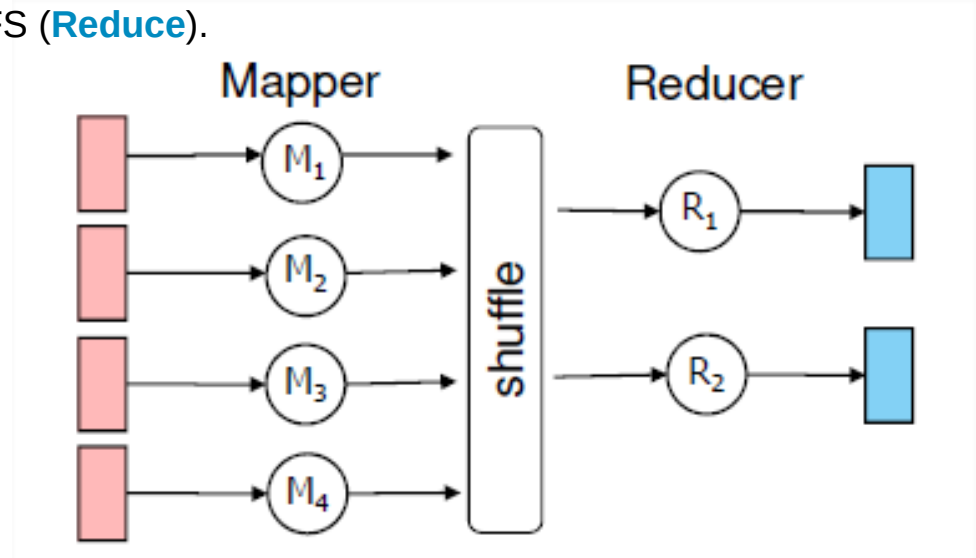
- Principe sous-jacent:
  - Beaucoup de disques redondants (si perte on remplace)
  - Beaucoup de machines (cœurs) avec des CPs bon marché (si perte on remplace)
  - Problèmes réseaux, on recommence
- Les fichiers sont éclatés en « Large Blocks » (64MB par défaut)
- Les blocks sont répliqués (3 fois par défaut) et distribués à travers le cluster
- Optimisé pour:
  - La lecture en Streaming des gros fichiers
  - Modèle d'accès « write-once-read-many » (une écriture, mise à jour via rajout uniquement)

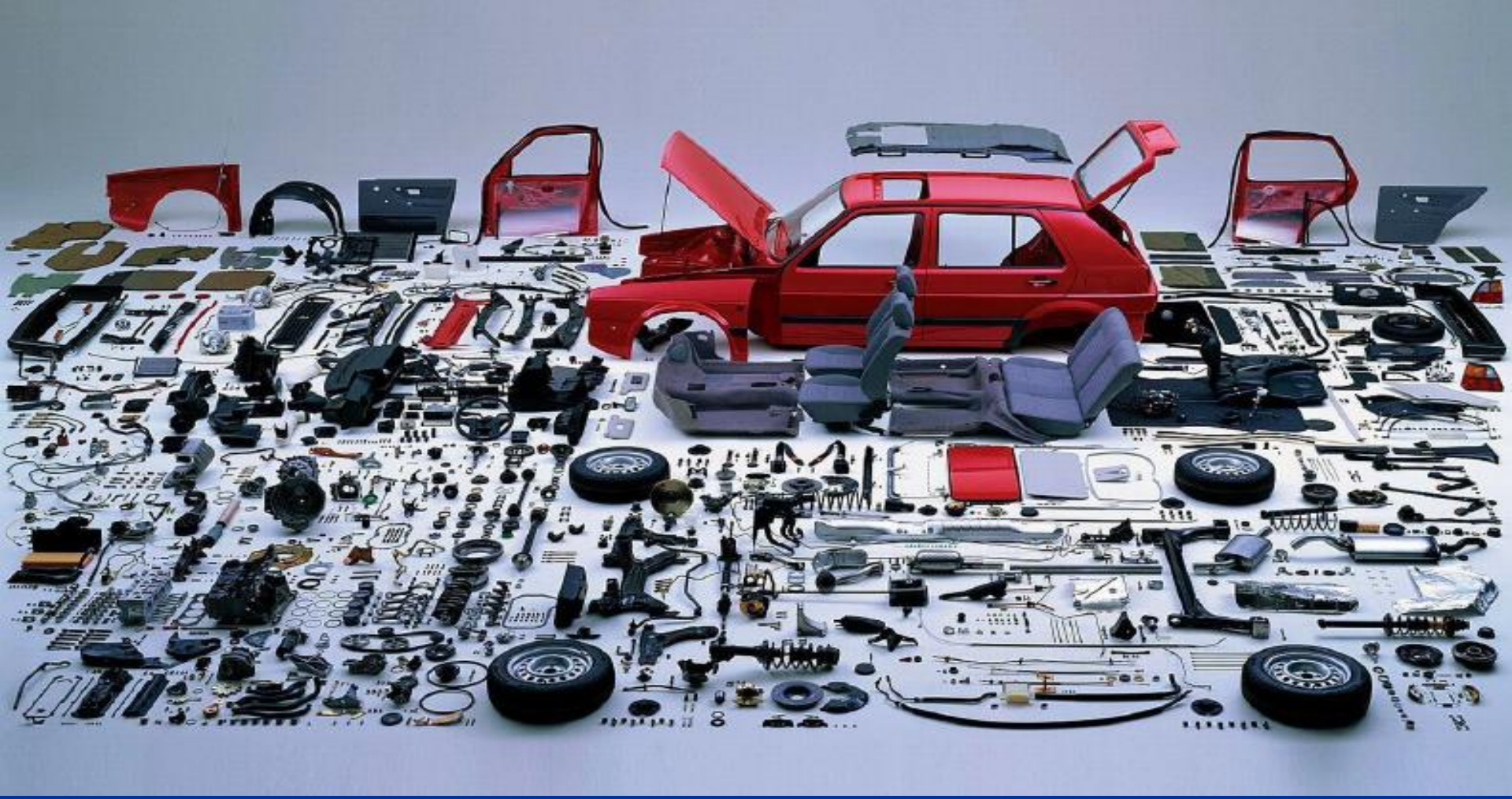


MapReduce

- Un Framework simple mais très puissant/efficace pour les calculs parallèles
  - Consiste en 2 fonctions **map()** et **reduce()**,
  - Mais applicable à l'analyse d'une multitude de problèmes (traitements de données, fouilles de données, graphes, …)

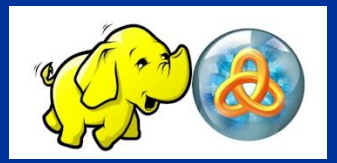
- Étapes de base :
  - Exécution des calculs parallèles (**Map**) sur chaque bloque (**split**) de données dans un fichier HDFS et envoi de l'Output (**Map Output**) constituée d'une **paire (clé, valeur)** dans le système de fichiers local,
  - Redistribution (**Shuffle**) de la « **Map Output** » par clé,
  - Exécution d'autres calculs parallèles sur le résultat de la « **Map Output** » redistribuée et écriture des résultats dans HDFS (**Reduce**).





Le “Time to value” est important

Même si vous *pouvez* construire vous même une solution à partir de 0, ça ne veut pas dire qu’il faut que vous le fassiez*!*



BigInsights: 100% Open Source Hadoop, mais au niveau Enterprise

Value-Added Capabilities

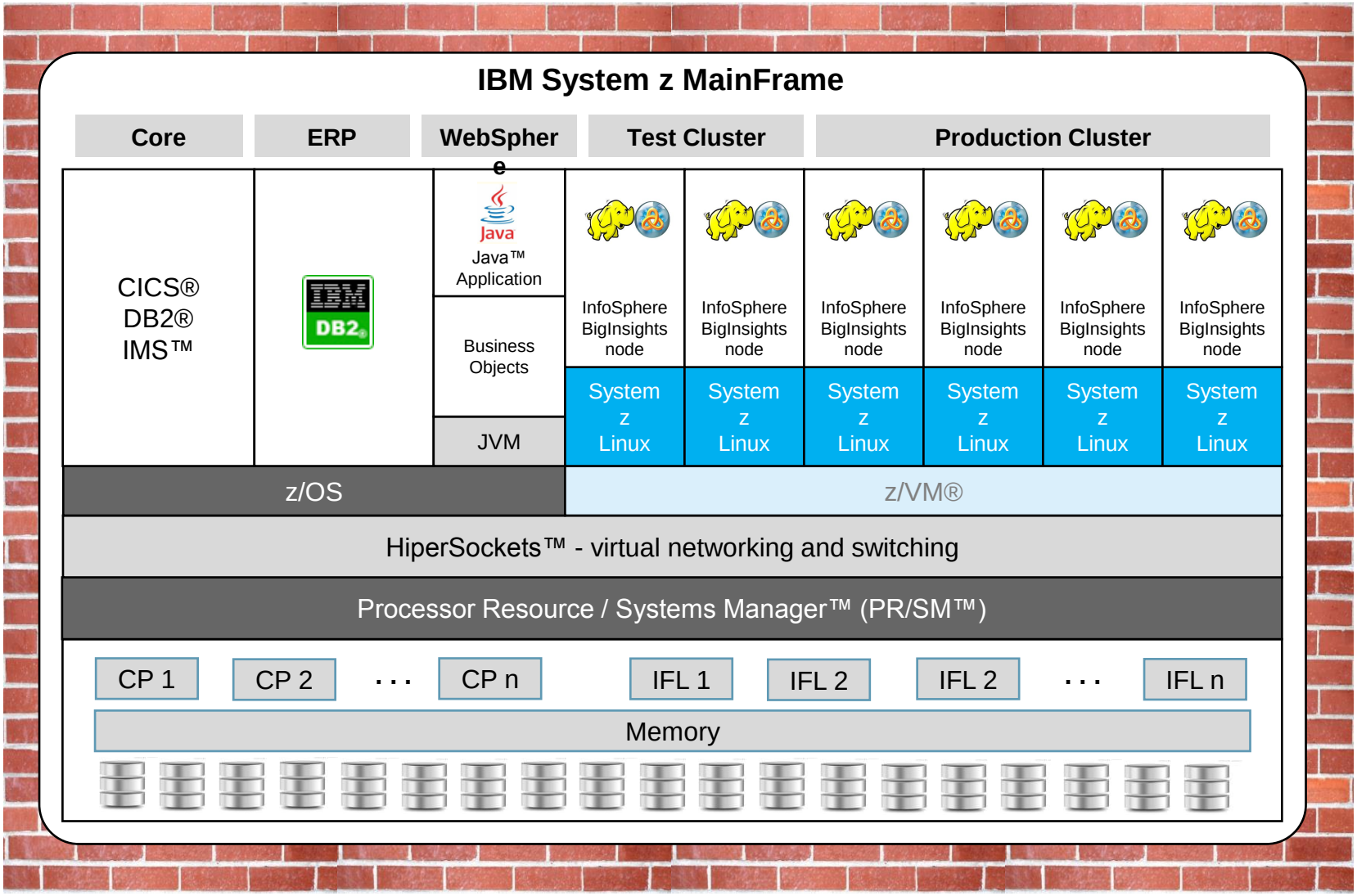
- **SQL on Hadoop**
  Big SQL – optimized ANSI compliant SQL
- **Application Tooling**
  Toolkits and accelerators
- **Search**
  BigIndex and Data Explorer
- **Data Exploration**
  BigSheets “schema-on-read”
- **Predictive Modeling**
  Big R – scalable data mining
- **Text Analytics**
  Advanced text processing with AQL
- **Real-time Analytics**
  InfoSphere Streams
- **Data Governance and Security**
  Data Click, LDAP, Secure cluster
- **Storage Integration**
  GPFS - POSIX Distributed Filesystem
- **Enterprise Features**
  Adaptive MapReduce, Recoverable jobs

100% Standard Apache Open-Source Components

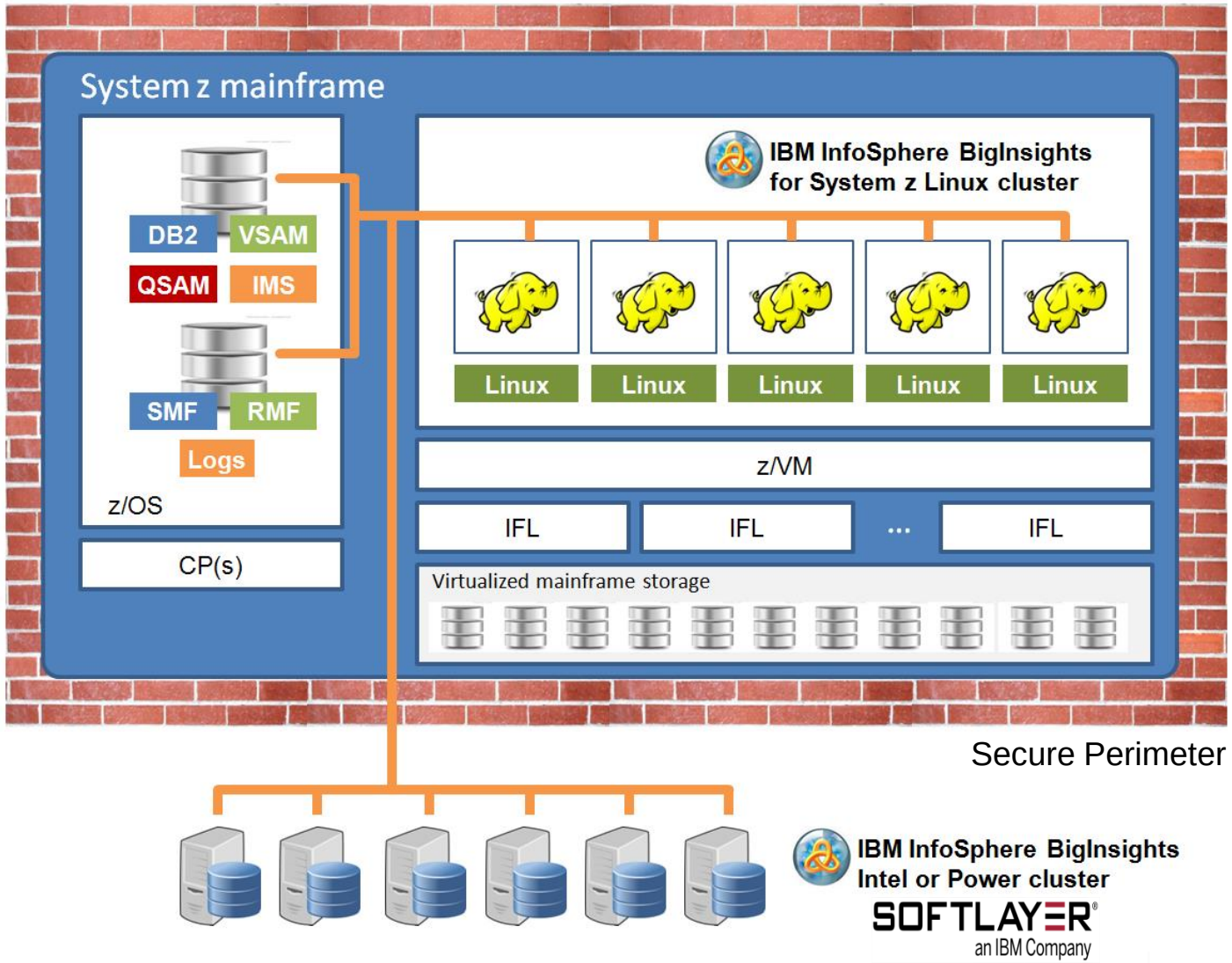
Oozie	Jaql	Zookeeper	Hive	HCatalog
HDFS	MapReduce	HBase	Flume	Sqoop
YARN*	Spark*	Avro	Pig	Solr/Lucene

IBM InfoSphere BigInsights sur System z

Secure Perimeter



Flexibilité de déploiement: Sur System z ou à l'extérieur



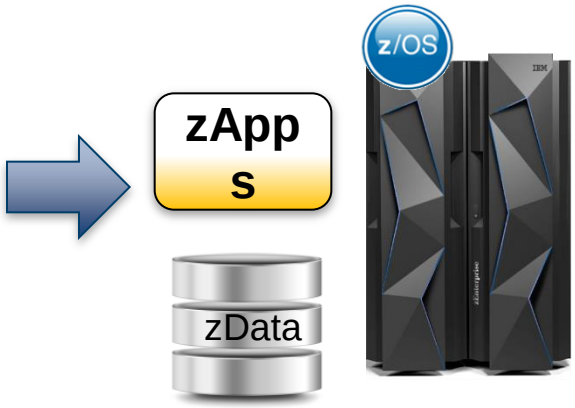
Agenda

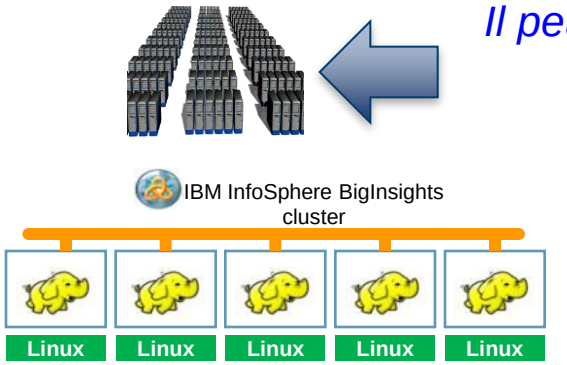
- Contexte
- **Cas d'utilisation**
- DB2 z/OS et Hadoop
- Connecteurs z pour Hadoop

Intégration Hadoop & Systems z – Analyses enrichies

- De très gros volumes de données non-relationnelles sont générés à l'extérieur du System z
  - Ex: e-mails envoyés par les clients, les tweets, les posts sur la page Facebook de l'entreprise
- Analyser les sentiments et identifier les clients mécontents d'une entreprise
  - Mots 'annuler', 'arrêter', 'changer' ou tout autre synonyme
  - Noms des concurrents
- Rassembler les noms et les adresses e-mail des clients à risque
- Jointure des résultats avec les données opérationnelles
  - Alerter les agents des clients « à risque »
  - Les agents travaillent avec les clients et leurs proposent des offres et des promotions pour éviter qu'ils ne s'en aillent

Les données & les applications opérationnelles sont ici

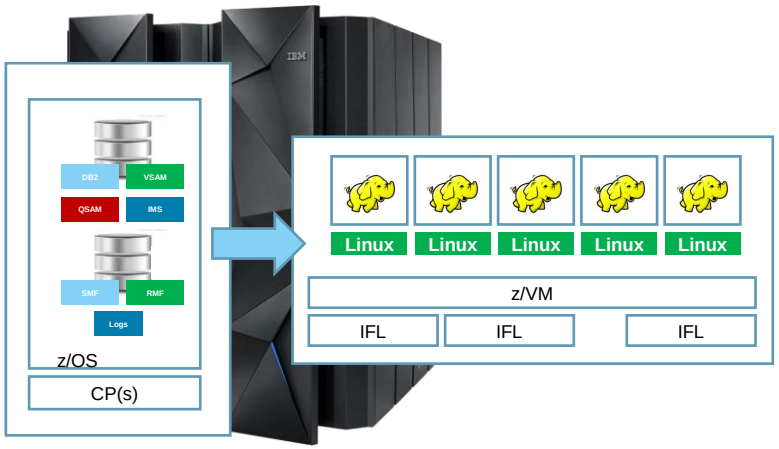




Il peut aussi y avoir des données pertinentes ici

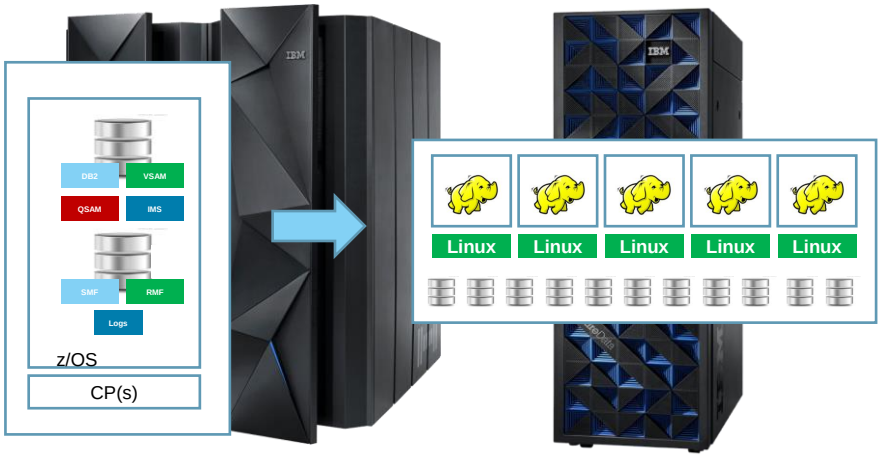
Qu'est ce qui fait sens et quand?

Cas 1: Hadoop sur z



- La plupart des données proviennent du z (fichiers de Log, extractions de base de données)
- La sécurité des données est la préoccupation principale
- Les clients ne veulent pas envoyer les données sur le réseau externe
- Volumes de données relativement faibles – 100 GB à une 10s de TBs
- Hadoop est apprécié principalement pour la richesse des outils
- Besoin des modèles de sécurité et de gouvernance Z

Cas 2: Hadoop en dehors du z

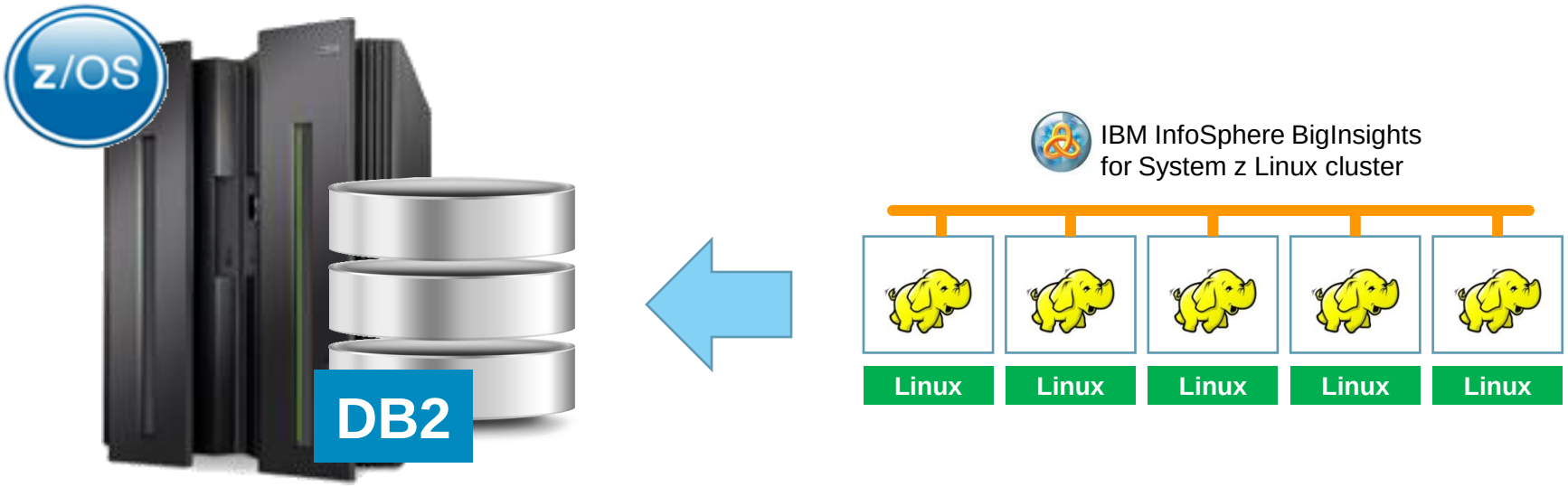


- La plupart des données ne proviennent pas du z
- La sécurité n'est pas la préoccupation principale. On pars du principe que les données ne sont pas de confiance de toute façon
- Très gros volumes de données – 100s de TBs à des PBs
- Hadoop est apprécié pour sa capacité à gérer de manière rentable les gros volumes de données
- Volonté de s'appuyer sur des capacités de traitement bon marché

Agenda

- Contexte
- Cas d'utilisation
- **DB2 z/OS et Hadoop**
- Connecteurs z pour Hadoop

DB2 version 11: Integration avec des sources de données Big Data



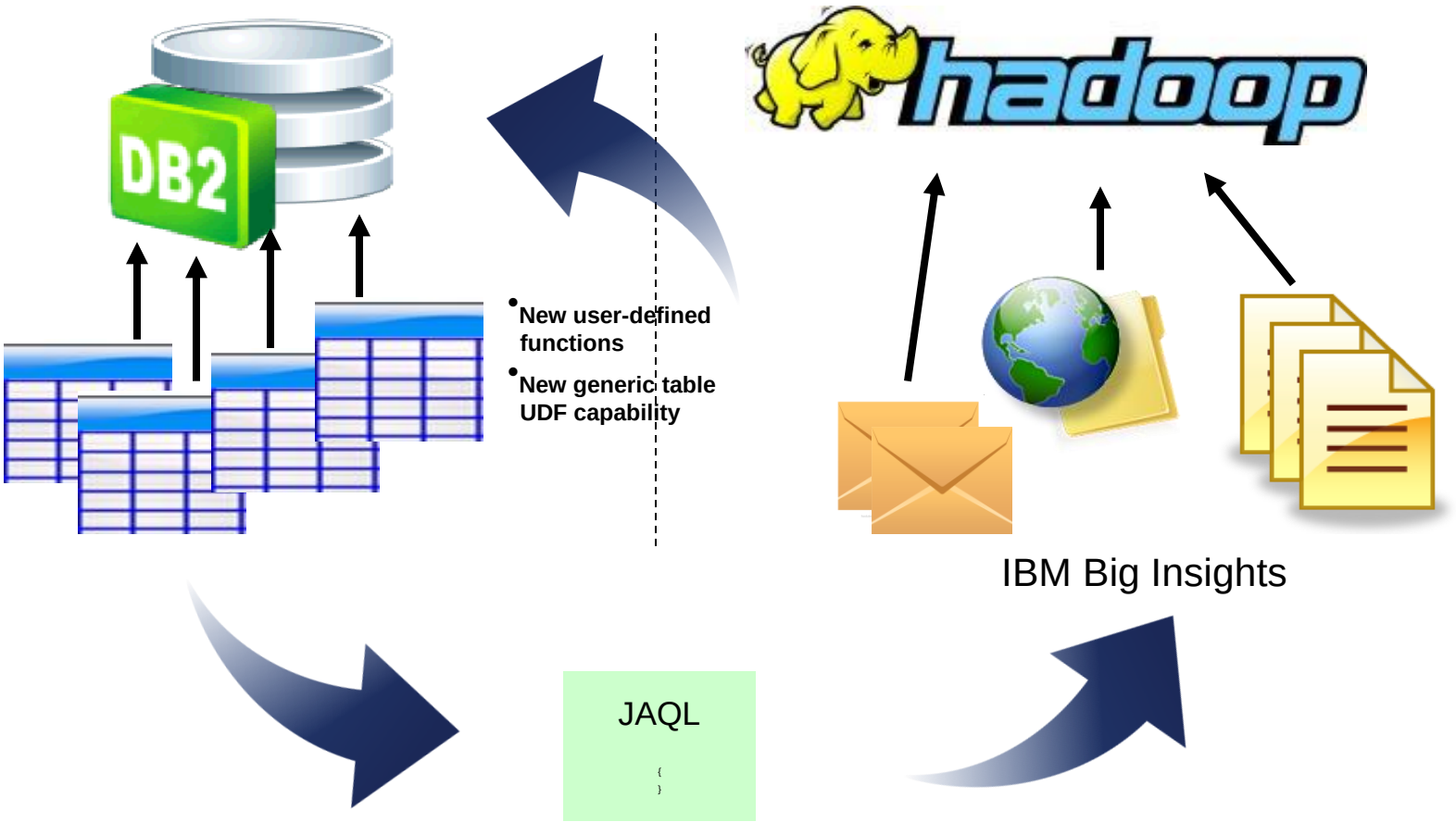
Sous-traiter à Hadoop depuis DB2

- Intégration de DB2z à BigInsights – fonctionnalité importante de DB2 v11
- Java MapReduce est « **l'assembly language**» d'Hadoop
- Lancer des requêtes JSON (via JAQL depuis des UDFs DB2)
  - JAQL est le « **high-level query language** » inclus dans BigInsights
  - Format d'échange de données léger
  - Utilisé pour échanger des données entre des programmes
- Récupérer le résultat depuis Hadoop via la fonction HDFS_READ

DB2 z/OS et le Big Data

Enrichir les applications DB2z avec les fonctions analytiques Big Data

- DB2 fournit les capacités d'une base de données ainsi que les connecteurs pour permettre aux applications DB2 d'accéder facilement et efficacement aux données dans Hadoop



Support de DB2 v11 pour Big Data

Objectif: Intégrer DB2 z/OS avec la plateforme IBM BigData (BigInsights) basée sur Hadoop et permettre aux applications traditionnelles DB2 z/OS d'accéder aux fonctionnalités analytiques Big Data

1) Les jobs analytiques peuvent être spécifiés en utilisant JSON Query Language (**Jaql**)
   1. Soumis pour exécution (JAQL_SUBMIT) sur la plateforme BigInsights
   2. **Résultats** stockés dans Hadoop Distributed File System (**HDFS**)

2) Une table UDF (HDFS_READ) lit le résultat du job analytique BigData depuis HDFS, et le stocke dans DB2 pour une utilisation ultérieure dans une requête SQL.
   - DB2 v11 supporte les « tables génériques » UDF, permttant d'utiliser ces fonctions

« Connecteurs » BigInsights pour DB2 for z/OS

- 2 fonctions DB2 « sample » :
  - JAQL_SUBMIT – Soumet un script JAQL pour exécution dans BigInsights depuis DB2
  - HDFS_READ – Récupère le résultat depuis les fichiers HDFS pour le stocker dans une table DB2 pour être utiliser par des requêtes SQL

- Notes:
  - Les fonctions sont développées par DB2 for z/OS
    - Livrées avec DB2 v11 dans prefix.SDSNLOD2
    - Les fonctions ne sont pas installées par défaut
  - Les fonctions et les exemples sont documentées par BigInsights
    - http://www.ibm.com/support/docview.wss?uid=swg27040438

JAQL_SUBMIT

Soumettre un script JAQL pour exécution dans BigInsights depuis DB2

```
SET RESULTFILE =
 JAQL_SUBMIT
  ('syslog = lines("hdfs:///idz1470/syslog3sec.txt");

    [localRead(syslog)->filter(strPos($,"$HASP373")>=0)->count()]->
     write(del(location="hdfs:///idz1470/iod00s/lab3e2.csv"));',

   'http://bootcamp55.democentral.ibm.com:14000/webhdfs/v1/idz1470/
    iod00s/lab3e2.csv?op=OPEN',


   'http://bootcamp55.democentral.ibm.com:8080',

   ''
  );
```

JAQL script containing the analysis

Intended HDFS file to hold the result

URL of the BigInsights cluster

options

Exemple HDFS_READ

Récupère le résultat depuis HDFS, et le stocke dans une table DB2 pour utilisation ultérieure avec des ordres SQL

```
SET RESULT_FILE = JAQL_SUBMIT(. . . . . );

SELECT BIRESULT.CNT FROM
TABLE(HDFS_READ(RESULT_FILE, '')

AS BIRESULT(CNT INTEGER);
```

URL of the CSV file to be read

options

Definition of the “table”, how to present the results to SQL

Exemple HDFS_READ

- Exemple de fichiers stocké dans HDFS

  1997,Ford, E350,"ac, abs, moon",3000.00

  1999,Chevy, "Venture ""Extended Edition""", ,4900.00

  1996,Jeep,Grand Cherokee,"MUST SELL! AC, moon roof, loaded",4799.00

- Exemple d'ordre SQL

```
SELECT * FROM TABLE (HDFS_Read('http://BI.foo.com/data/controller/dfs/file.csv',
                                '))
        AS X (YEAR INTEGER, MAKE VARCHAR(10), MODEL VARCHAR(30),
              DESCRIPTION VARCHAR(40), PRICE DECIMAL(8,2));
```

- Résultat

YEAR	MAKE	MODEL	DESCRIPTION	PRICE
1997	Ford	E350	ac, abs, moon	3000.00
1999	Chevy	Venture "Extended Edition"	(null)	4900.00
1996	Jeep	Grand Cherokee	MUST SELL! AC, moon roof, loaded	4799.00

Exemple de requête intégrée

```
INSERT INTO BI_TABLE (CNT)
(SELECT CNT FROM TABLE
(HDFS_READ
 (JAQL_SUBMIT
  ('syslog = lines("hdfs:///idz1470/syslog3sec.txt");

    [localRead(syslog)->filter(strPos($,"$HASP373")>=0)->count()]->
     write(del(location="hdfs:///idz1470/iod00s/lab3e2.csv"));' ,

   'http://bootcamp55.democentral.ibm.com:14000/webhdfs/v1/idz1470/
    iod00s/lab3e2.csv?op=OPEN',

   'http://bootcamp55.democentral.ibm.com:8080',
   ''
  ),
  ''
 )
)
AS BIGINSIGHTS(CNT INTEGER));
```

JAQL_SUBMIT can be embedded in HDFS_READ for a synchronous execute/read workflow

Agenda

- Contexte
- Cas d'utilisation
- DB2 z/OS et Hadoop
- **Connecteurs z pour Hadoop**

L'intégration des Big Data est critique pour le succès des implémentations Hadoop

La majorité des initiatives Hadoop impliquent de collecter, déplacer, transformer, nettoyer, intégrer, explorer, et analyser des volumes issus de sources de données et de types hétérogènes.

"Pour la majorité des clients **80%** *De l'effort de développement dans un projet Big Data va dans l'intégration des données*

…et seulement **20%** *Va dans l'analyse des données."*

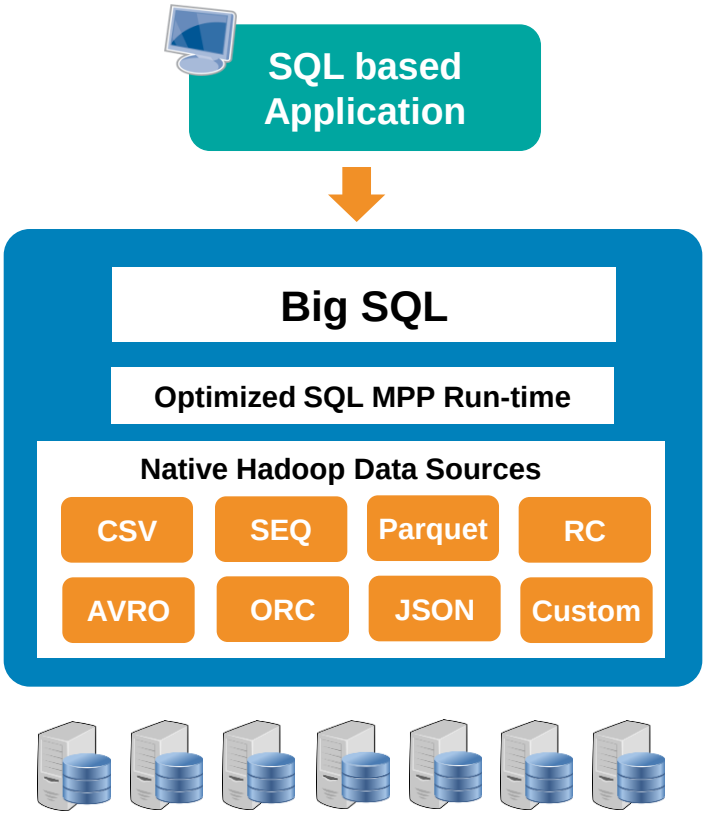
Extract, Transform, and Load Big Data With Apache Hadoop - Whitepaper:

https://software.intel.com/sites/default/files/article/402274/etl-big-data-with-hadoop.pdf

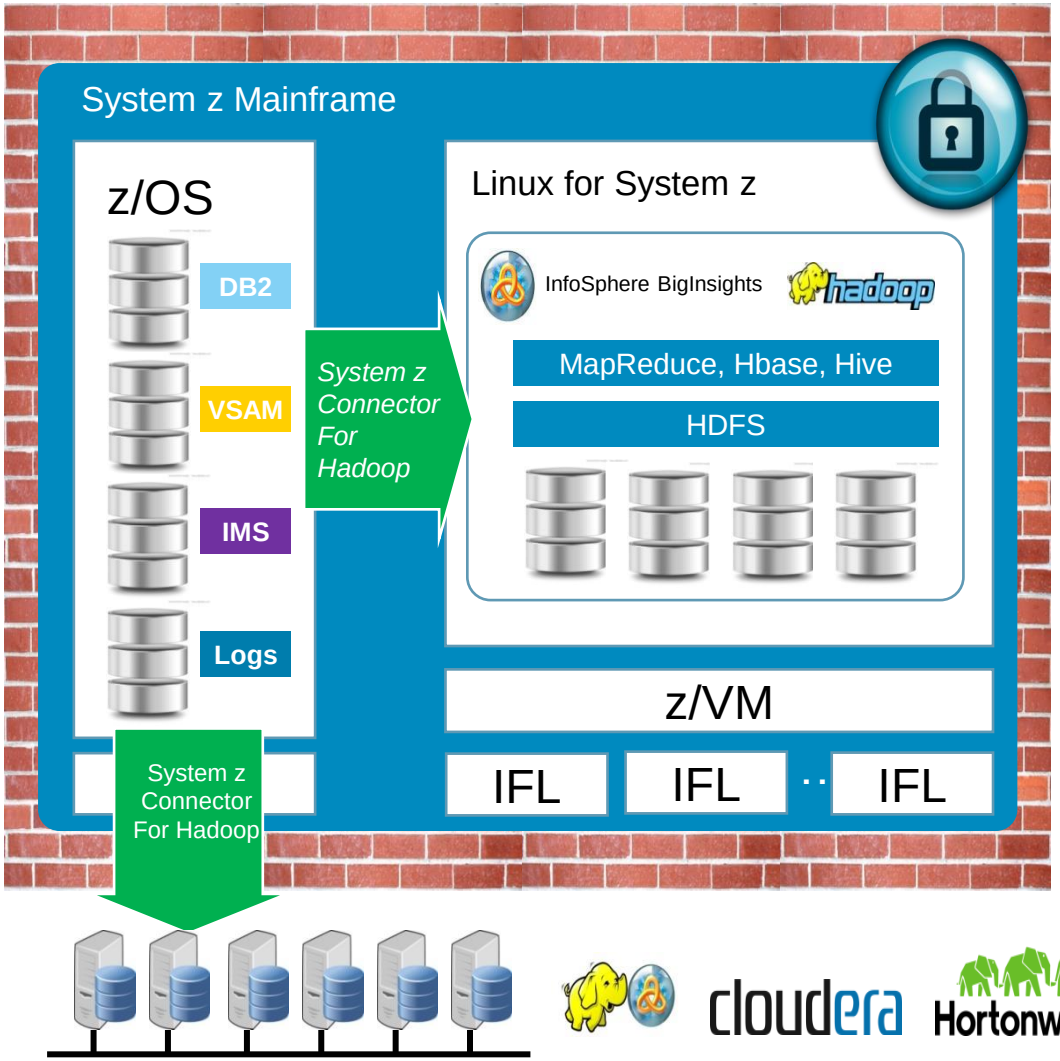
IBM InfoSphere BigInsights – Big SQL

Big SQL = Big Investment Protection

- SQL IBM pour Hadoop
  - Rend les données Hadoop accessibles par une plus grande population
  - Syntaxe familière, et connue par un large public
  - S'appuie sur les sources de données native Hadoop
- Complémente le Data Warehouse
  - Analyses exploratrices
  - Sandbox, Data Lake
- Inclus dans BigInsights
- Utilisation SQL familiers
  - Cognos, SPSS, Tableau, MicroStrategy



IBM InfoSphere System z Connector for Hadoop



- Enrichir les données Big Data avec les données z data avec Hadoop sur la plateforme de votre choix
  - ✓IBM System z pour la sécurité
  - ✓Power Systems
  - ✓Intel Servers
  - ✓SoftLayer
- Accès aux données par « Point and click » ou par batch
- Réduction des coûts de traitement & de stockage

29

IBM InfoSphere System z Connector for Hadoop

Fonctionnalités clés:

- Supporte plusieurs distributions Hadoop – sur z et à l'extérieur du z
- Plusieurs formats de sources de données: DB2, VSAM/QSAM, IMS, Logs
- Transfert de fichier par les HiperSockets et par une connexion 10 GbE
- Interface Drag-and-drop – pas de programmation
- Destinations multiples : Hadoop, Linux File Systems, Streaming endpoints
- Possibilité de définir différents profiles pour le transfert de données
- Interface de Streaming – filtre et transforme les données & les colonnes à a volée, sur la cible
- Canal sécurisé – Intégration à RACF

Merci !