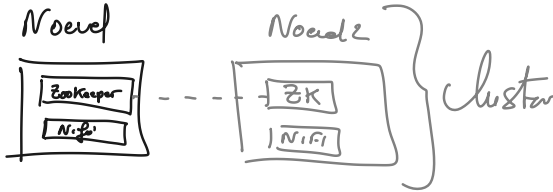
Apache Nifi

Présentation Hadoop

① installation "On Prem"



② Installe un cluster Hadoop
(HDInsight sous Azure)

Environnement:

- micro → nouvelles techno
  → java, sql, NoSql, ...
- mini → SAP, IBM, Cobol, ...
- HPC → calcul, simulation, ...
- info en grille
- Hadoop (OpenSource)

Historique de Hadoop :

~ 2000 Google → bcp de recherche

chercheurs → modèle Paradigme Map Reduce

objectif → calcul distribué

Pour Tester :

Hadoop → Très Complexe à installer

Cloudera, Hortonworks } sociétés qui font une distribution de Hadoop et vendent du service

file System : HDFS Hadoop File System

→ pas équ. ext3, glusterFS

→ encapsulé en ligne de Commande

Pas de point de montage

→ fichiers sous HDFS sont répliqués et Tronqués verticalement (CSV)

→ Calcul → MapReduce → Java

↙

Pig Ambari

Hive Spark (equ sge)

etc.... (ligne de commande)

Livraison "accessible" ⇒ DATABRICKS

- Environnement de dév en html5 (sans shell) → Jupyter Notebooks

↙

juste un éditeur dans le navigateur

DATABRICKS Community → gratuit pour faire du stockage / calcul.

Naissance de Hadoop

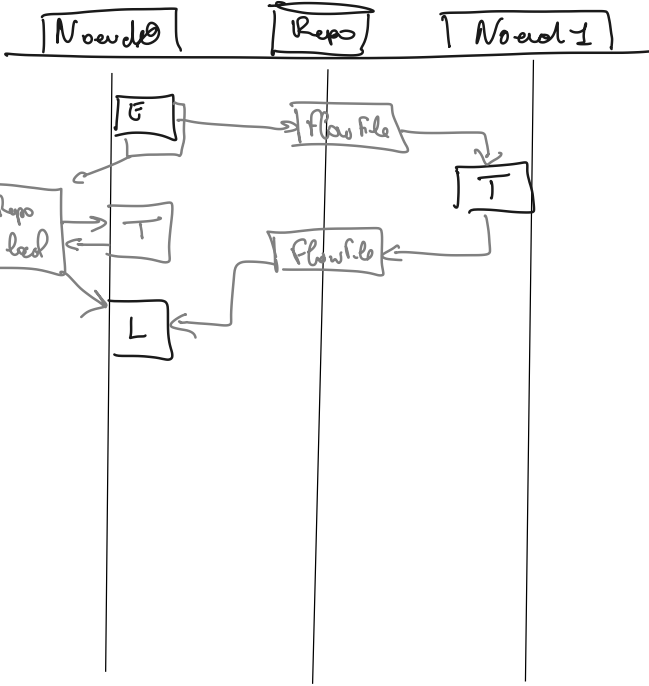
~2000 google → Map Reduce

↓

Doug Cutting → développé la solution en java

→ fonctionnel

cluster 3 machines } → 100 machines



graph.

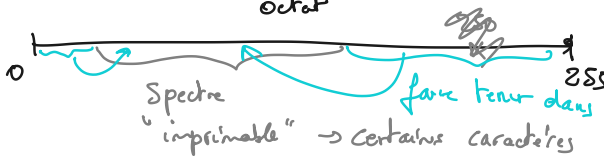
Encodage Avro → binaire, multiplateforme

Efficace

JSON, XML → pas de contenu binaire !

binaire → encodé/décodé en base 64

octet



8 bits → base 256 → 256 possibilités

6 bits → base 64 → 64 possib. Textuel

A - Z → 26

a - z → 26

0 - 9 → 10

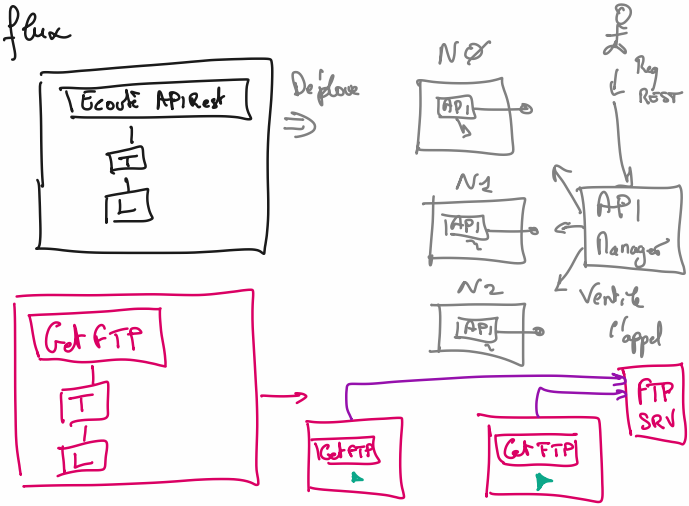
52 + 12 = 64

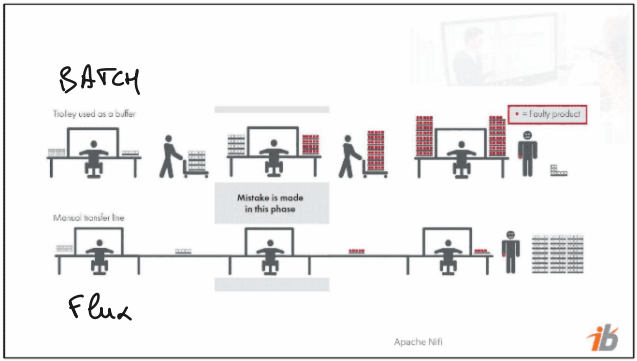
- 12 autres caractères.

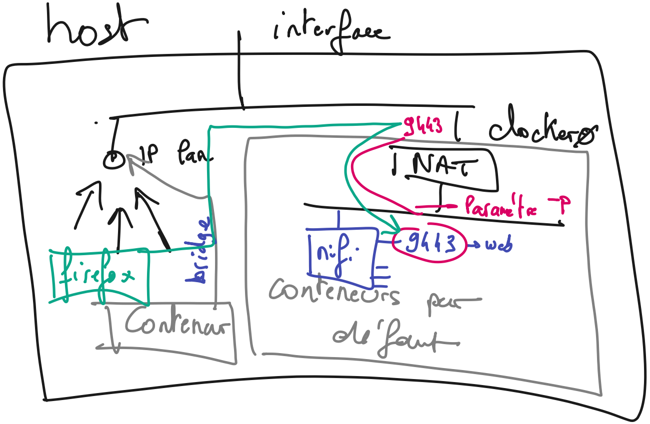
Différence entre données froides et données chaudes

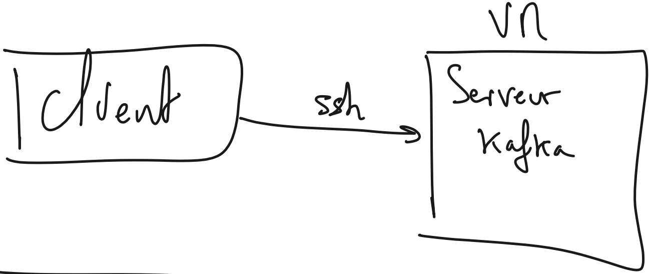
- Différence entre Flux, et Batch
- Batch (Talend, SSIS, ETL conventionnels)
  - Les opérations en batch s'effectuent régulièrement, et par lot.
  - Entre l'apparition de la donnée, et l'exécution du batch, la donnée a « refroidi », il y a une latence, il ne s'agit pas de temps réel
  - Au moment du batch, l'intégralité des données « côte à côte » sont disponibles
  - Le traitement n'est pas distribué
  - La transformation de données est aisée (opérations ETL)
- Flux (Nifi)
  - Les systèmes en flux fonctionnent en continu, en temps réel
  - Les données « côte à côte » ne sont pas disponibles
  - Les opérations sont distribuables (BigCompute)
  - Les transformations de données sont plus difficiles (il faut préférer le modèle ELT)

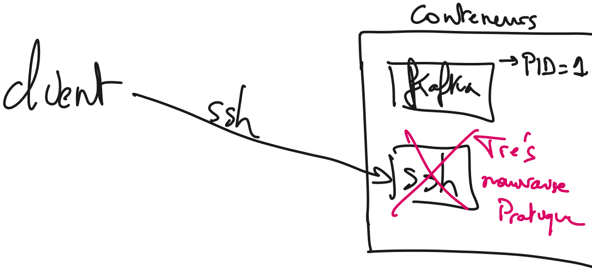
Apache Nifi

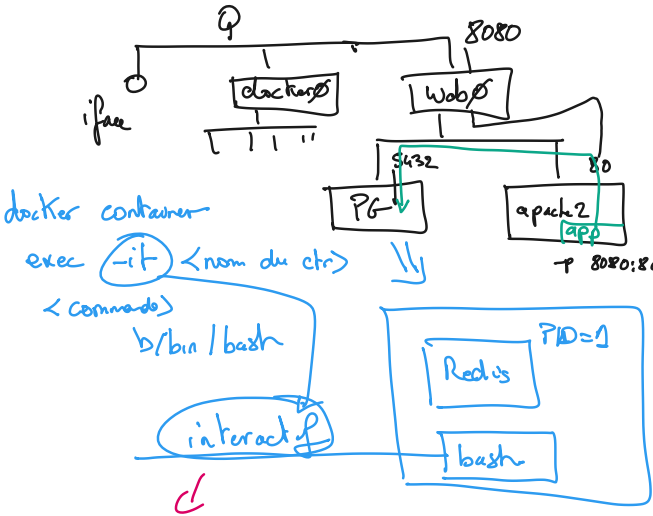



BATCH
Trolley used as a buffer
• = Faulty product
Mistake is made in this phase
Manual transfer line
Flux
Apache Nifi


host
interface
IP Pan
9443
docker0
NAT
Paramètre -p
nifi
9443 → web
bridge
firefox
Conteneur
conteneurs par défaut


VM
client
ssh
Serveur Kafka

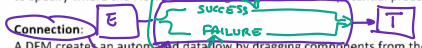

Conteneurs
client
ssh
Kafka
→ PID=1
ssh
Très mauvaise Pratique



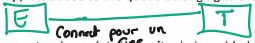
équivalent du ssh en conteneurs

information from FlowFiles.

Relationship:
Each Processor has zero or more Relationships defined for it. These Relationships are named to indicate the result of processing a FlowFile. After a Processor has finished processing a FlowFile, it will route (or "transfer") the FlowFile to one of the Relationships. A DFM is then able to connect each of these Relationships to other components in order to specify where the FlowFile should go next under each potential processing result.



Connection:
A DFM creates an automated dataflow by dragging components from the Components part of the NiFi toolbar to the canvas and then connecting the components together via Connections. Each connection consists of one or more Relationships. For each Connection that is drawn, a DFM can determine which Relationships should be used for the Connection. This allows data to be routed in different ways based on its processing outcome. Each connection houses a FlowFile Queue. When a FlowFile is transferred to a particular Relationship, it is added to the queue belonging to the associated Connection.



Controller Service:
Controller Services are extension points that, after being added and configured by a DFM in the User Interface, will start up when NiFi starts up and provide information for use by other components (such as processors or other controller services). A common Controller Service used by several components is the StandardSSLContextService. It provides the ability to configure keystore and/or truststore properties once and reuse that configuration throughout the application. The idea is that, rather than configure this information in every processor that might need it, the controller service provides it for any processor to use as needed.

Reporting Task:
Reporting Tasks run in the background to provide statistical reports about what is happening in the NiFi instance. The DFM adds and configures Reporting Tasks in the User Interface as desired. Common reporting tasks include the ControllerStatusReportingTask, MonitorDiskUsage reporting task, MonitorMemory reporting task, and the StandardGangliaReporter.

Funnel:
A funnel is a NiFi component that is used to combine the data from several Connections into a single Connection.

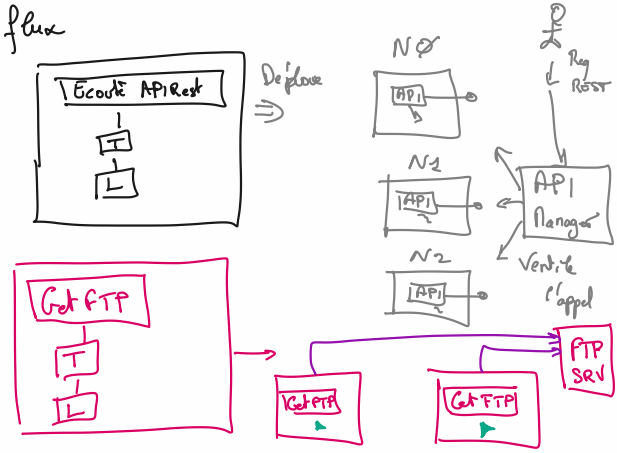
Process Group:
When a dataflow becomes complex, it often is beneficial to reason about the dataflow at a higher, more abstract level. NiFi allows multiple components, such as Processors, to be grouped together into a Process Group. The NiFi User Interface then makes it easy for a DFM to connect together multiple Process Groups into a logical dataflow, as well as allowing the DFM to enter a Process Group in order to see and manipulate the components within the Process Group.

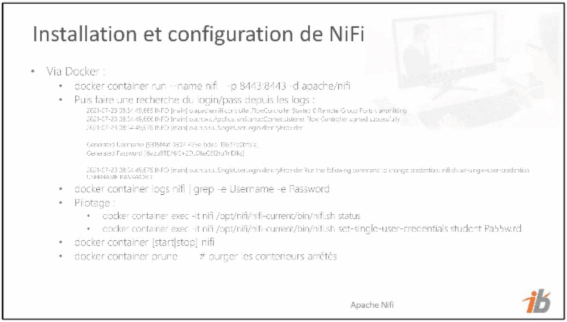
7

Différence entre données froides et données chaudes

- Différence entre Flux, et Batch
- Batch (Talend, SSIS, ETL conventionnels)
  - Les opérations en batch s'effectuent régulièrement, et par lot.
  - Entre l'apparition de la donnée, et l'exécution du batch, la donnée a « refroidi », il y a une latence, il ne s'agit pas de temps réel
  - Au moment du batch, l'intégralité des données « côte à côte » sont disponibles
  - Le traitement n'est pas distribué
  - La transformation de données est aisée (opérations ETL)
- Flux (Nifi)
  - Les systèmes en flux fonctionnent en continu, en temps réel
  - Les données « côte à côte » ne sont pas disponibles
  - Les opérations sont distribuables (BigCompute)
  - Les transformations de données sont plus difficiles (il faut préférer le modèle ELT)

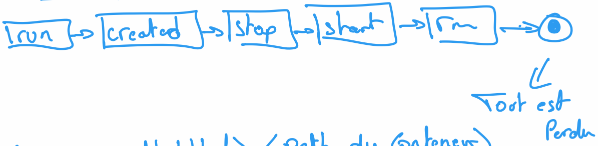
Apache Nifi



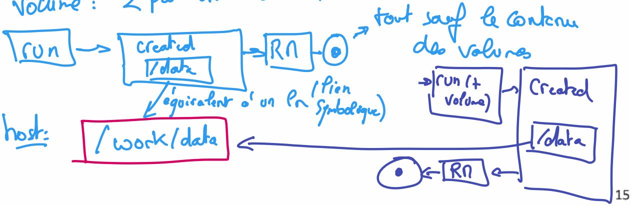


Volumes Docker

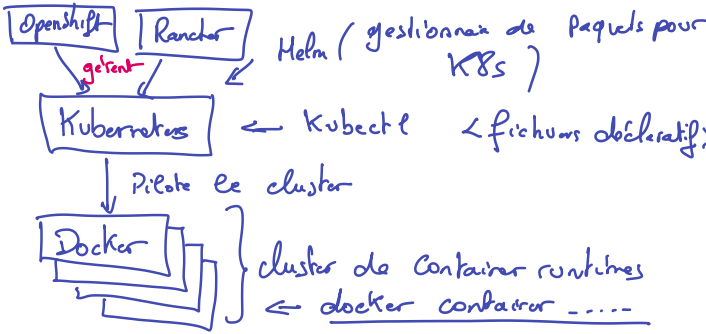
Conteneurs



volume : <path du host>:<path du conteneur>



Docker → Pilotage mono-machine



http://localhost:18080/nifi-registry

Ajouter le serveur registry
L'enregistrer au niveau du cluster menu principal / controller setting / registry client
Clic droit dans le canevas, et activer la gestion de version

Aller dans l'environnement de destination, ajouter aussi le registry client
Créer un ProcessGroup qui sera importé depuis le registry

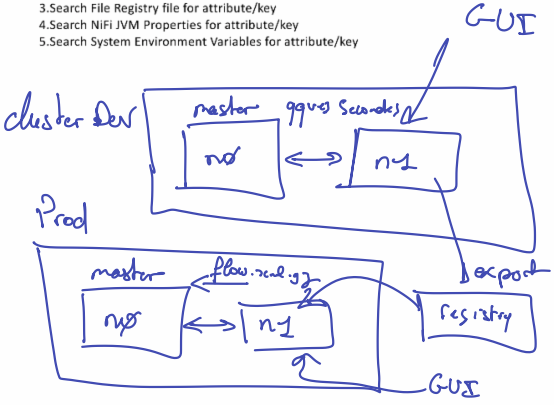
Utiliser le langage d'expressions EL

- https://docs.cloudera.com/cfm/2.1.1/nifi-express-guide/topics/nif-expression-language-overview.html
- Vous pouvez faire référence à des attributs, des méthodes, des tests conditionnels, via le langage d'expressions.
- Par exemple :
  - ${hostname():toUpper()}
  - ${filename:equals(${uuid})}
- Evaluations booléennes
  - ${anyAttribute("abc", "xyz"):contains("bye")}
- Les valeurs référencées sont recherchées par hiérarchie (voir les commentaires)
- La référence des fonctions :
  - https://nifi.apache.org/docs/nifi-docs/html/record-path-guide.html#standalone-functions

Apache Nifi

Hierarchie de recherche des valeurs :

1.Search FlowFile for attribute/key
2.Search Process Group Variables for attribute/key
3.Search File Registry file for attribute/key
4.Search NiFi JVM Properties for attribute/key
5.Search System Environment Variables for attribute/key



Utiliser le langage d'expressions EL

- https://docs.cloudera.com/cfm/2.1.1/nifi-express-guide/topics/nif-expression-language-overview.html
- Vous pouvez faire référence à des attributs, des méthodes, des tests conditionnels, via le langage d'expressions.
- Par exemple :
  - ${hostname():toUpper()}
  - ${filename:equals(${uuid})}
- Evaluations booléennes
  - ${anyAttribute("abc", "xyz"):contains("bye")}
- Les valeurs référencées sont recherchées par hiérarchie (voir les commentaires)
- La référence des fonctions :
  - https://nifi.apache.org/docs/nifi-docs/html/record-path-guide.html#standalone-functions

Apache Nifi

Hierarchie de recherche des valeurs :

1.Search FlowFile for attribute/key
2.Search Process Group Variables for attribute/key
3.Search File Registry file for attribute/key
4.Search NiFi JVM Properties for attribute/key
5.Search System Environment Variables for attribute/key