

**INSA**

INSTITUT NATIONAL  
DES SCIENCES  
APPLIQUÉES  
TOULOUSE

**ThalesAlenia**  
*a Thales / Leonardo company*  
**Space**

# Soutenance Projet Intégrateur 5SDBD – Mega Team

Adrian MEGA, Hao HU, Mohamed EL FILALI



Université  
de Toulouse



**1. Architecture du projet**

**2. Enregistrement des données**

**3. Apprentissage Supervisé**

**4. Ajout de nouvelles données**

**5. Démonstration & Conclusion**

**1. Architecture du projet**

**2. Enregistrement des données**

**3. Apprentissage Supervisé**

**4. Ajout de nouvelles données**

**5. Démonstration & Conclusion**

## 1. Architecture du projet



**MINIO**



elasticsearch



kibana

APACHE  
**Spark**<sup>™</sup>



## 1. Architecture du projet

# MINIO



- 4 instances pour stocker les images
- Les images sont splittées dans les instances
- Script exécuté par le host



split\_data.py

## 1. Architecture du projet



elasticsearch

- Données indexées avec le split
- Rendues accessibles pour Spark
- Données consultables avec Kibana



kibana



split\_data.py

## 1. Architecture du projet



- Sert à faire transiter les données
- Un worker par instance Minio
- Le host récupère et envoie les données



learning.py



**1. Architecture du projet**

**2. Enregistrement des données**

**3. Apprentissage Supervisé**

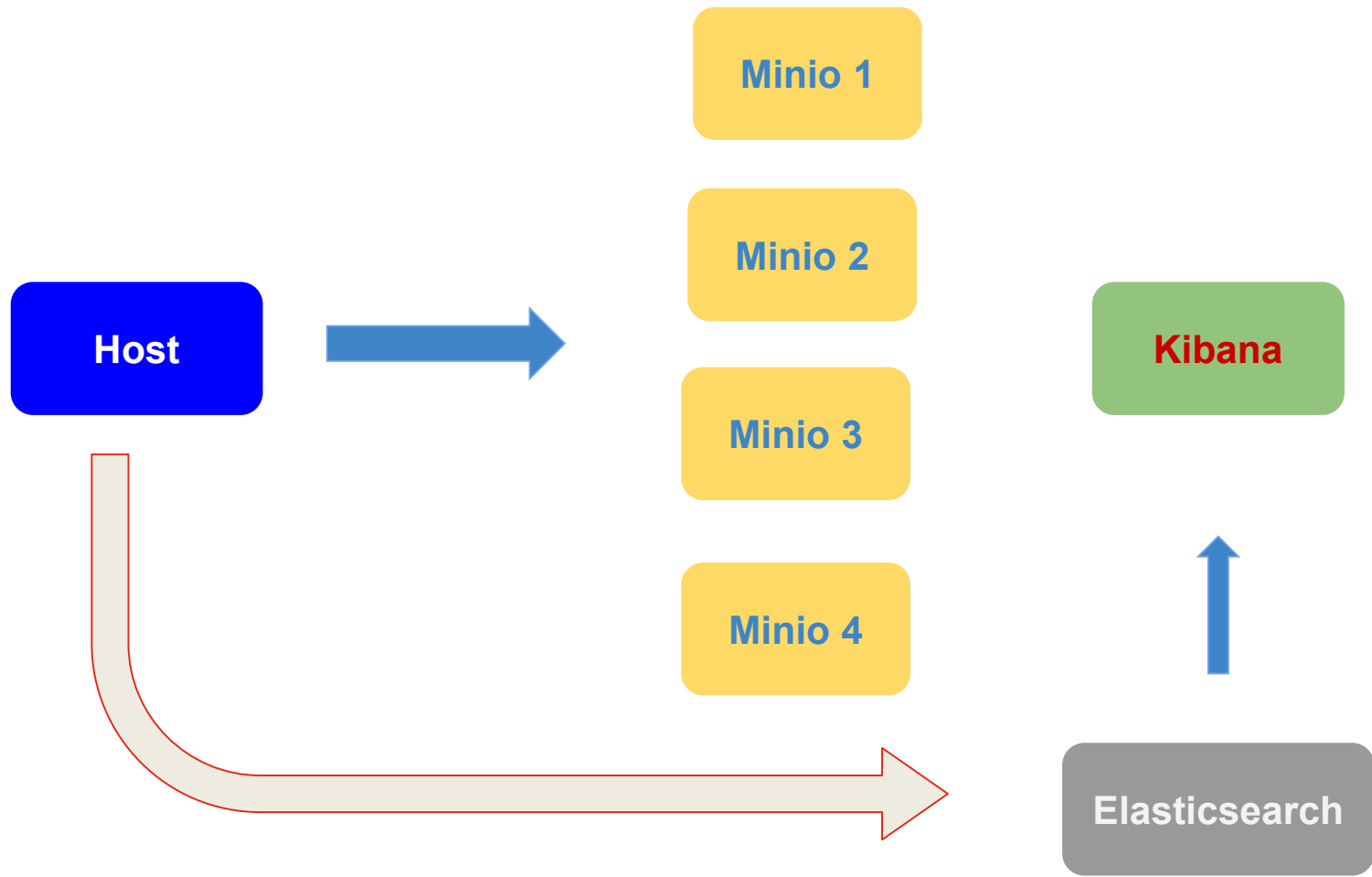
**4. Ajout de nouvelles données**

**5. Démonstration & Conclusion**

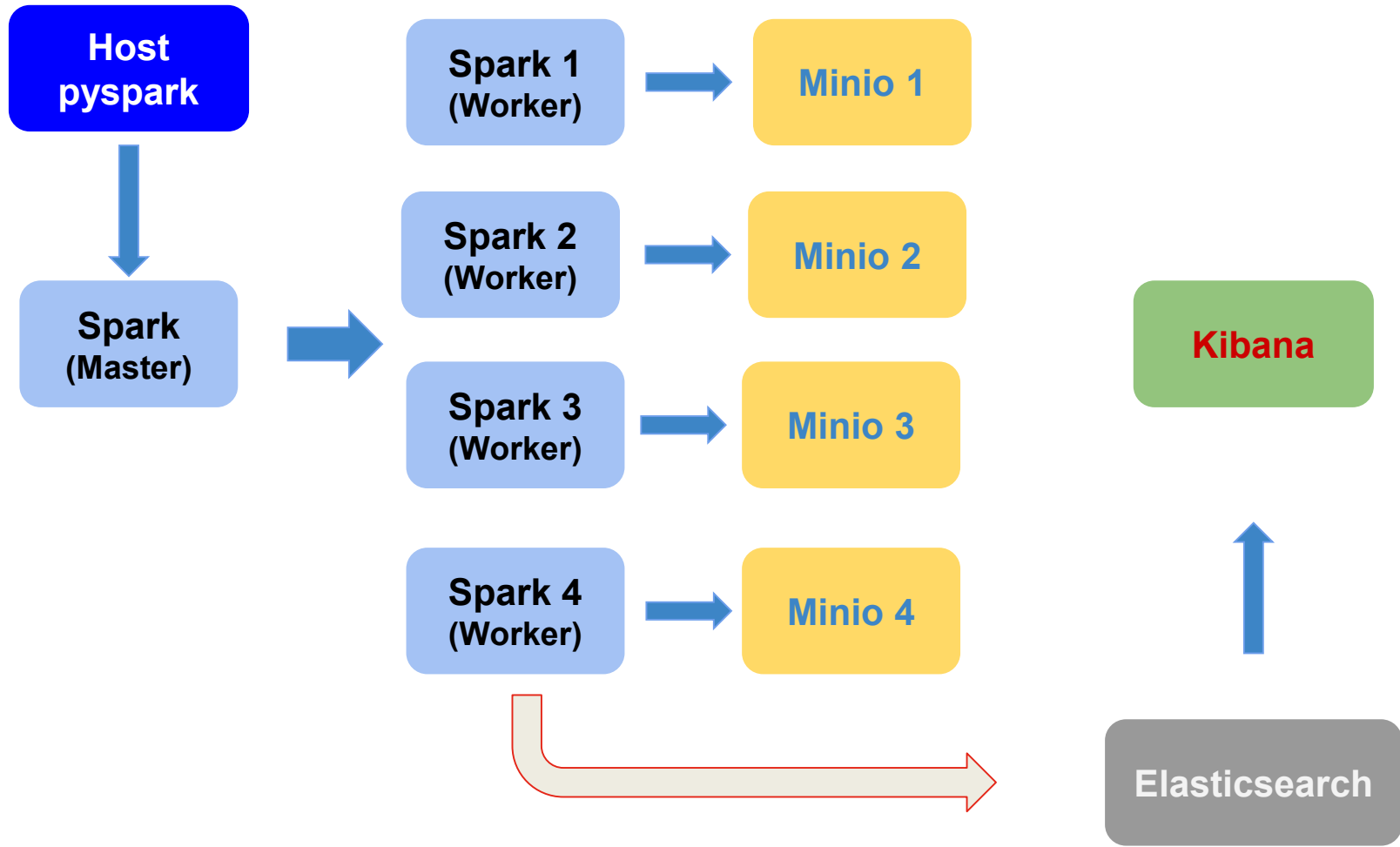


## 2. Enregistrement des données

Des données stockées en mode distribué



## 2. Enregistrement des données



**1. Architecture du projet**

**2. Enregistrement des données**

**3. Apprentissage Supervisé**

**4. Ajout de nouvelles données**

**5. Démonstration & Conclusion**

Des informations:



- CNN (Convolutional Neural Network)
- AlexNet (le meilleur algorithme en ImageNet 2012), 8 layers.
- TensorFlow + Keras
- Taille de données entraînement: **20000**
- Temps d'entraînement: **un peu plus de 20min pour 8 epochs**



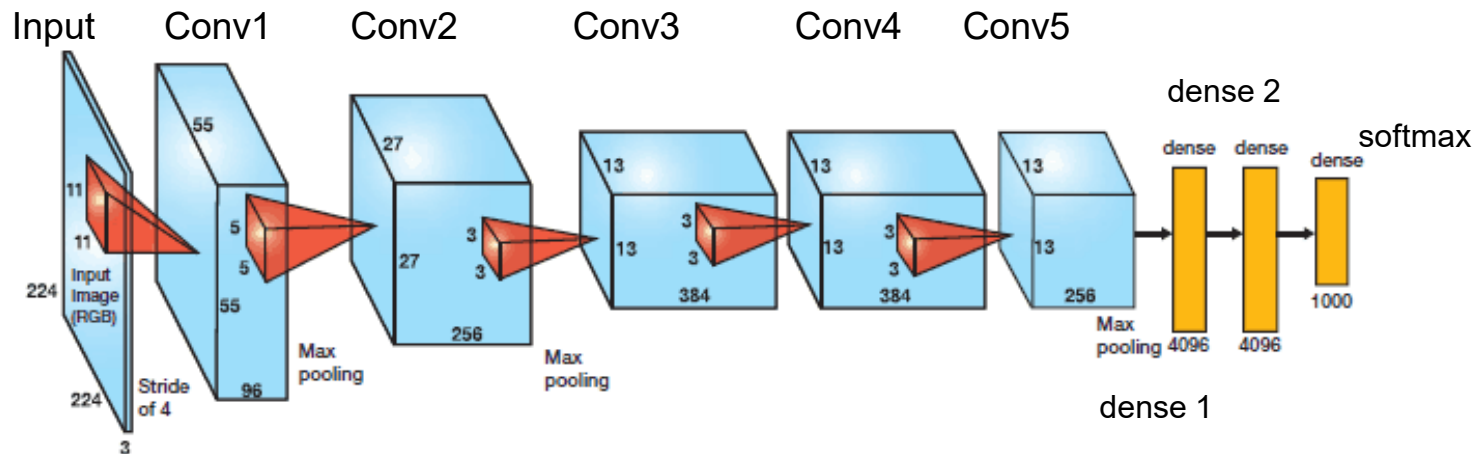
### 3. Apprentissage supervisé

Des défauts pour le “All-connected Layer(Core Layer)” en classification d’images:

**Beaucoup trop de paramètres !!**

- Plus de temps de calcul -> pas possible d’ajouter plus de couches
- facile de “overfitting”

#### AlexNet (2012)



### 3. Apprentissage supervisé

Paramètres :

- training set: 20000, test set: 1000
- epoch 8, fonction de loss: categorical\_crossentropy, optimizer: SGD

Le résultat:

- 80.7% de rate d'accuracy pour des données validations.
- 81.9% de rate d'accuracy pour des données test.

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
1: bash
Epoch 1/8
2019-01-24 22:36:08.480806: W tensorflow/core/framework/allocator.cc:122] Allocation of 33554432 exceeds 10% of system memory.
2019-01-24 22:36:08.569882: W tensorflow/core/framework/allocator.cc:122] Allocation of 33554432 exceeds 10% of system memory.
2019-01-24 22:36:08.569882: W tensorflow/core/framework/allocator.cc:122] Allocation of 33554432 exceeds 10% of system memory.
2019-01-24 22:36:08.653922: W tensorflow/core/framework/allocator.cc:122] Allocation of 16777216 exceeds 10% of system memory.
2019-01-24 22:36:08.864214: W tensorflow/core/framework/allocator.cc:122] Allocation of 16777216 exceeds 10% of system memory.
20000/20000 [=====] - 161s 8ms/step - loss: 1.1948 - acc: 0.5712 - val_loss: 0.8650 - val_acc: 0.7330
Epoch 2/8
20000/20000 [=====] - 160s 8ms/step - loss: 0.9321 - acc: 0.7076 - val_loss: 0.7455 - val_acc: 0.8030
Epoch 3/8
20000/20000 [=====] - 165s 8ms/step - loss: 0.8081 - acc: 0.7587 - val_loss: 0.6457 - val_acc: 0.7950
Epoch 4/8
20000/20000 [=====] - 170s 9ms/step - loss: 0.7368 - acc: 0.7793 - val_loss: 0.6626 - val_acc: 0.7720
Epoch 5/8
20000/20000 [=====] - 171s 9ms/step - loss: 0.6858 - acc: 0.7887 - val_loss: 0.6483 - val_acc: 0.7740
Epoch 6/8
20000/20000 [=====] - 231s 12ms/step - loss: 0.6516 - acc: 0.7993 - val_loss: 0.9218 - val_acc: 0.6890
Epoch 7/8
20000/20000 [=====] - 192s 10ms/step - loss: 0.6276 - acc: 0.8019 - val_loss: 0.5908 - val_acc: 0.8000
Epoch 8/8
20000/20000 [=====] - 191s 10ms/step - loss: 0.6025 - acc: 0.8077 - val_loss: 0.6003 - val_acc: 0.8190
1000/1000 [=====] - 2s 2ms/step
('Test loss:', 0.6002827582359314)
('Test accuracy:', 0.819)
nwpuhh@nwpuhh-ThinkPad-X230:~/Workplace/TASImageProject/INSA_data_images$
15 64-bit 0 0 0 Ln 131, Col 20 Spaces: 8 UTF-8 LF Python

```

**1. Architecture du projet**

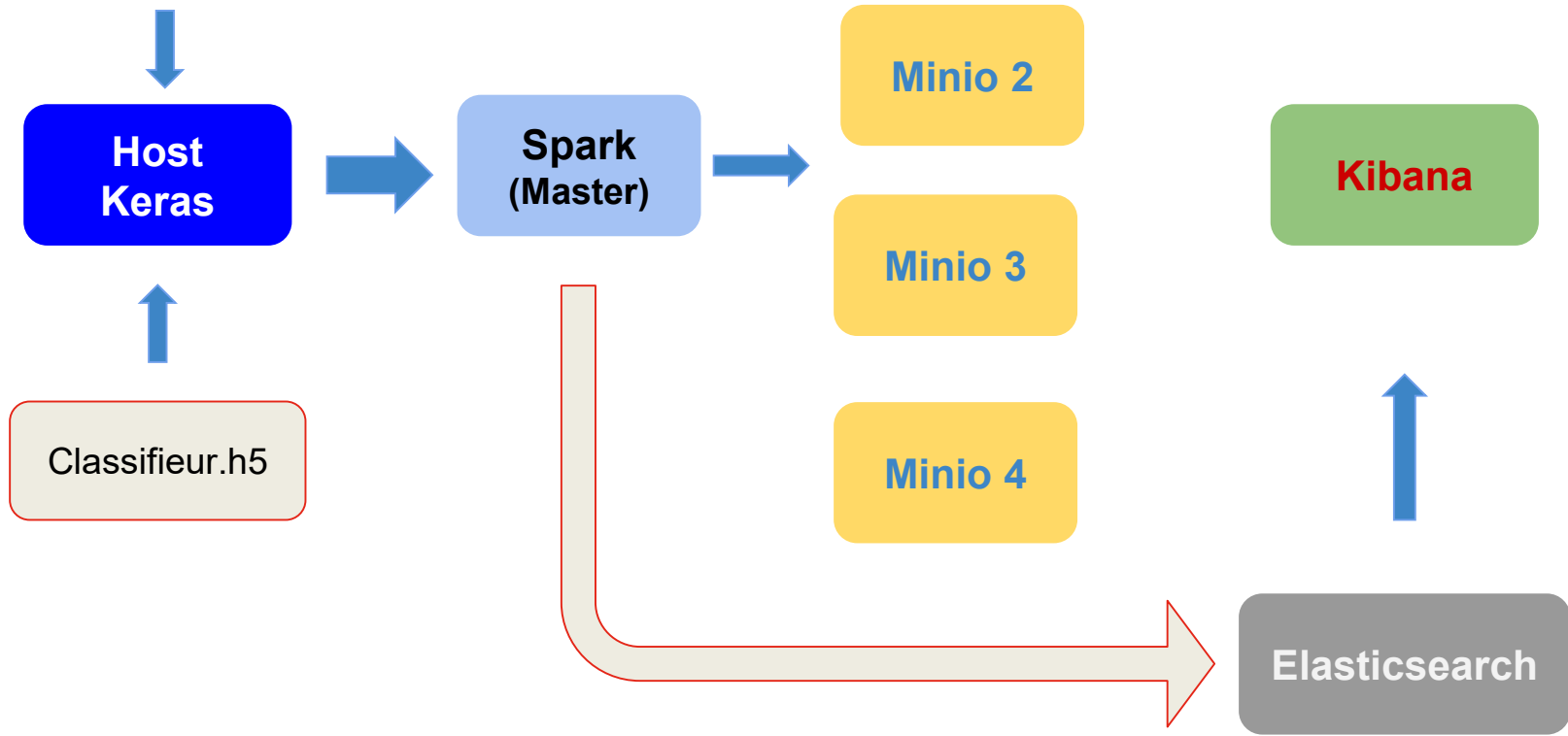
**2. Enregistrement des données**

**3. Apprentissage Supervisé**

**4. Ajout de nouvelles données**

**5. Démonstration & Conclusion**

## 4. Ajout de nouvelles données





**1. Architecture du projet**

**2. Enregistrement des données**

**3. Apprentissage Supervisé**

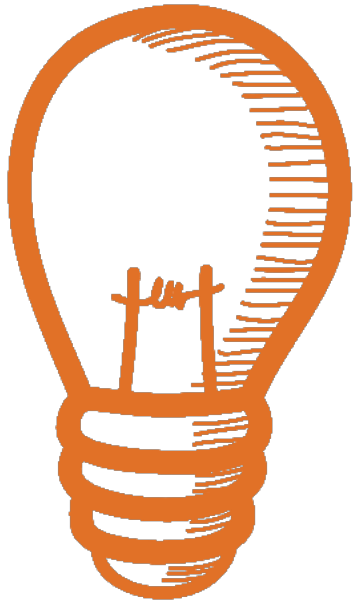
**4. Ajout de nouvelles données**

**5. Démonstration & Conclusion**



# Aperçu du Projet

## 5. Démonstration & Conclusion



- Travail sur un projet de grande envergure
- Application des connaissances
- Portainer
- Manipulation de grosses données (TensorFlow + Keras)
- Des algorithmes sur Deep Learning

**INSA**

INSTITUT NATIONAL  
DES SCIENCES  
APPLIQUÉES  
TOULOUSE

**ThalesAlenia**  
*a Thales / Leonardo company* **Space**

**Merci pour votre attention !**  
**Soutenance Projet Intégrateur**

**Adrian MEGA, Hao HU, Mohamed EL FILALI**



Université  
de Toulouse