Thales Services Numériques

COMET

Alain MONTMORY
Antoine TRAN

31 Mai 2022

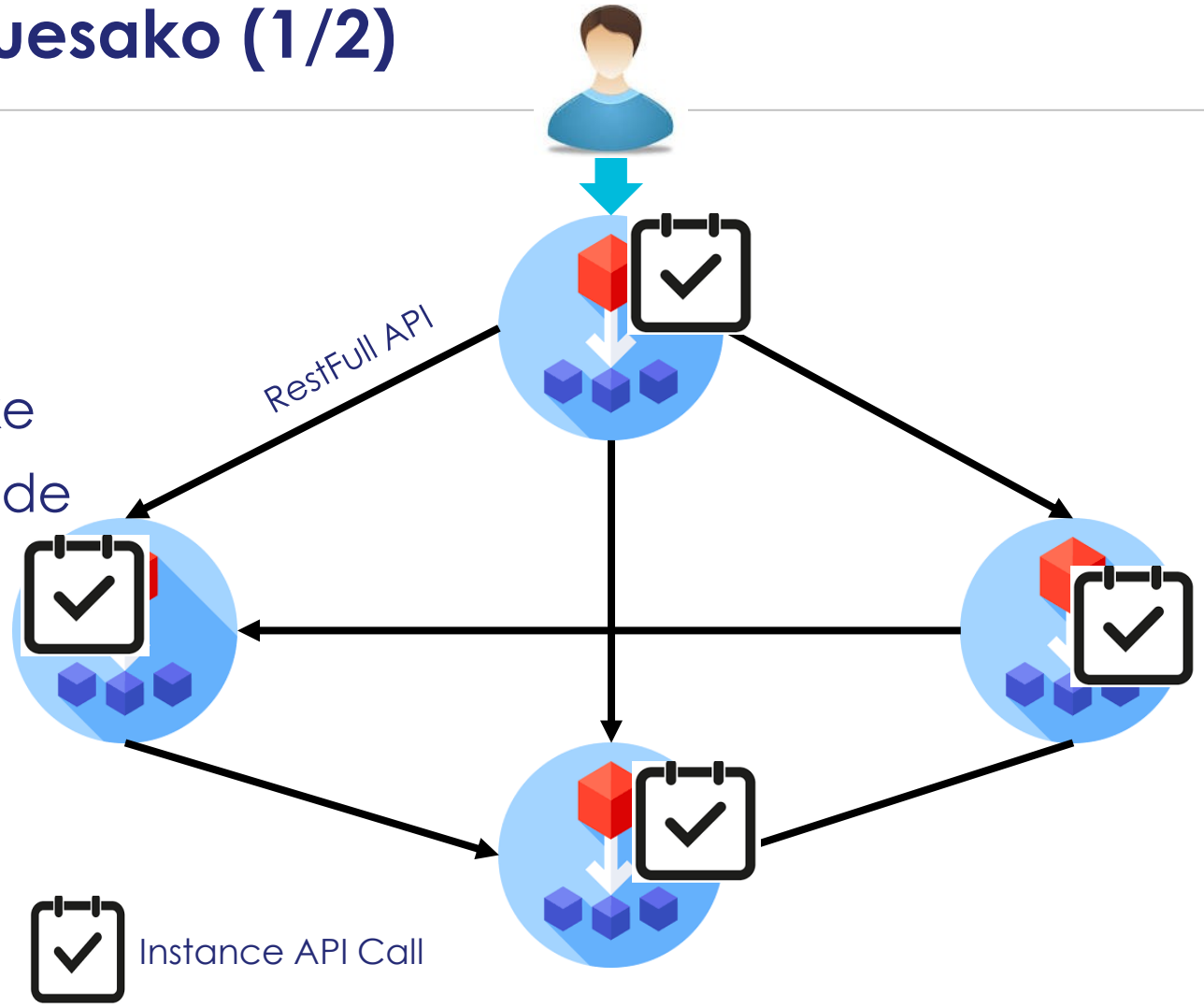
Agenda

- Micro Service Choregraphié? Quesako
- L'application au projet L2PF
- L'apport de Kubernetes
- Les bénéfices en termes de stabilité et d'explotation
- Vos questions ?

Micro Service Choregraphié? Quesako (1/2)

L'architecture classique des Micro Services

- Des interfaces REST entre les MS
- Une intégration qui peut être complexe
- Des tests d'intégration qui nécessitent de simuler plusieurs interfaces
- Des failures potentiellement difficiles à diagnostiquer

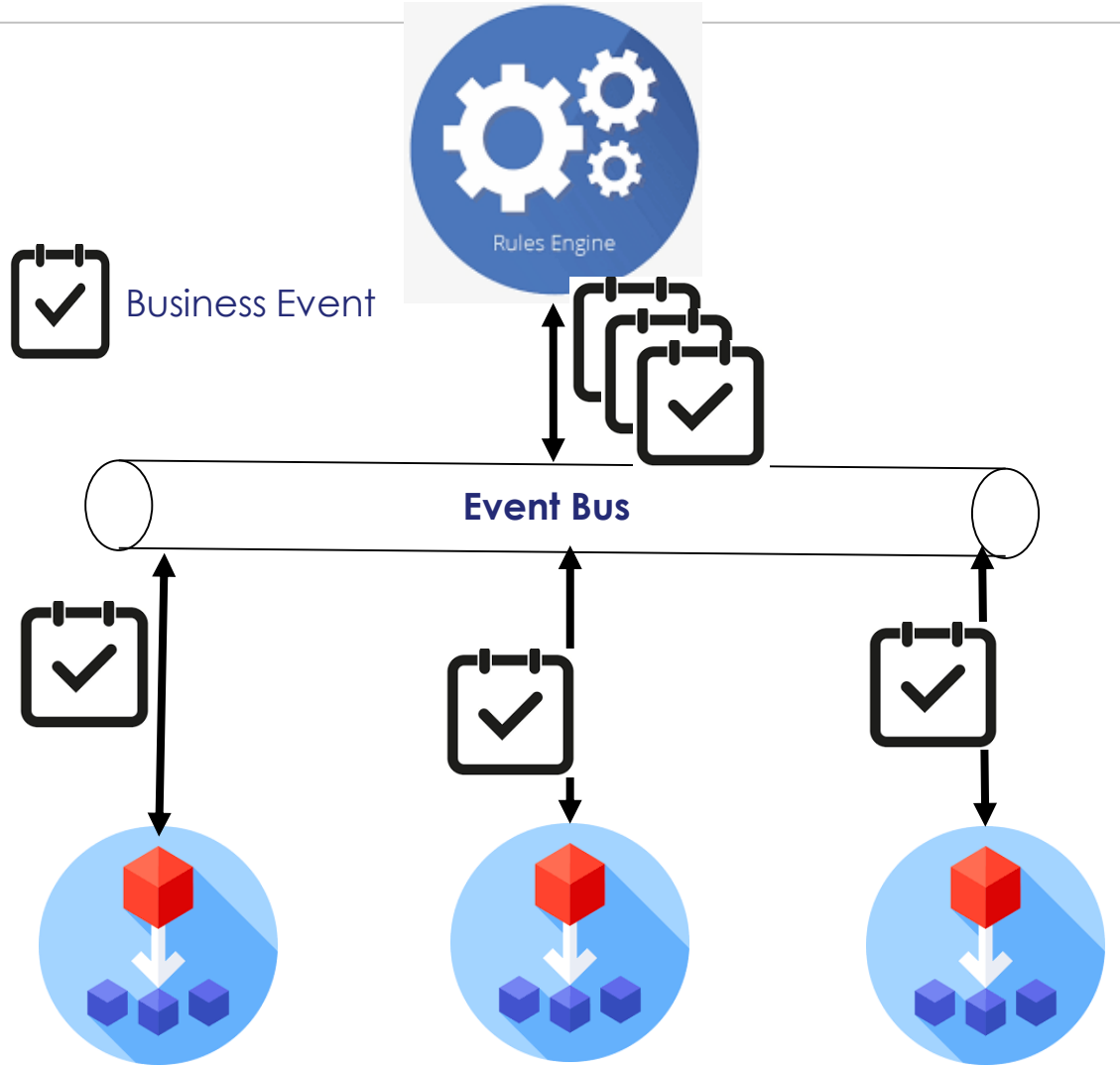


On peut arriver à un effet plat de spaghettis dans un système complexe.

Micro Service Choregraphié? Quesako (2/2)

Une architecture chorégraphiée

- Chaque MS s'interface physiquement avec un EventBus (EB)
- Chaque MS lit en entrée sur l'EB les tâches qui lui sont confiées
- Publie en sortie sur l'EB l'état auquel il est parvenu
- Un scheduler (règles business) analyse les états entrants et déclenche les états sortant consommés par les MS



Un plan integration plus simple à définir, des Evt faciles à simulés.

L'application au projet L2PF (1/2)

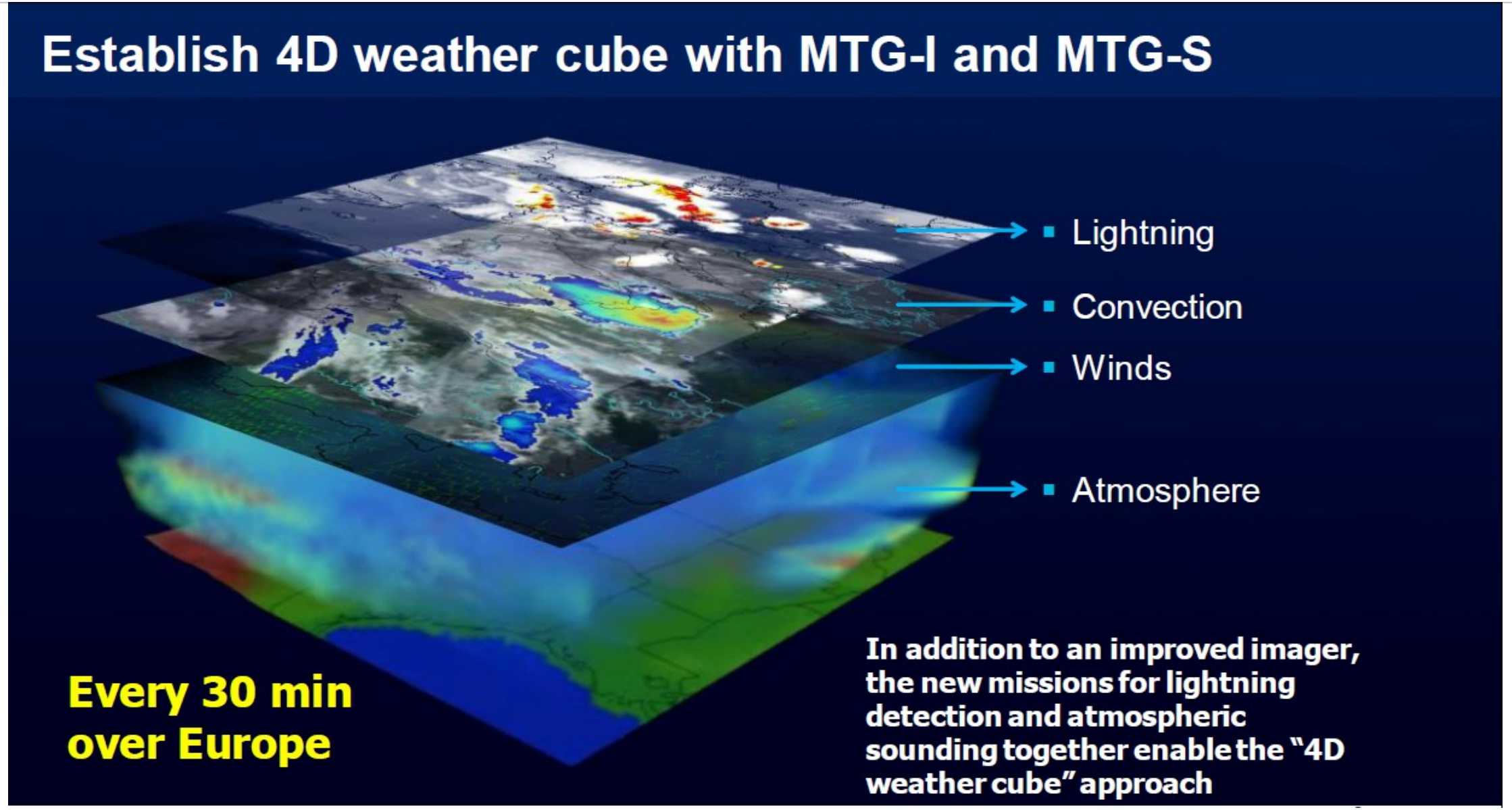
EUMETSAT The European Organisation for the Exploitation of Meteorological Satellites

MTG – Meteosat Third Generation
Geostationary orbit satellites are vital for short range weather **forecasts** up to **a few hours** (nowcasting)

An huge enhancement regarding previous generation

- 60 x more data downloaded by the satellites
- Completly new instrument Light Imager to detect Storm and Lightning for Aircraft
- FCI, LI, IRS and UVN instruments allow to establish a « 4D weather cube »

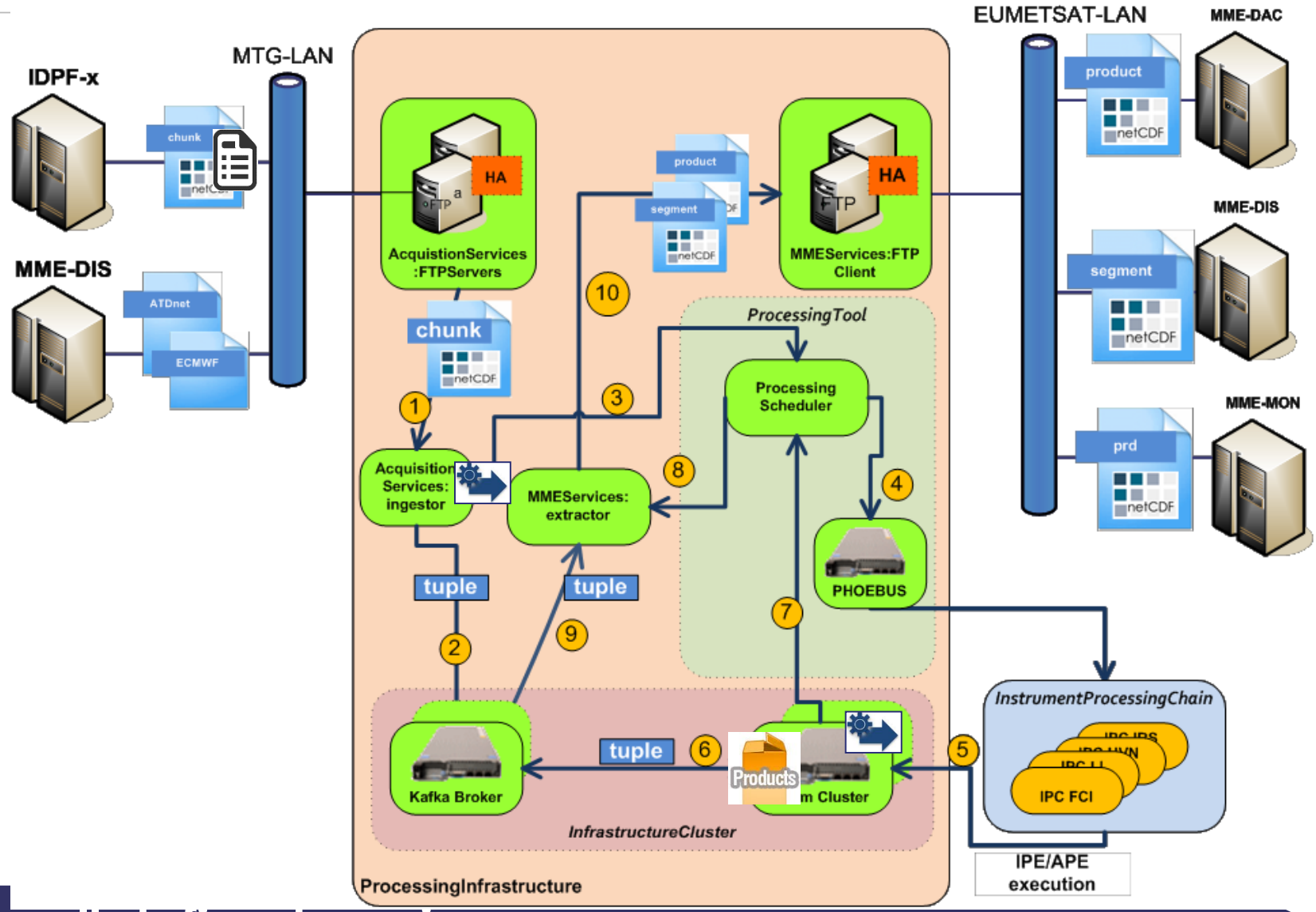
L'application au projet L2PF (2/2) (source EUMETSAT)



L2PF General overview

The Goal of the PI design is to:

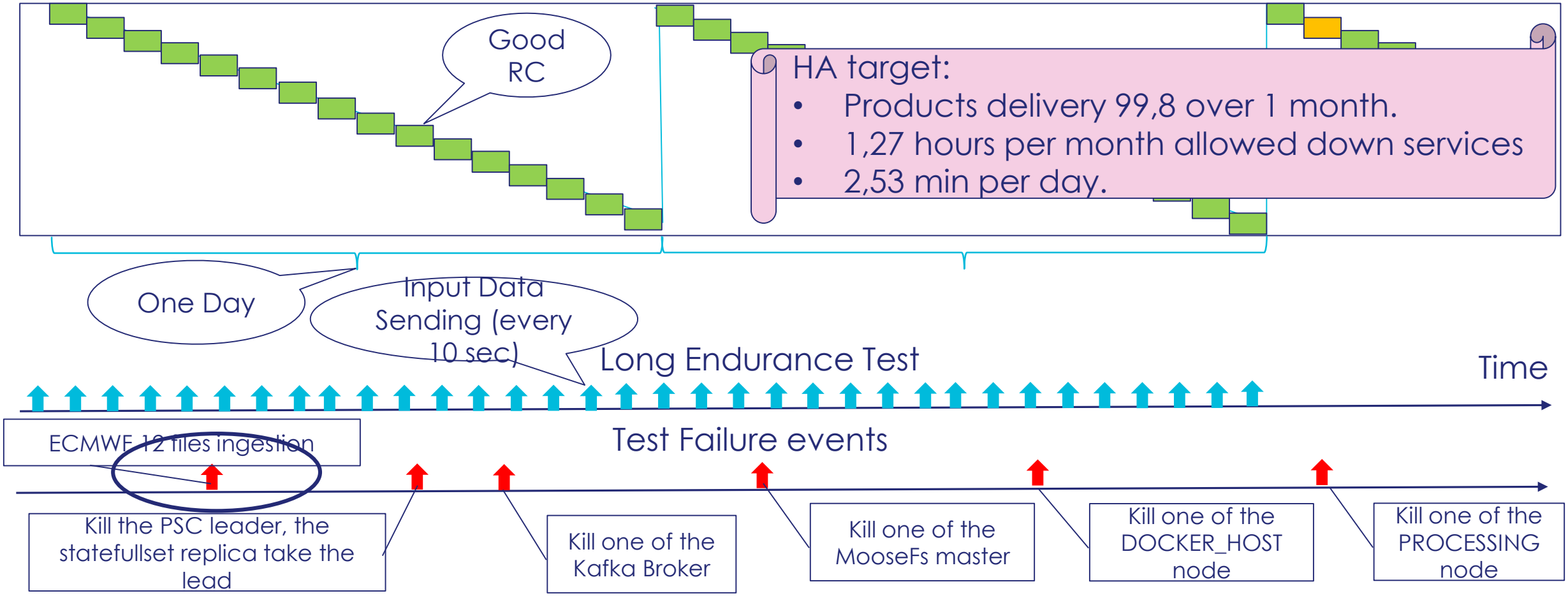
- to avoid process fork on each APE submission
- to split I/O load all along the timeline
- to minimize the time lost in pooling
- To allow a streaming data flow policy



A quite simple design with well defined services

Le Test de High Availability “End To End” .

Production chain is resilient against failure of each individual component:



Kubernetes offers all the mechanism needed to build a Highly Available System

Comment faire pour que l'ingestion de 12x1,3 GB ne perturbe pas le TR ?

Elasticité basé sur la CPU

- K8S offre la possibilité de mettre en place des Horizontal Pod Autoscaler basés sur la CPU => Ca marche mais c'est pas assez réactif (1 à 2 min pour scaler)

Elasticité sur le nombre de msg en attente ds la queue de l'EB

- Des points de mesures mécaniques peuvent être mis en œuvre sur l'EB et provoquer la mise en œuvre de réplicats => Ca marche mais c'est pas assez réactif (30 sec à 1 min pour scaler)

Elasticité sémantique

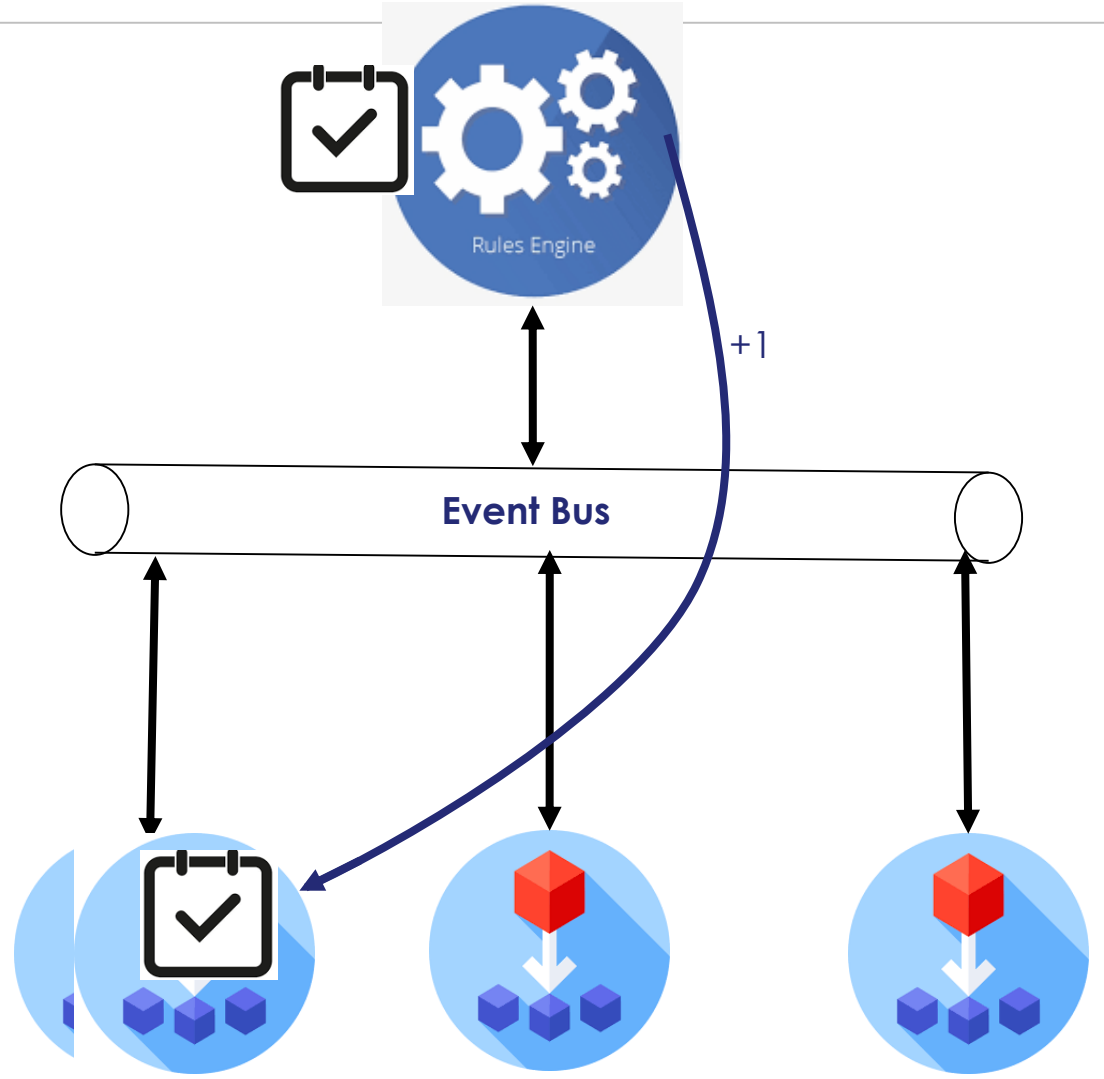
- Le Rule Engine peut déterminer selon la nature des données qu'il faut adapter le nombre des « Ingestors » => Ca marche et c'est assez réactif (2 sec pour scaler)
- Il faut prévoir dans vos composants de terminer l'action en cours proprement(sigTerm)

Tout celà est possible grâce à la capacité de K8S de démarrer très rapidement des réplicats.

L'elasticité Mise En place

Basée sur les busines Evt apparaissant sur l'EB:

- 1 nvx fichier ECMWF a été reçu en entrée du system « FileAvailableEvt »
- Le rule engine demande un scale + au niveau des « Ingestors »
- « l'ingestor » lancé ingère le fichier ECMWF
- Lorsque l'ingestion est finie l'ingestor publie un Evt « FileIngested »
- Le rule Engine fait un scale down de (-1) sur les Ingestors.



THALES

Building a future we can all trust

Q&A

www.thalesgroup.com