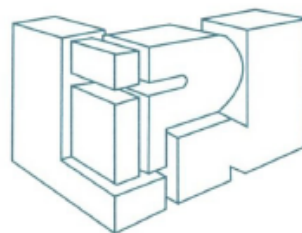




Laboratoire LaTICE
Univ. de Tunis



Laboratoire d' Informatique
de Paris Nord (LIPN)

INRIA LYON
Avalon Team



BONJOURGRID : VERSION ORIENTÉE DONNÉE & MAPREDUCE SÉCURISÉ

Heithem Abbes

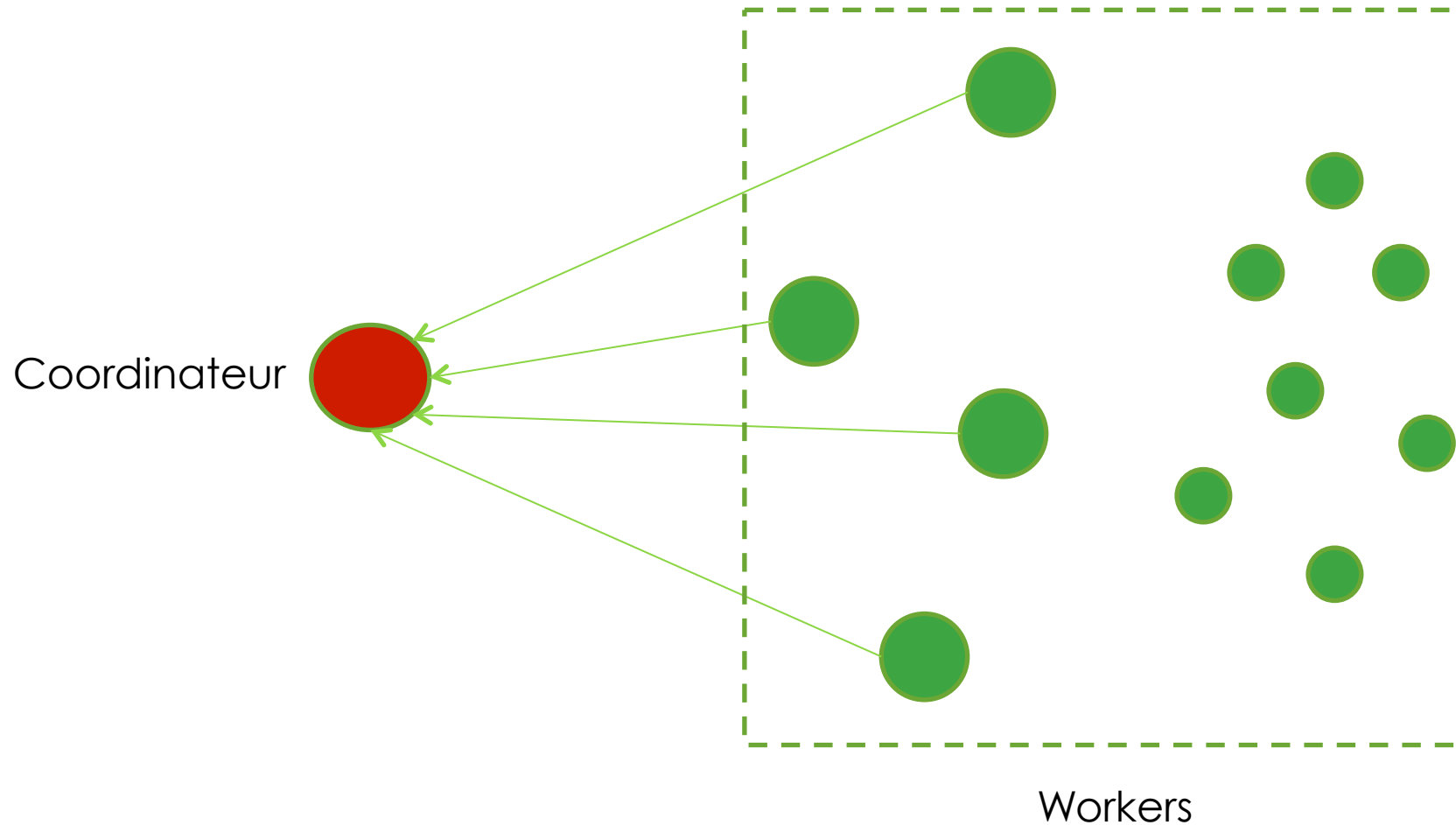
PLAN

- BonjourGrid : version orientée calcul
- BonjourGrid : version orientée données
- Data Management as a Service
- MapReduce sécurisé



**BONJOURGRID : VERSION
ORIENTÉE CALCUL**

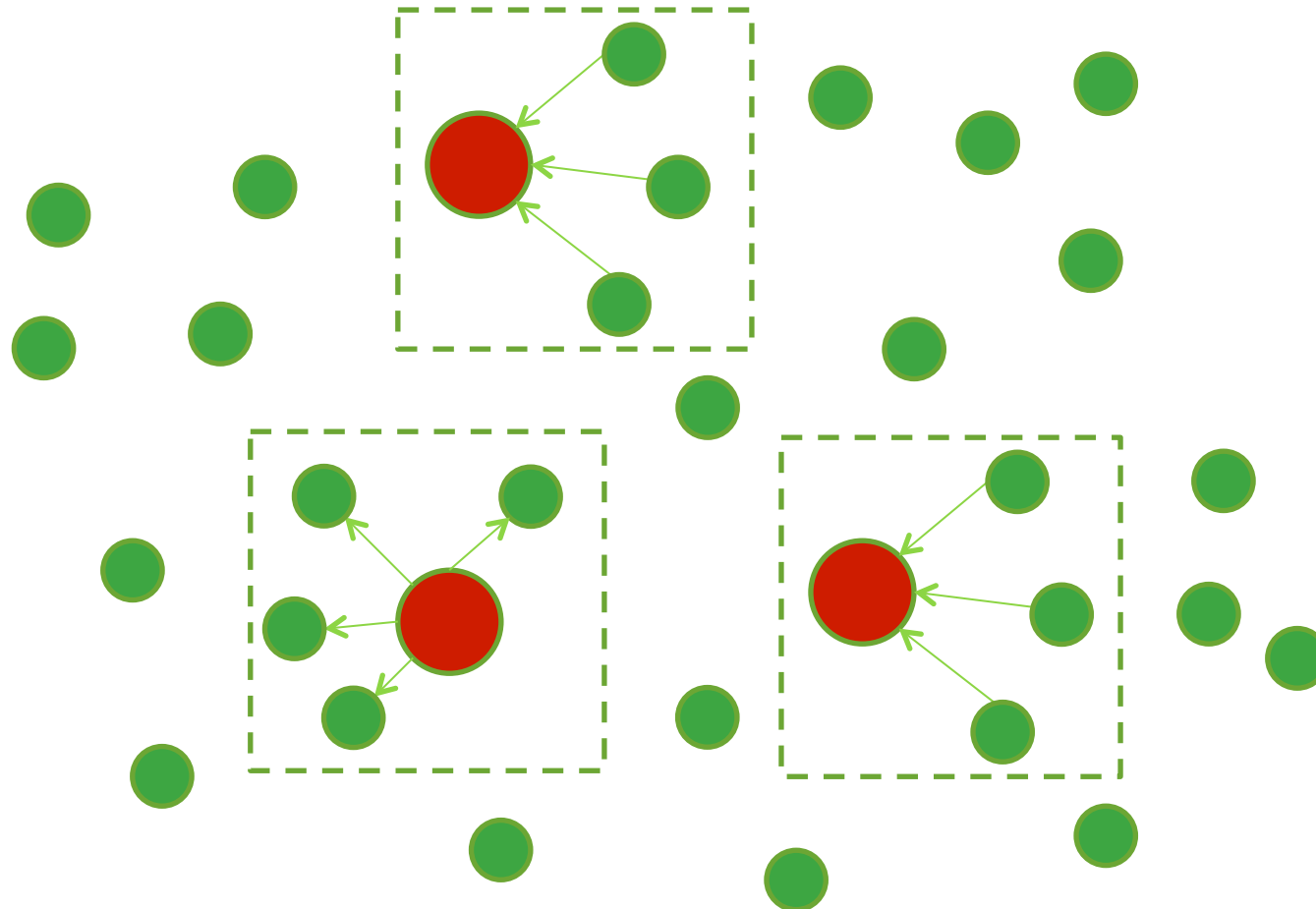
BONJOURGRID : BASIC DESIGN (1/3)



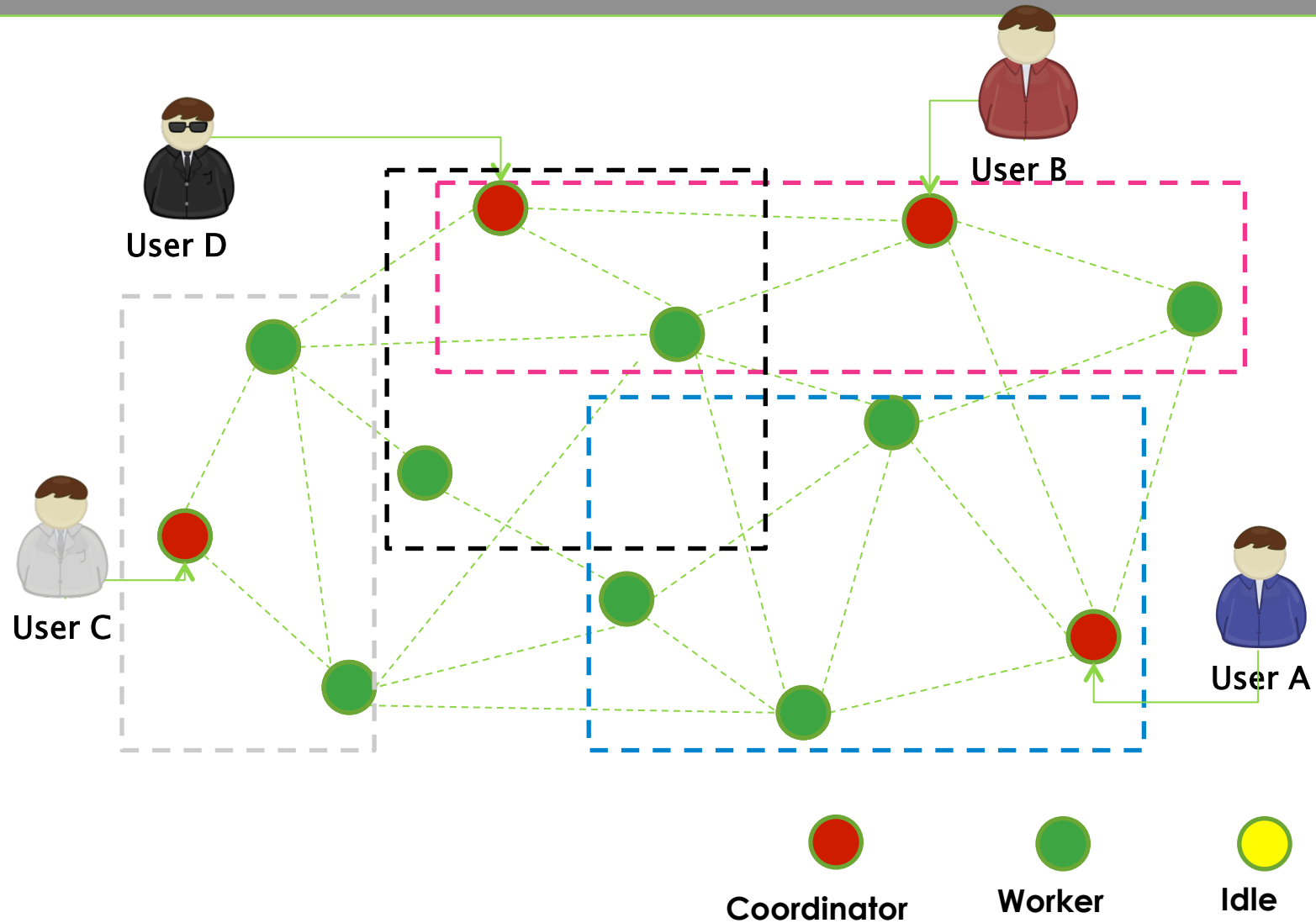
Élément de Calcul (CE) = 1 Cordinateur + N Workers

BONJOURGRID : BASIC DESIGN (2/3)

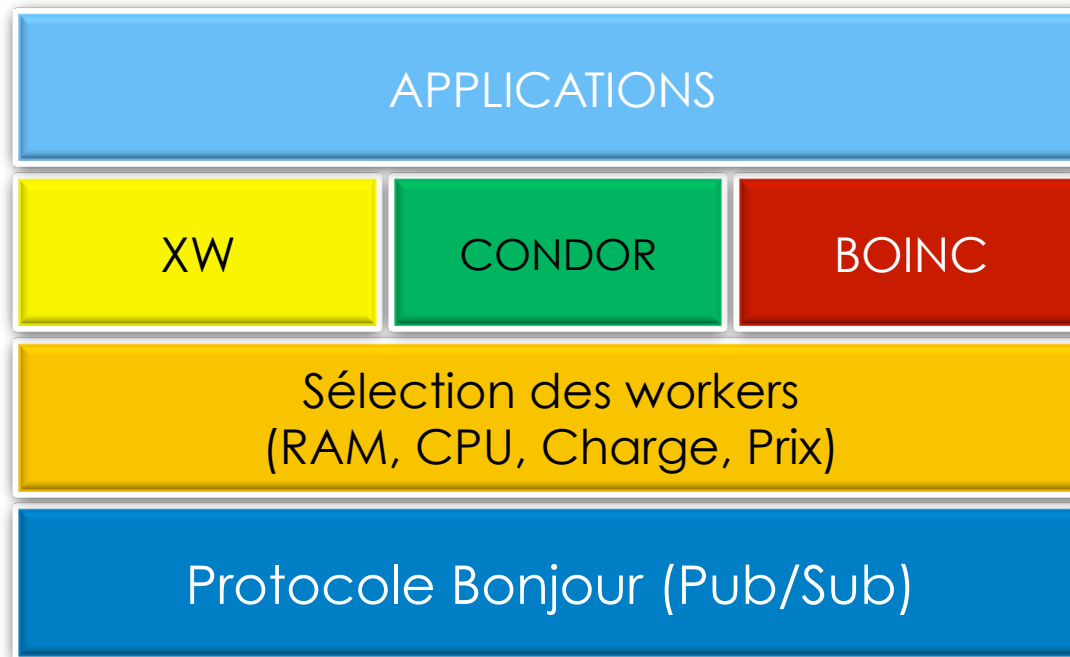
Contrôler et orchestrer de multiple instances via un système de Publish/Subscribe



BONJOURGRID : BASIC DESIGN (3/3)



ARCHITECTURE EN COUCHES

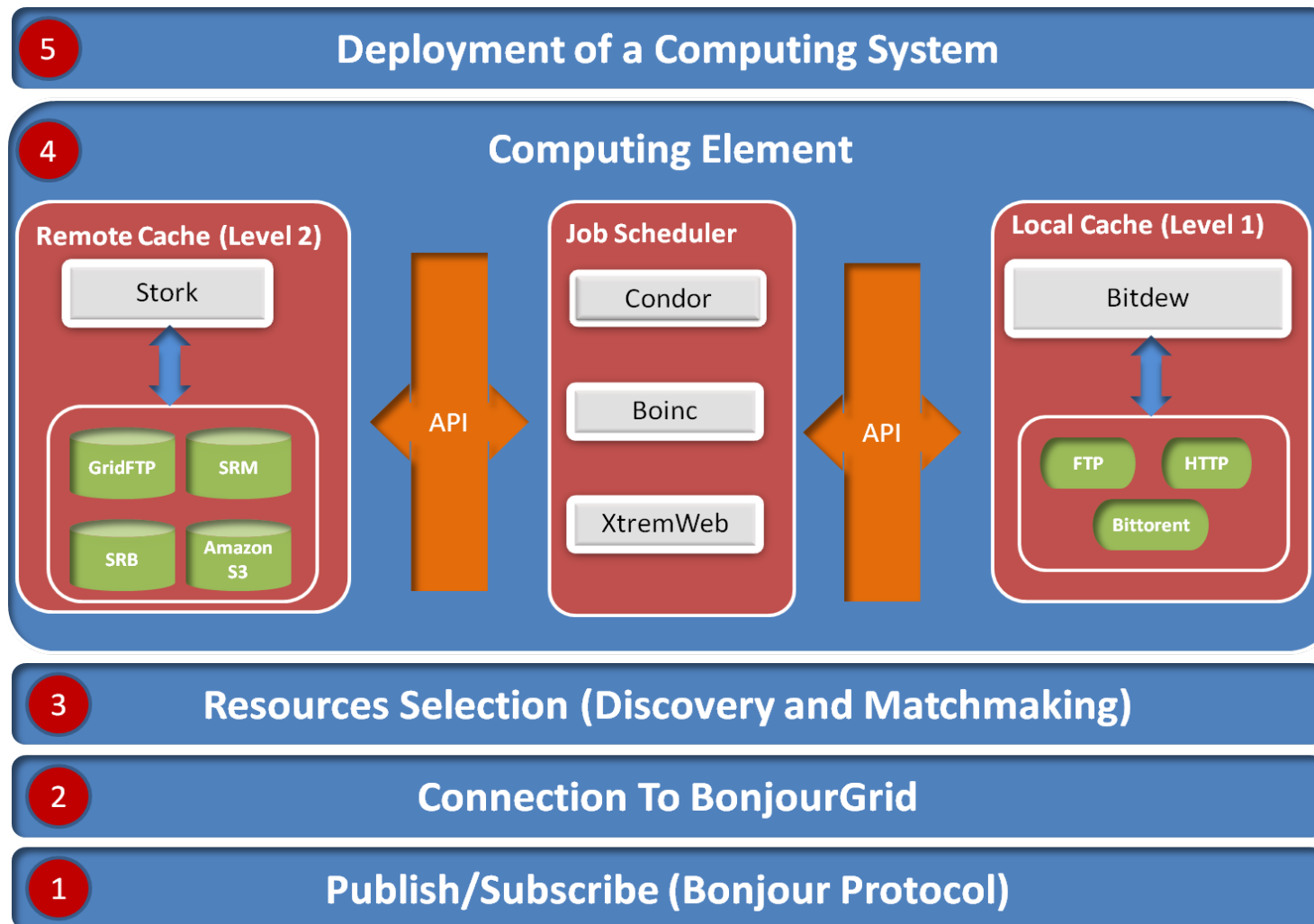


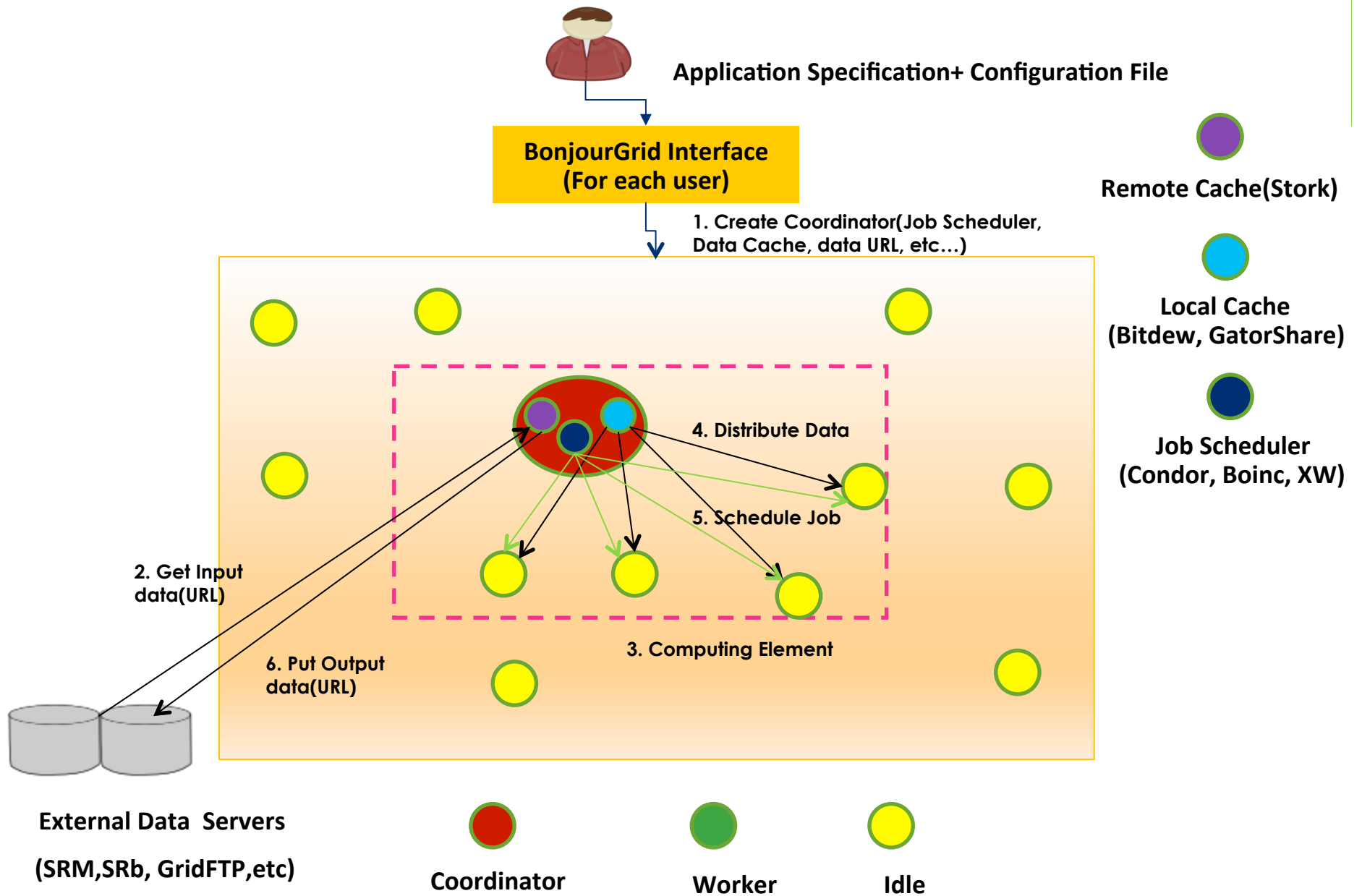
BONJOURGRID : VERSION ORIENTÉE DONNÉES

BONJOURGRID : VERSION DONNÉES

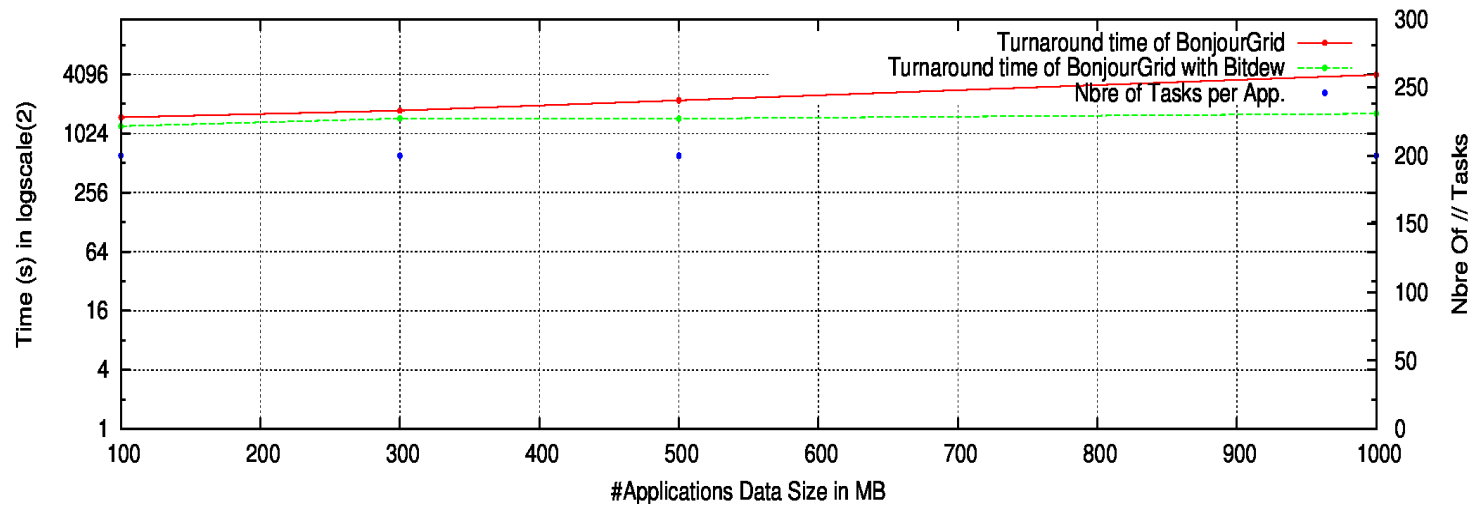
- Un environnement auto-configurable qui supporte différents systèmes de gestion des données
- Extension de BonjourGrid: Une méta-grille pour orchestrer plusieurs instances simultanées de gestionnaires de données et d'intergiciels de calcul

BONJOURGRID : VERSION DONNÉES

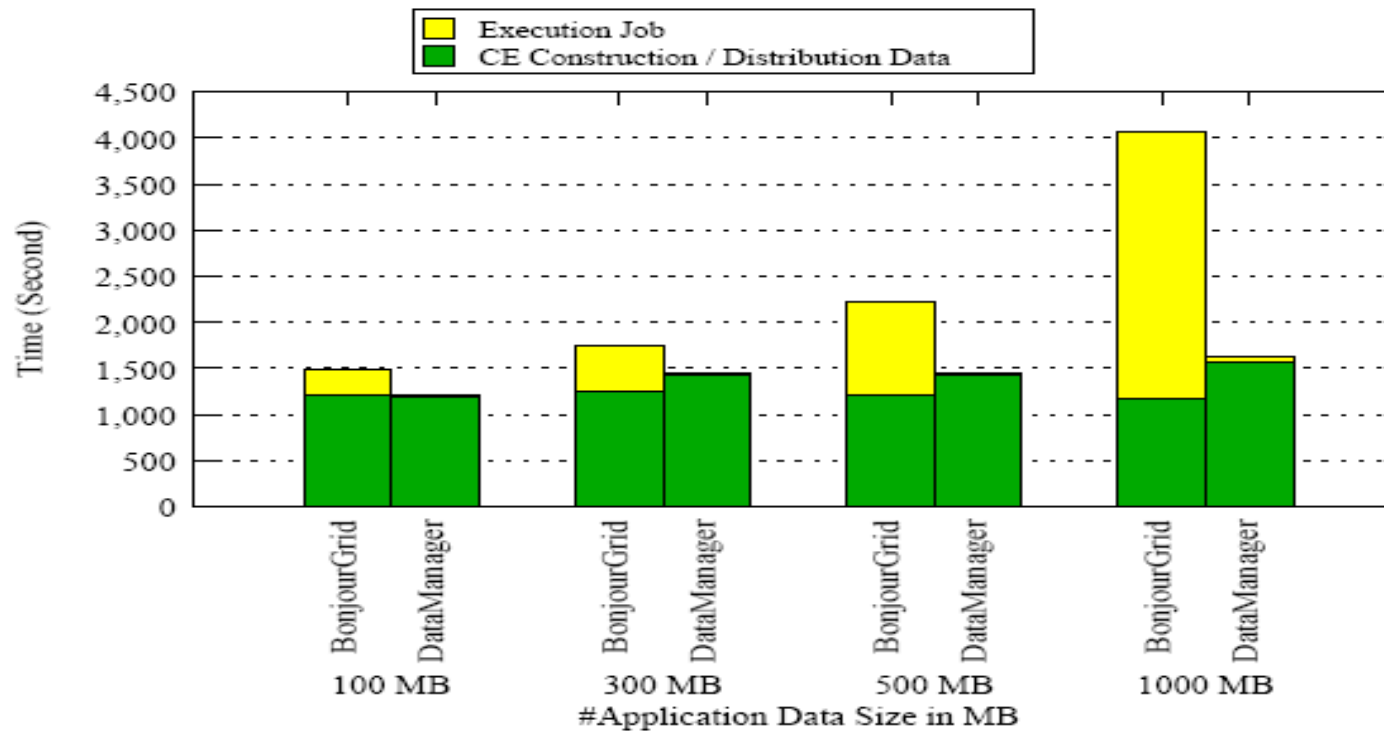




EXPÉRIMENTATIONS



EXPÉRIMENTATIONS



DATA MANAGEMENT AS A SERVICE

DATA MANAGEMENT AS A SERVICE

- **Pourquoi ?**

- Transférer les données depuis le site utilisateur vers le site d'expérimentation
- Transférer les données résultats depuis le site d'expérimentation vers le site utilisateur
- L'utilisateur doit installer et configurer sur sa machine un service client pour transférer les données

- ✓ **Solution**

- ✓ Déployer un service client de gestion de données d'une manière totalement transparente

DATA MANAGEMENT AS A SERVICE

- **Pourquoi ?**

- L'utilisateur a besoin de partager ses données sur l'ensemble de machines disponibles pour effectuer un éventuel traitement
- Pour cela, l'utilisateur doit installer et configurer un environnement comportant:
 - Un serveur de stockage partagé (un serveur NFS ou un serveur GridFTP)
 - Un client sur chaque machine (*client tools*) pour télécharger les données

- ✓ **Solution**

- ✓ Déployer un environnement de gestion de données d'une manière totalement transparente

DATA MANAGEMENT AS A SERVICE

Globus Toolkit

Build the Grid



Components for building custom grid solutions

globustoolkit.org

Globus Online

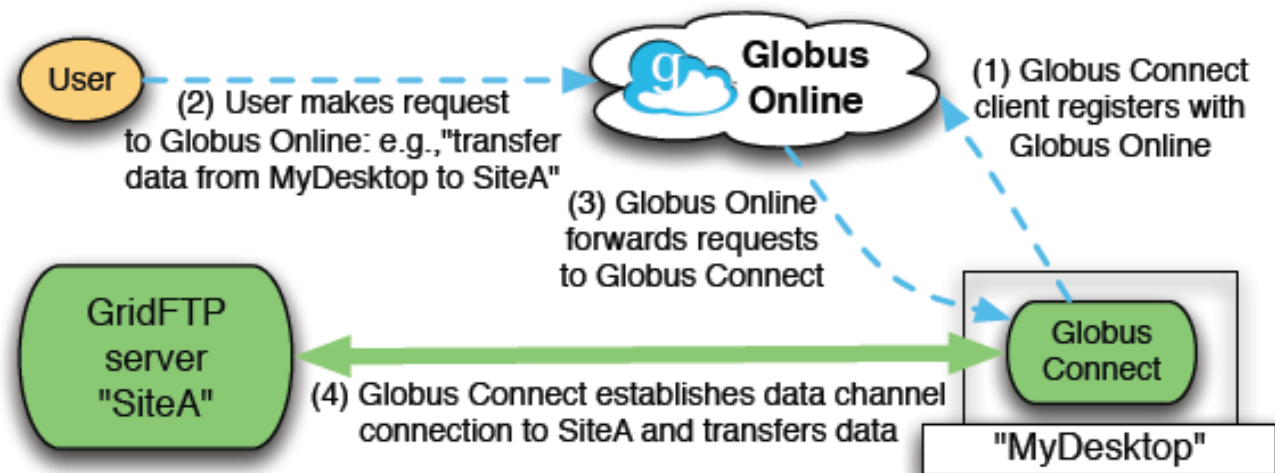
Use the Grid



Reliable file transfer Software-as-a-Service

globusonline.org
www.globusonline.org

- Solution SaaS basée sur des serveurs GridFTP (Laboratoires de recherche, centre de calcul);
- Accès simple via une interface web UI / CLI;
- May 2014: > 3 Millions utilisateurs, >49 petabytes, >150 Millions fichiers transférés



DATA MANAGEMENT AS A SERVICE

- ✓ Proposer un SaaS (*Software As A Service*) à base du Stork et Bitdew pour la création dynamique et à la demande d'un service de gestion de données dans les cloud

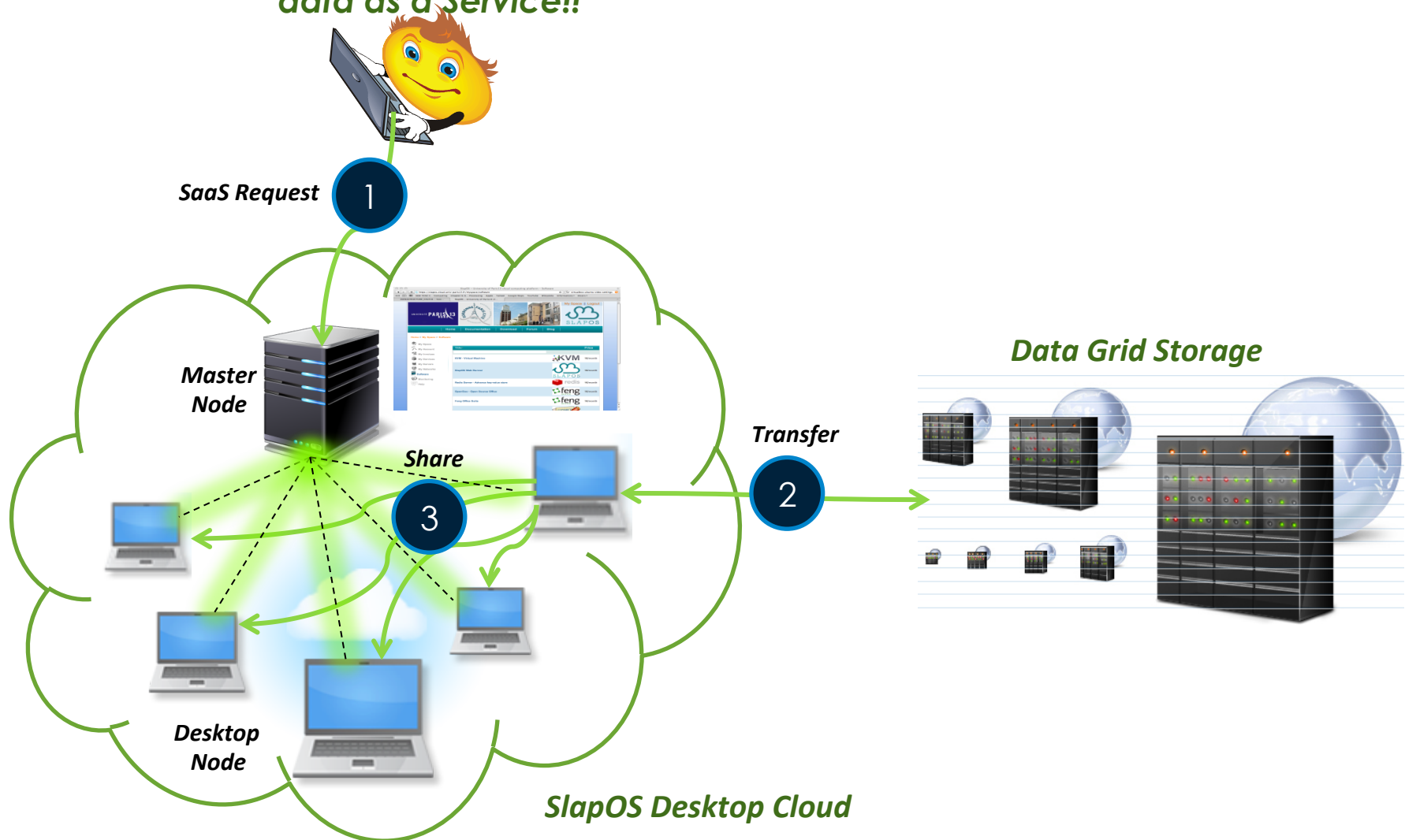
- i. Gestion de données :
 - Authentification, transfert, optimisation et Interopérabilité (Stork)
 - Partage et stockage de données (Bitdew)

- ii. Déploiement de la plateforme

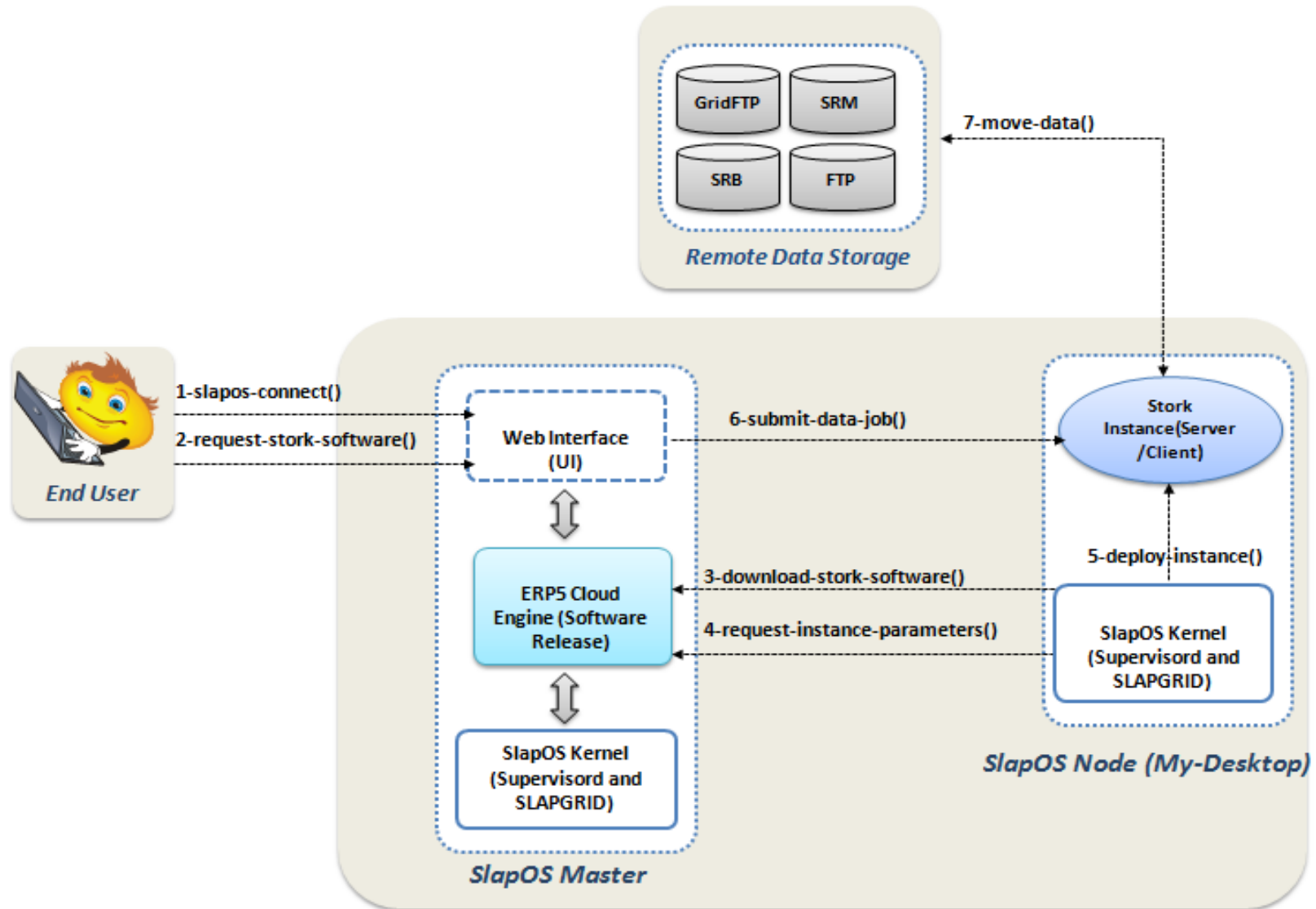
- iii. Accès, sécurité et monitoring

DÉPLOIEMENT DE DMaaS SUR LE CLOUD

End User: I need to manage data as a Service!!



STORK AVEC SLAPOS

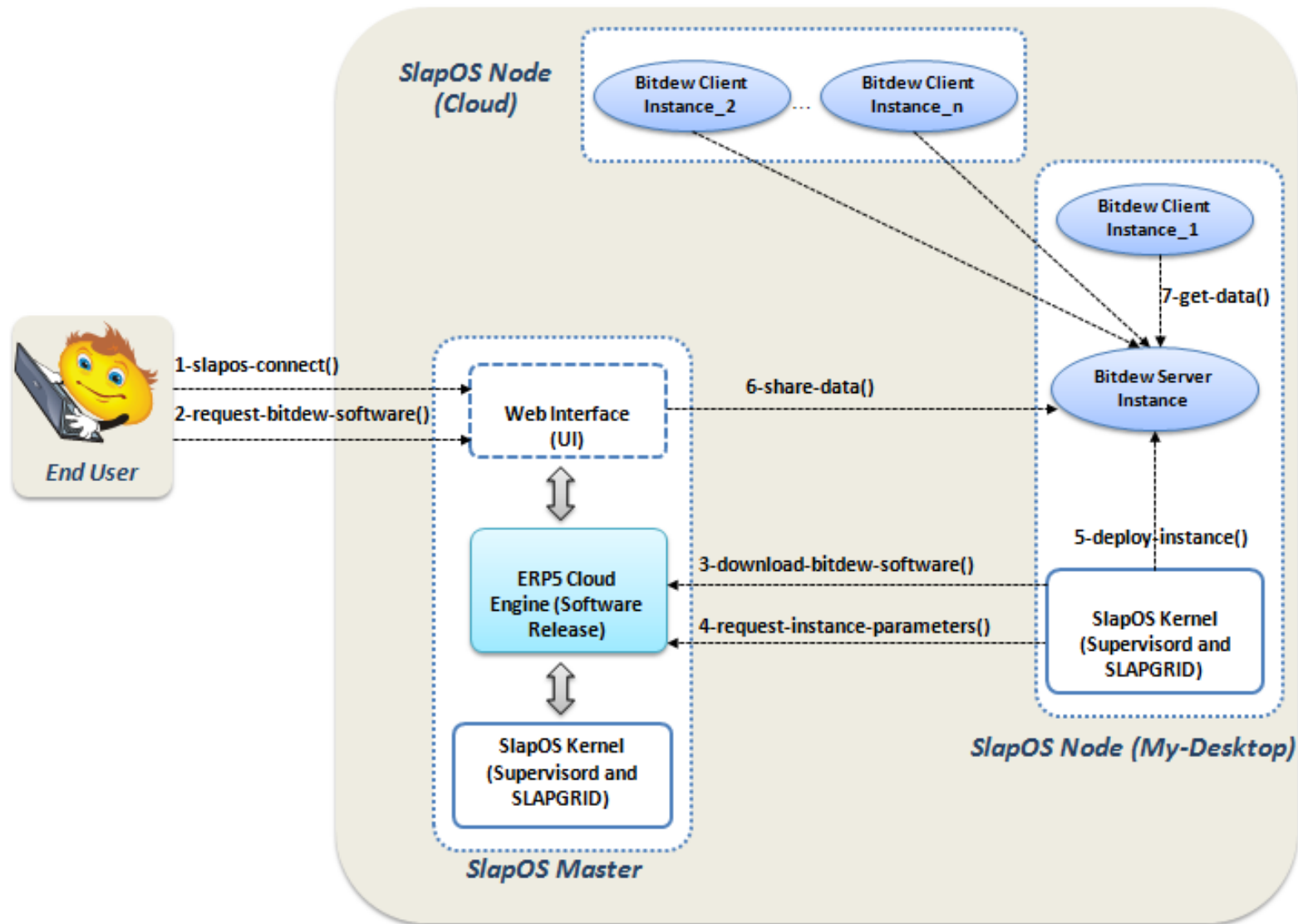


STORK AVEC SLAPOS

Table I
BUILDOUT MAIN FILES FOR DEPLOYING STORK INSTANCES WITH SLAPOS

SlapOS internals	Buildout and Recipes Files	Description
Component	<i>Buidlout.cfg</i>	Install the the list of dependencies required to compile and run Stork.
Software Release (SR)	<i>Software.cfg</i> <i>Instance – Stork.cfg</i>	Build all binaries and download template file to submit Stork jobs. Deploy Stork instances based on user parameters (Stork_Server, data URLs, protocols, etc).
	<i>Template/submit.dap</i>	The user can submit job to Stork_Server using this default template dap file as follows: <pre>[protocol = "USER PROTOCOL"; src_url="USER SOURCE URL"; dest_url="USER DESTINATION URL"; err = "log.err"; output = "log.out"; dap_type = "transfer"; x509prox="USER PROXY PATH";]</pre>
Software Instance	<i>slapos.cookbook : stork</i> <i>slapos.cookbook : stork.submit</i>	Called from Stork-Instance.cfg file, reuse installed SR by creating wrappers (Start_Stork, Start_Submit), configuration files and anything specific to Stork instance (Client or Server).

BITDEW AVEC SLAPOS



CLOUD DE PARIS 13

<https://slapos.cloud.univ-paris13.fr/>

Reference

COMP-3



- ✓ Update allocation scope
- ✓ Update categories
- ✓ Request SSL certificate
- ✓ Revoke SSL certificate

Image	Title	Version	State	Usage
	Slapos Web Runner	dev-1.1	Installation requested	1
	Stork - batch scheduler for data placement and data movement	branch	Installation requested	0
	BitDew - Open Source Data Management for Grid, Desktop Grid and Cloud Computing	branch	Installation requested	0

EXPÉRIMENTATIONS

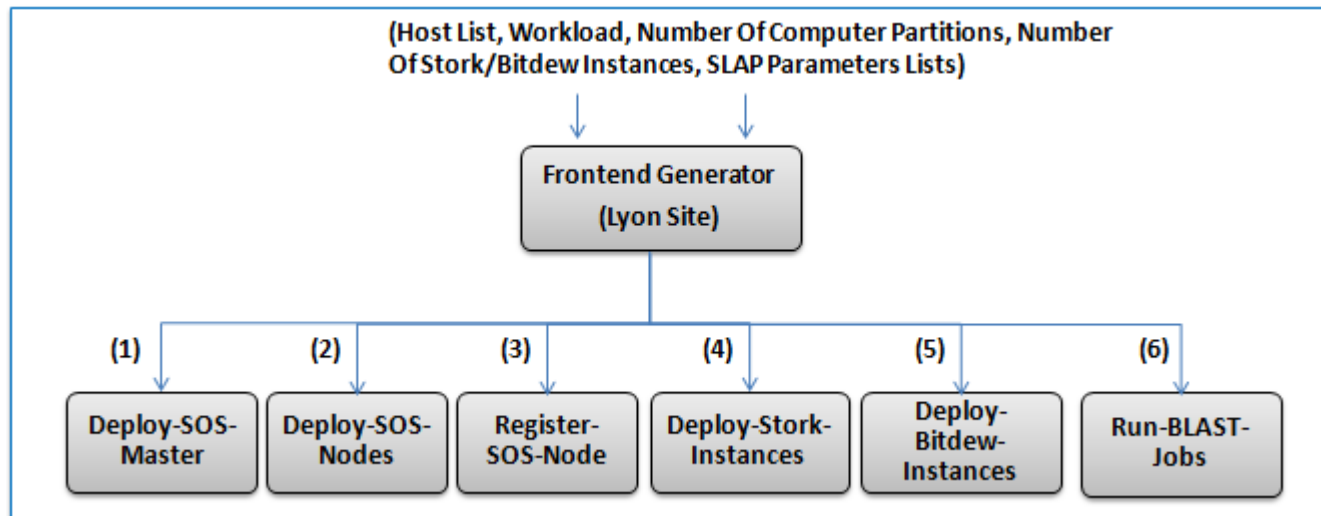
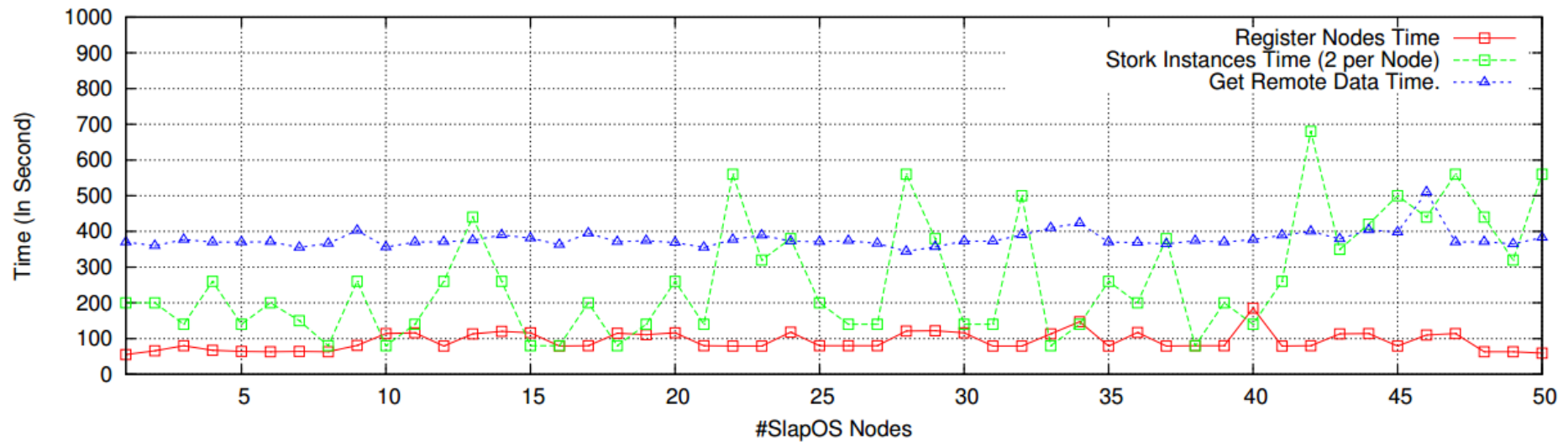


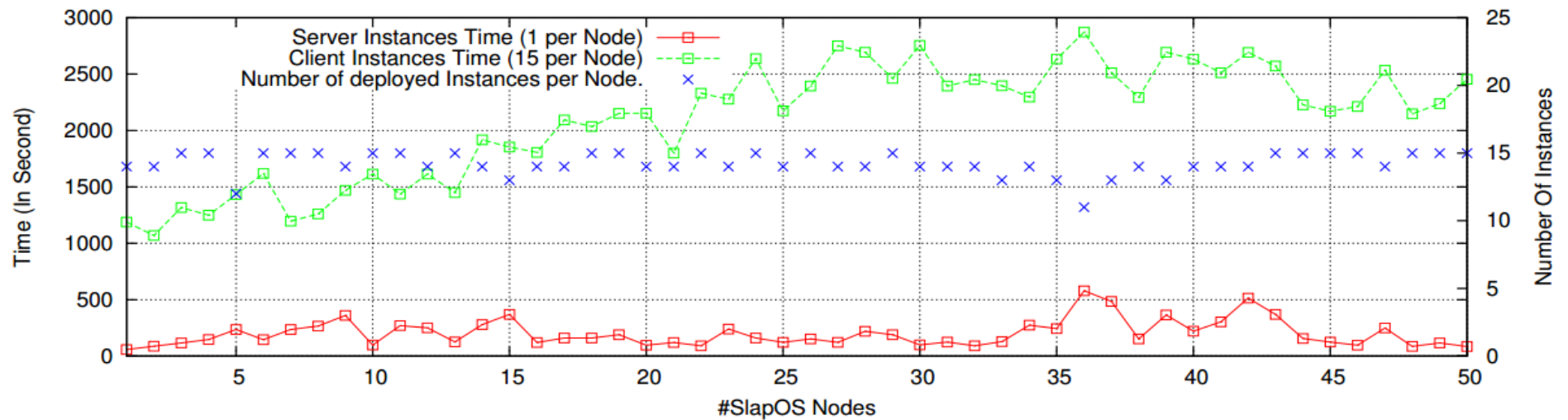
Table 5 Workload used for the experimentation.

Workload Information	BLAST Applications
Number of SlapOS Nodes : <i>50 machines of Lyon Site Debian Wheezy 86_64 3.2.0-4-amd64 (clusters: sagittaire, orion, hercule, taurus)</i> Total number of SaaS instances per Node : 18 Total number of instances : $50 * 18 = 900$	Human GenBase: <i>est-human.01.tar.gz (199 MB)</i> Queries Sequences: <i>queries.tar.gz (150MB)</i> BLAST Program: <i>ncbi-blast-2.2.29 (175 MB)</i> Total size of Data : <i>~550 MB</i>

EXPÉRIMENTATIONS



EXPÉRIMENTATIONS



MAPREDUCE SÉCURISÉ

MAPREDUCE SÉCURISÉ

- Assurer la **sécurité de la distribution des données** sur les clouds en vue de les traiter dans des applications **Map-Reduce**
- Des alternatives proposent:
 - des techniques de cryptographie (Infrastructures PKI)
 - de contrôle d'accès (MAC: Mandatory Access Control) ou de contrôle de résultats (DP: Differential Privacy) (Airavat)
 - **Solutions très coûteuses**: Cryptage/Décryptage/Gestion de droits..
- **Notre proposition**: utiliser l'**algorithme de dispersion IDA** pour sécuriser les données à traiter par les mappers.

IDA (1/3)

1. Principe

- Étant donné un fichier de données, IDA en génère n morceaux dont m sont seulement suffisant pour reconstruire le fichier. ($m < n$)
- Exemple: 10 morceaux de F, 8 sont suffisants pour restituer F.

2. Objectifs

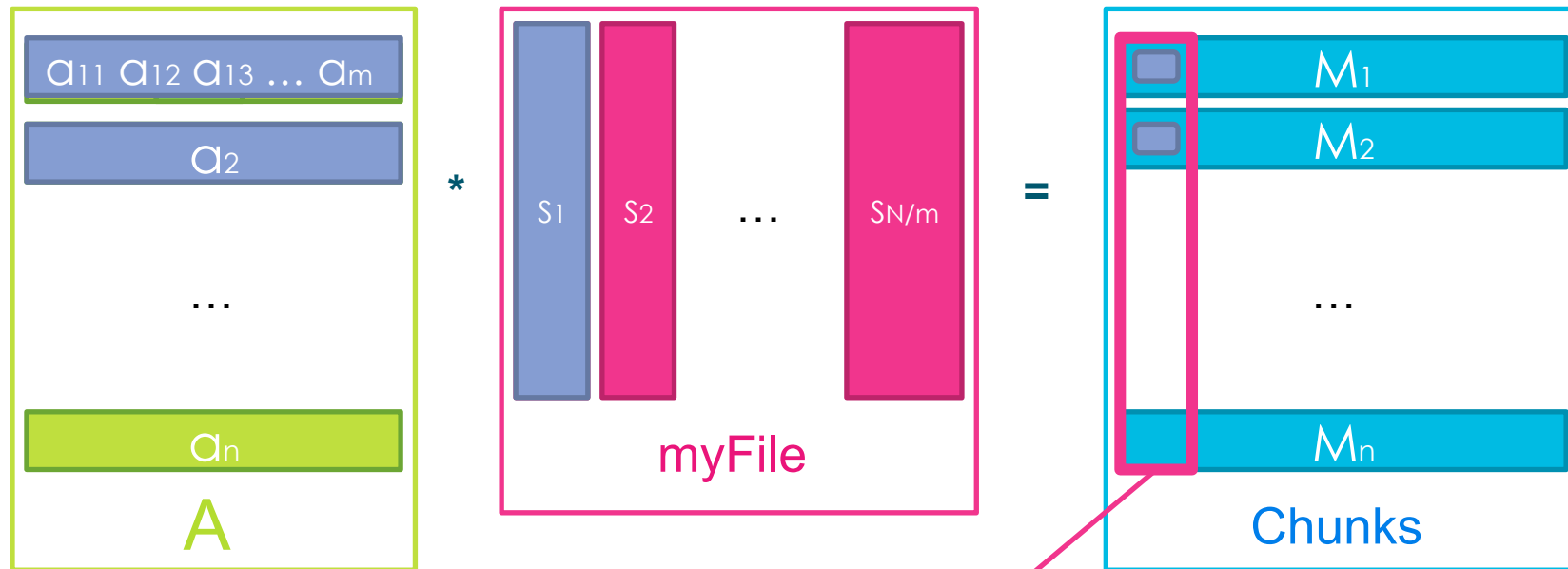
- Remédier aux problèmes liés à la transmission et au stockage.

3. Fonctionnement

- Split
- Combine

IDA (2/3)

Phase de Split



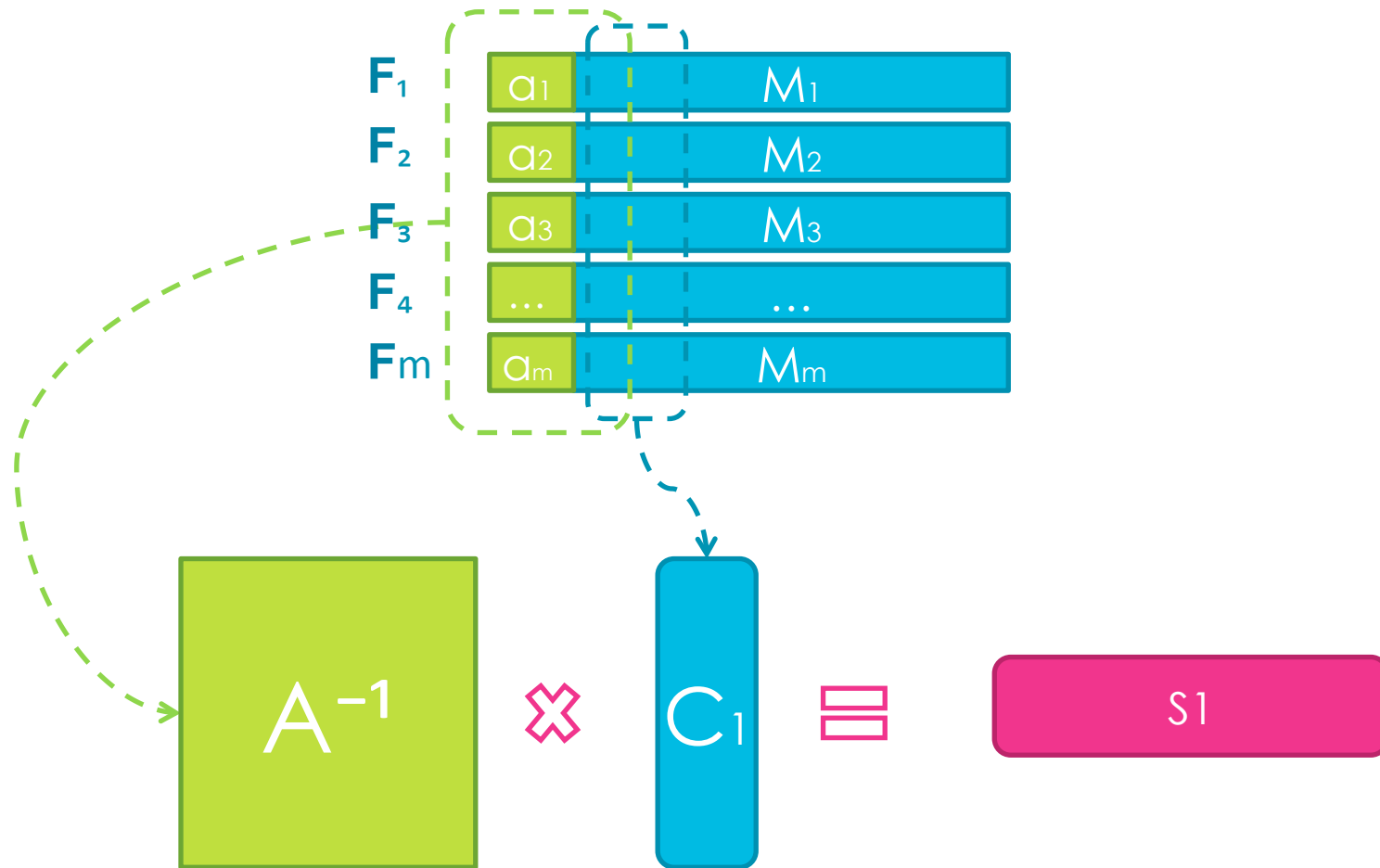
$$C_1 = A * S_1$$

→ $S_k = A^{-1} * C_k$

À envoyer **Fi** :

IDA (3/3)

Phase de Combine

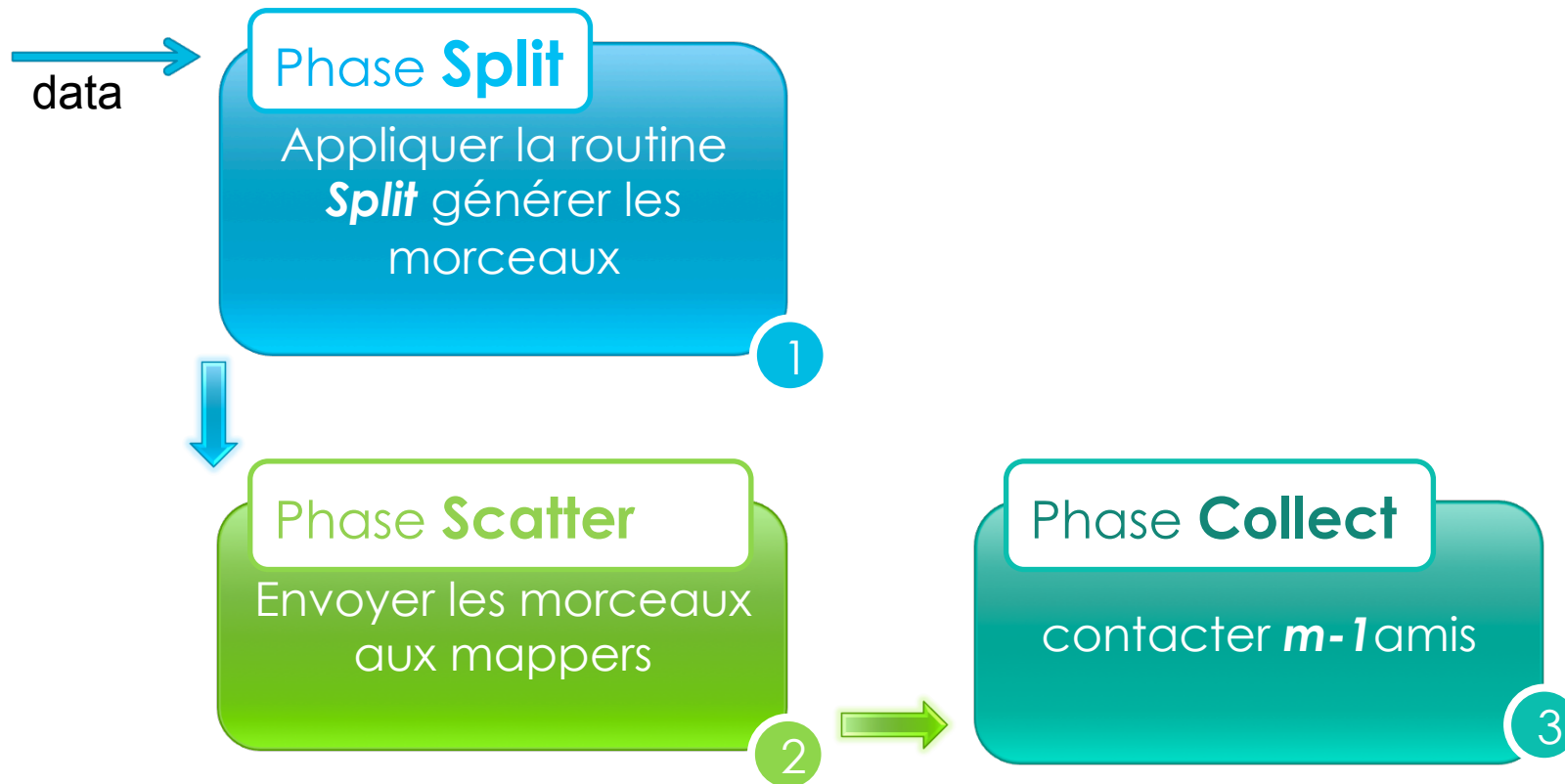


NOTRE APPROCHE (1/5)



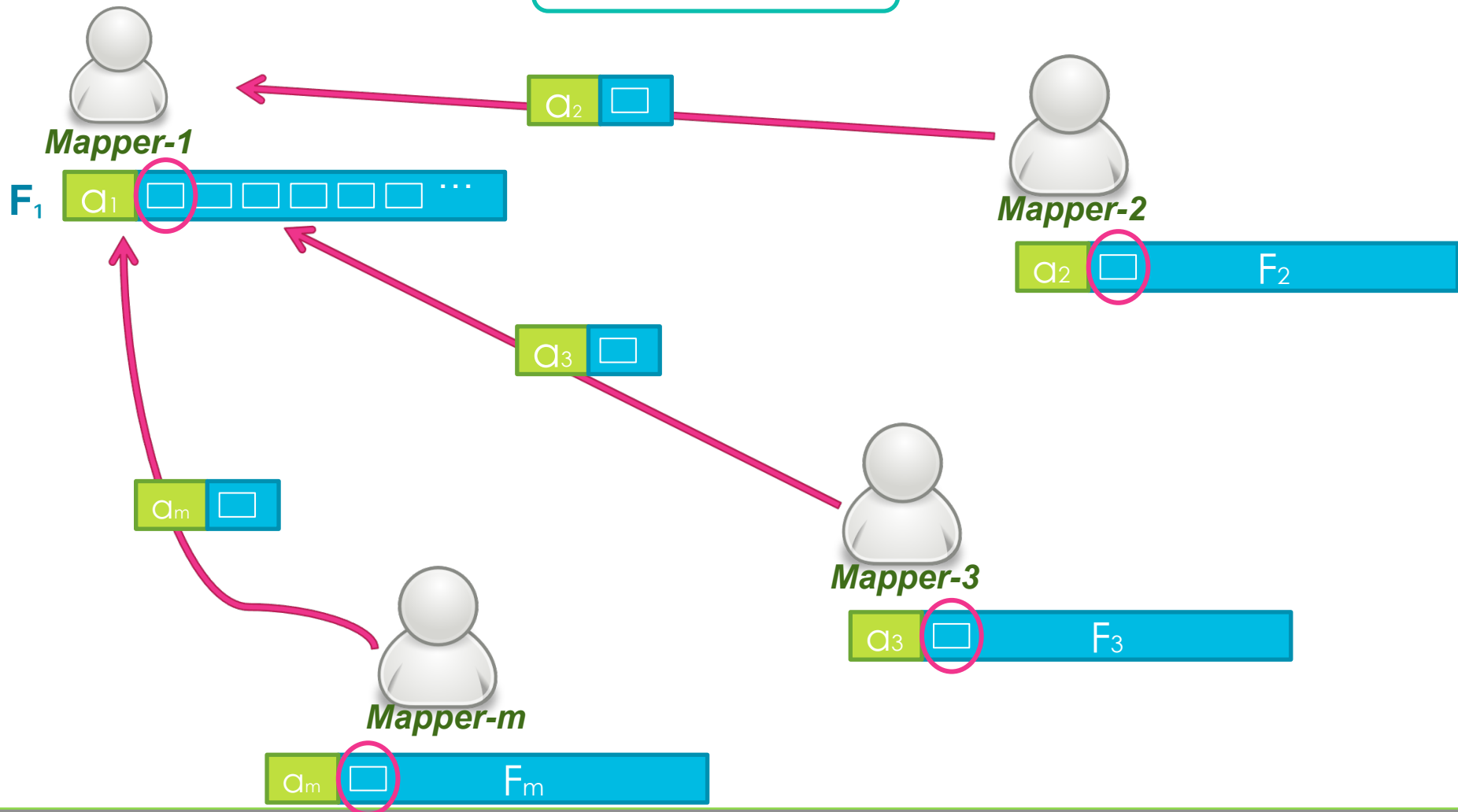
- Distribuer les données sur les mappers
 - Chaque mapper reçoit une partie des données **non valables**
 - Un mapper **doit contacter** d'autres mappers (amis) pour pouvoir reconstruire des données valides
 - Si un mapper ne peut avoir accès à une des données, il **échoue** dans la reconstruction des données valides
-
- **Comment appliquer IDA dans MapReduce pour bénéficier de l'aspect de dispersion de données ?**
 - **Comment peut-on pouvoir se bénéficier des caractéristiques du cloud hybride ?**

NOTRE APPROCHE (2/5)

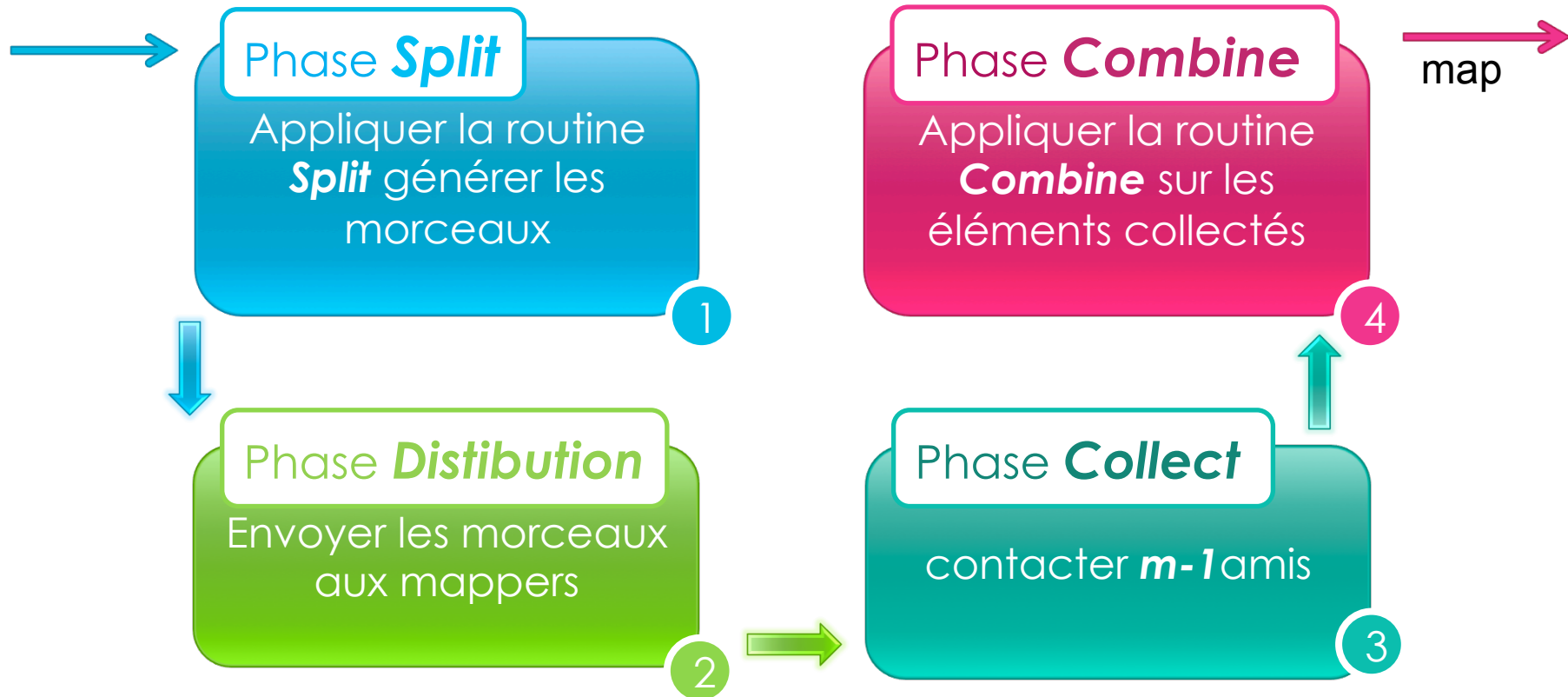


NOTRE APPROCHE (3/5)

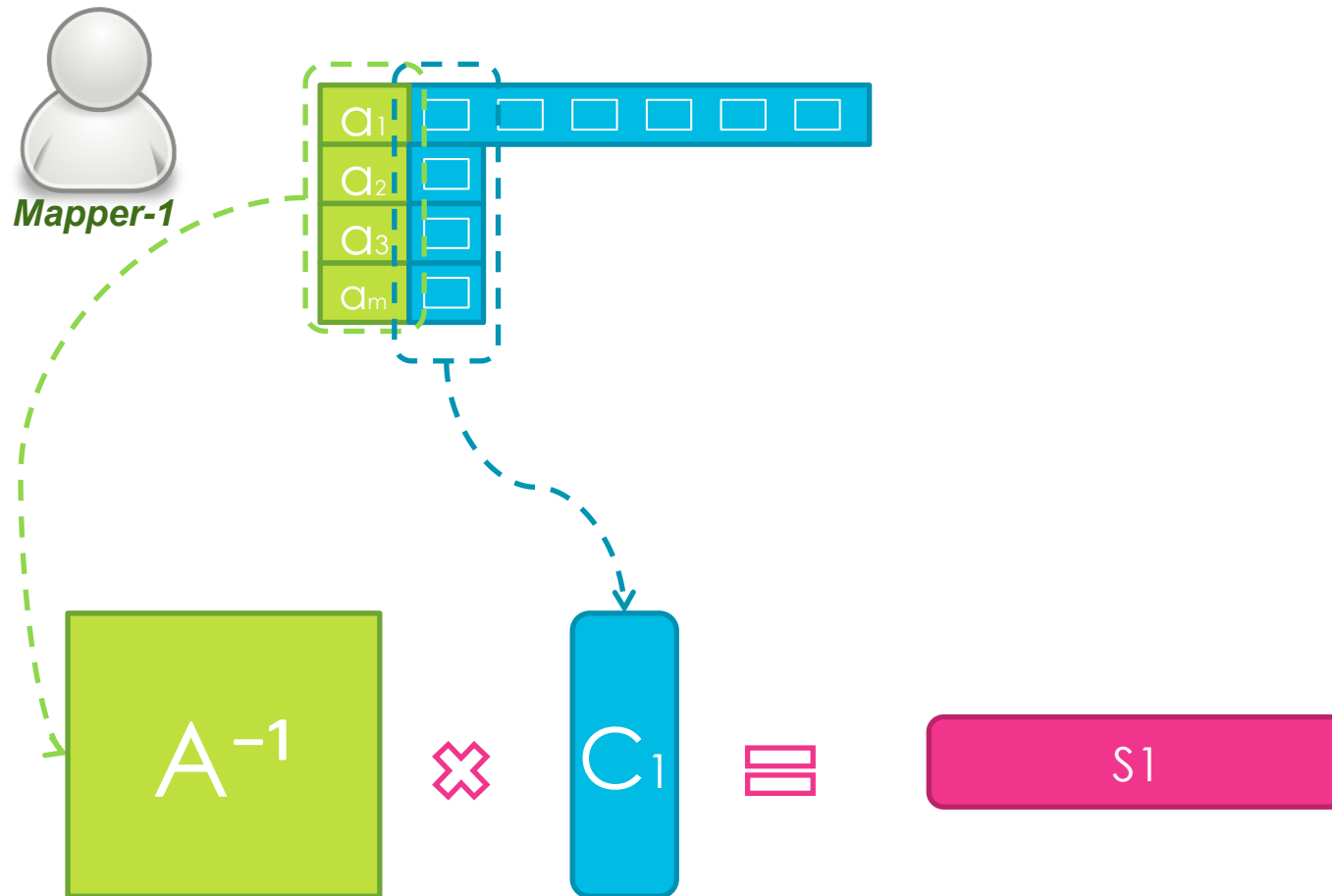
Phase **Collect**



NOTRE APPROCHE (4/5)



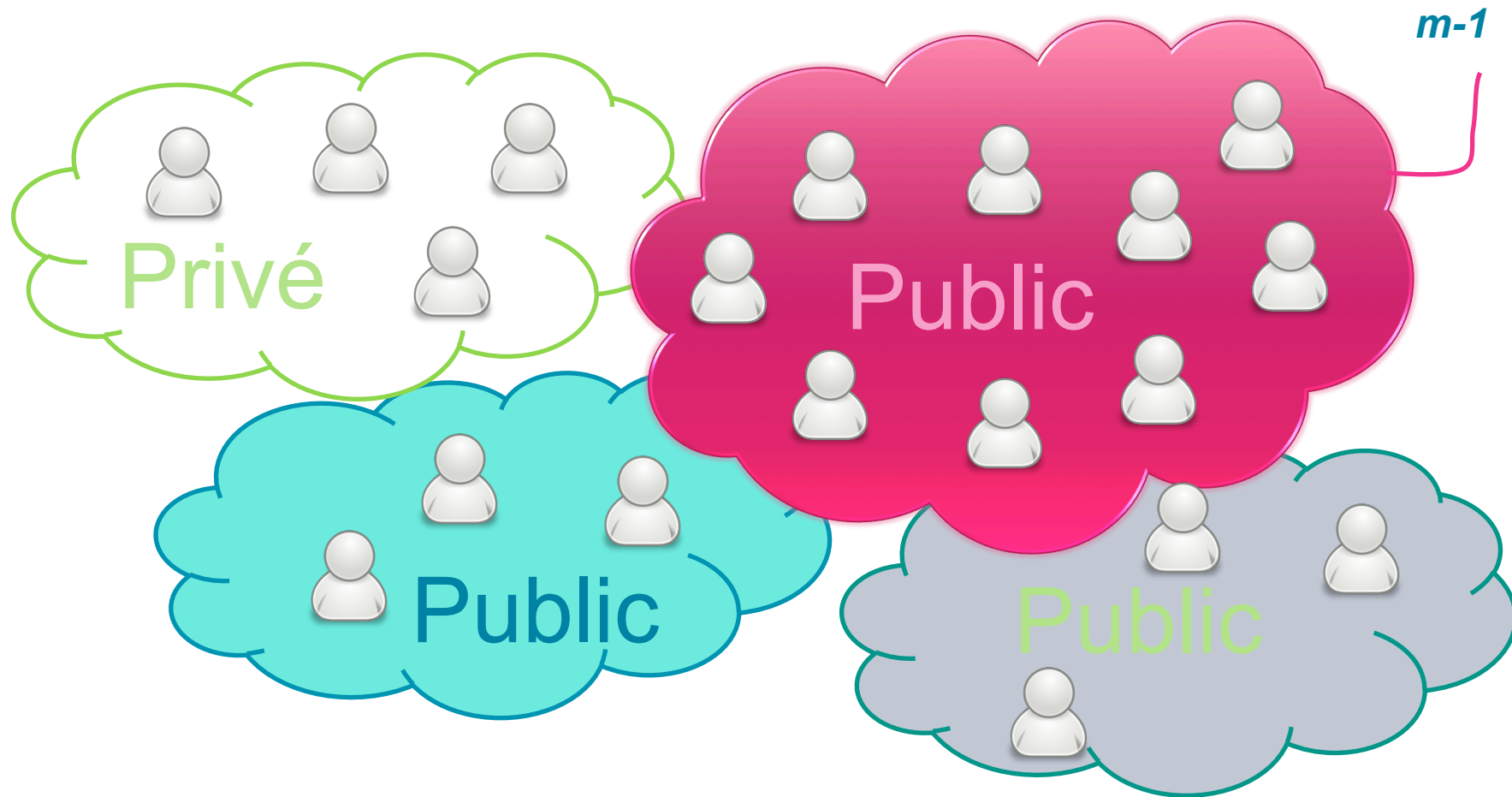
NOTRE APPROCHE (5/5)



DÉPLOIEMENT

- Un mapper malicieux a accès à ses données.
 - 1 morceau est en clair (déchiffré)
- Un 2^{ème} mapper malicieux peut exposer ses données.
 - 2 morceaux déchiffrés
-
- Et si tous les mappers sont malicieux ?
 - ***m*** morceaux peuvent restituer la totalité des données
- **Solution:** Utiliser le cloud privé pour cacher un nombre **nécessaire** de morceaux.

DÉPLOIEMENT



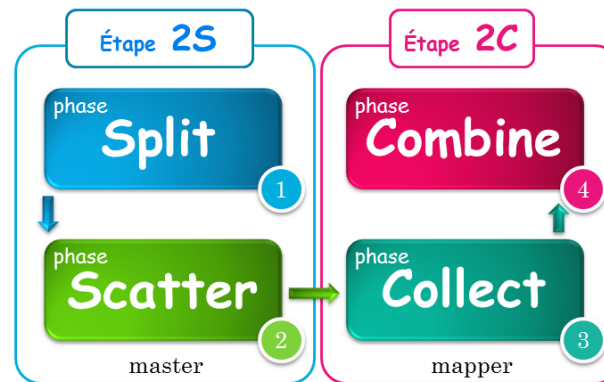
RÉALISATION

- **CRYPT-IDA** : bibliothèque des routines de IDA en Perl
- La séquence de IDA restituée est de taille très petite
- Nous avons **modifié CRYPT-IDA** pour manipuler une entité de données plus considérable: **paquet**
 - **Un paquet = un ensemble de séquences**
- Durant la phase Collect, un mapper demande plusieurs éléments
- La phase combine recompose un paquet

EXPÉRIMENTATIONS



- Plateforme: Grid'5000
- 180 Machines : Nancy(Griffon:8, Graphene:4), Lyon(Sagittaire:12)
- Cycle de vie:



- Paramètres:
 - ***data_size***, Taille des données (variant de 100 Mo à 1,3 Go)
 - ***n***, nombre de morceaux
 - ***m***, nombre de morceaux nécessaires pour restituer une information valide
 - ***pack_size***, taille de packet

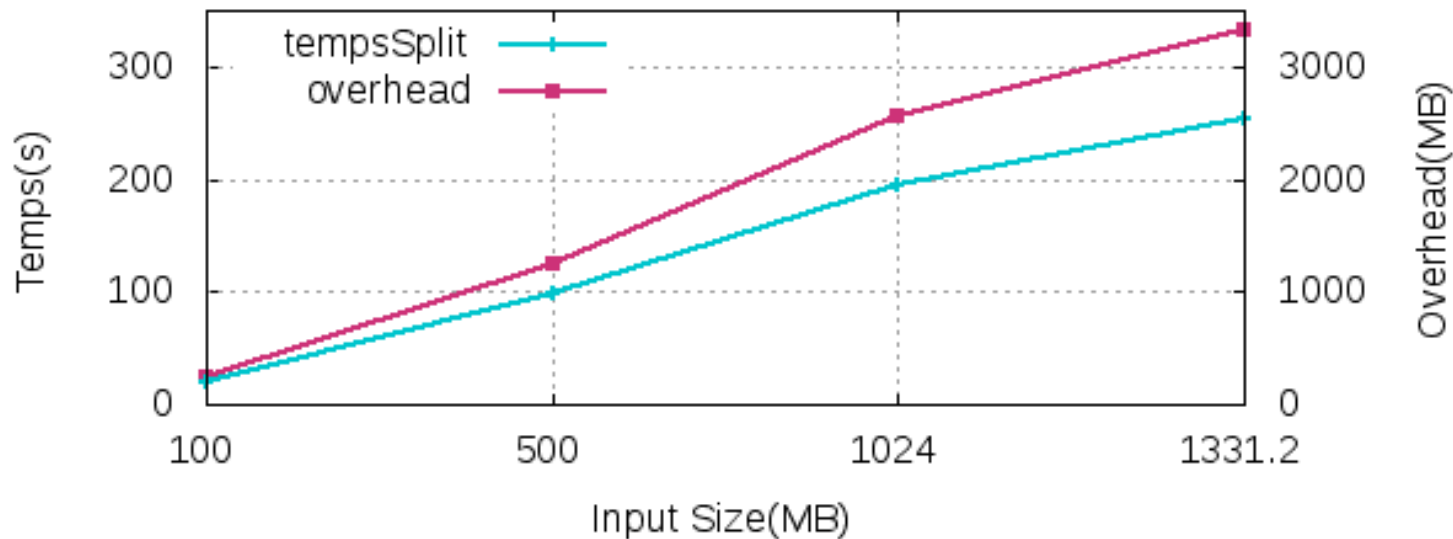
ÉVALUATION: SPLIT & SCATTER (2S)

Objectif

- Impact de *data_size* sur :
 1. la durée de la phase Split
 2. La quantité de données

Paramètres

- Nœud: Griffon(8 cœurs)
- *Data-size*: de 100Mo à 1,3Go
- $n=25$, $m=10$ **Overhead=150%**



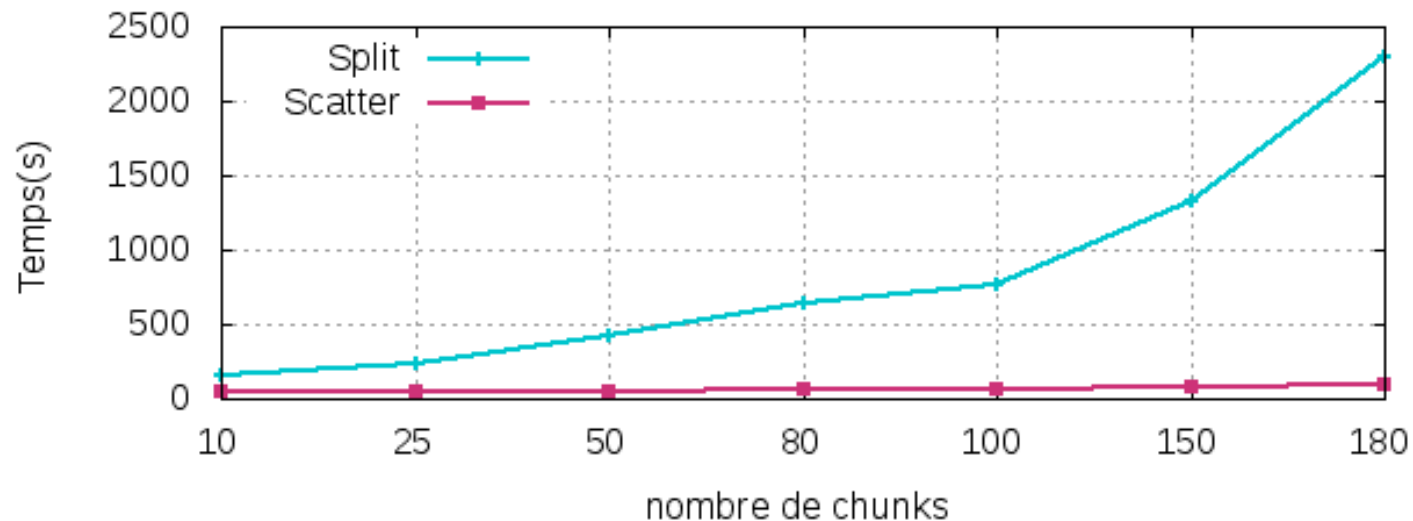
ÉVALUATION: SPLIT & SCATTER (2S)

Objectif

- Impact de n sur :
 1. la durée de la phase Split
 2. La durée de la phase Scatter

Paramètres

- Nœud: Griffon(8 cœurs)
- ***data-size*** = 1,3GB



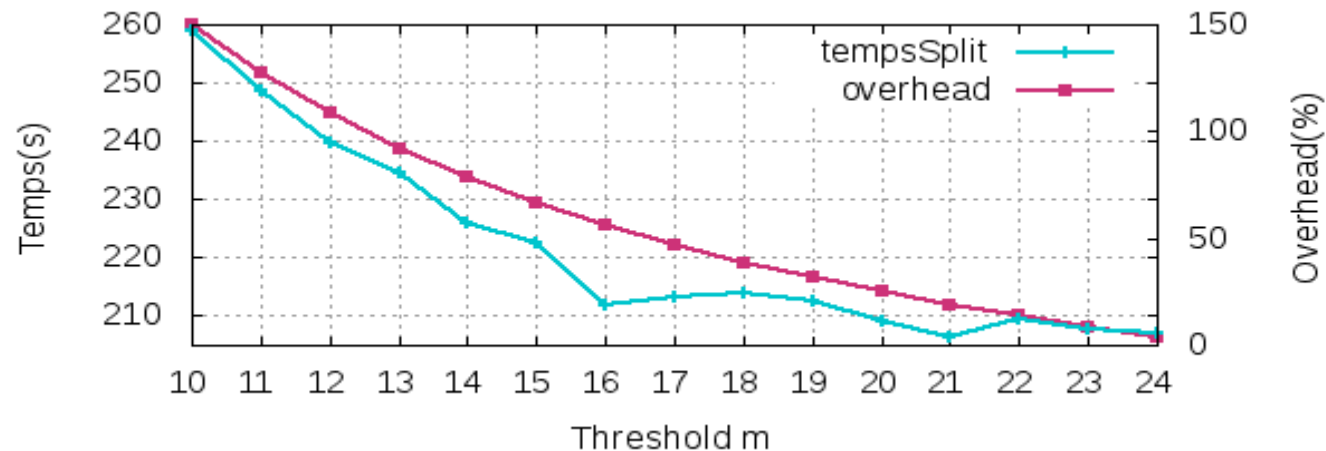
ÉVALUATION: SPLIT & SCATTER (2S)

Objectif

- Impact de m sur :
 1. la durée de la phase Split
 2. La quantité des données

Paramètres

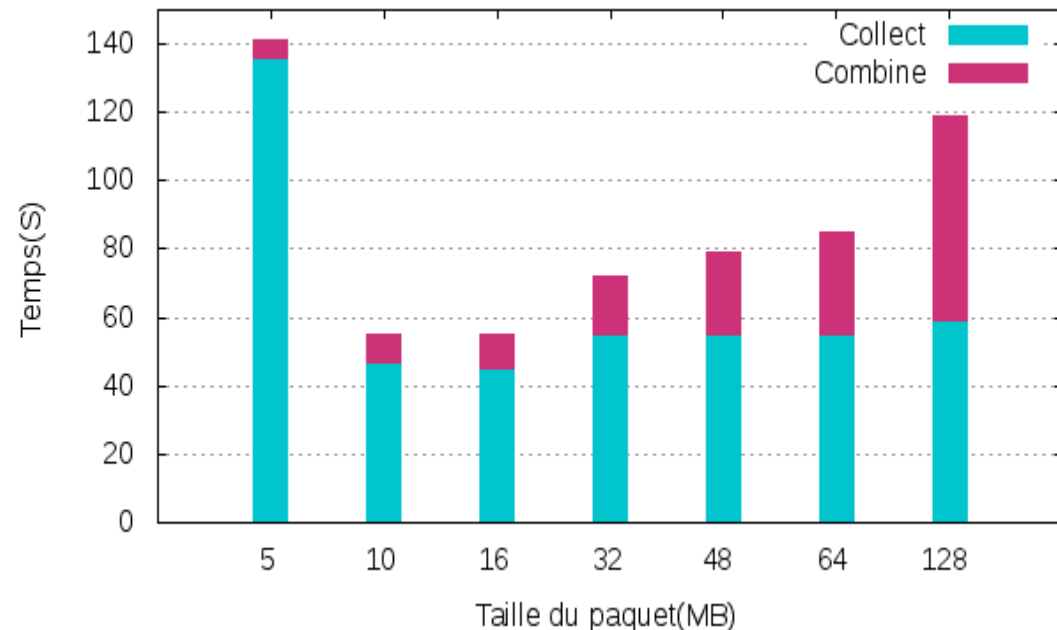
- Nœud: Griffon(8 cœurs)
- ***data-size*** = 1,3GB
- ***n*** = 25



ÉVALUATION : COMBINE & COLLECT (2C)

- Un mapper :
 1. m amis
 2. paquet de taille pack_size
- **Data-size = 1,3GB**
- $n = 180, m = 68$

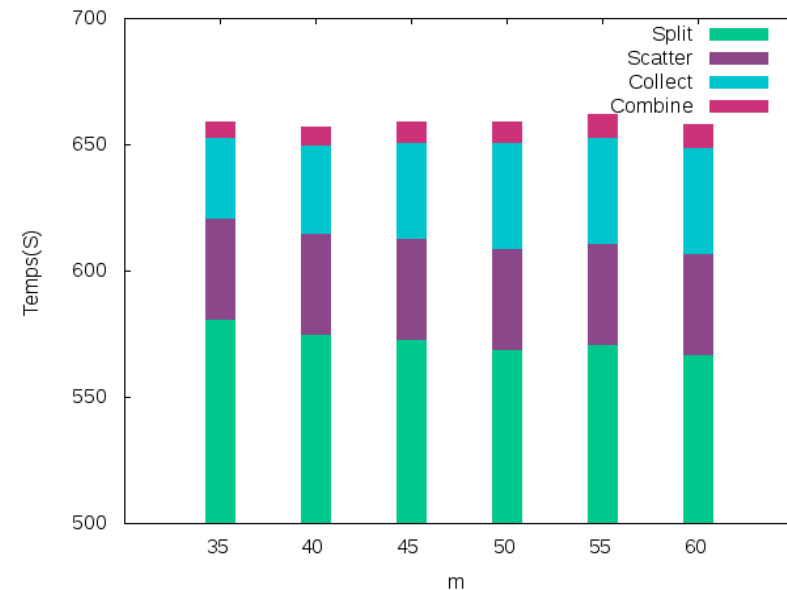
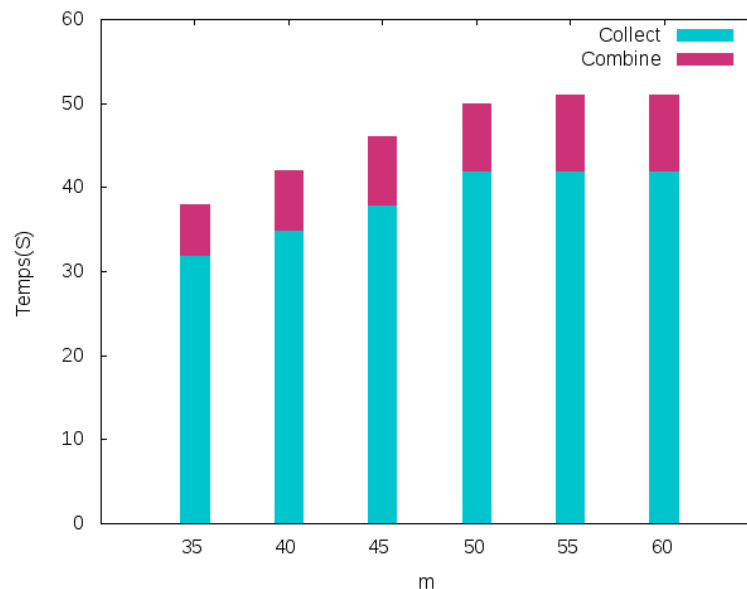
La durée de l'étape 2C en fonction de la taille du paquet



- Optimum **pack_size=16MB**.

ÉVALUATION : COMBINE & COLLECT (2C)

- Impact de m sur les 2 phases Collect et Combine.
- **data-size = 1,3GB**
- **pack_size = 16MB**
- **$n = 82$**



- Optimum durant l'étape 2C, $m=35$: le minimum est l'optimum.
- Durant un cycle complet, l'optimum est $m=40=n/2$.

**MERCI DE VOTRE
ATTENTION**