Analyse distribuée

Une plongée dans l'analyse du CPU et réseau du cluster du Lapp (analyse site-dépendante)

11/06/09

Gaël Rospabé

Contexte :

But : - Comparer CPU / réseau
- Pour : - différentes localisations des données
- différentes façons de travailler (série / parallèle)
- jobs sur WN ou UI ?

ROOT : PyRoot (interface Python)
- chainage des fichiers à analyser
(fichiers concaténés lors de leur ouverture)
- lecture de l'ensemble des fichiers
- merger les fichiers pour n'en avoir que un seul

⇨ CPU mesuré = CPU utilisé pour remplir des histogrammes / données

Plan :

1. Accès depuis une UI de fichiers sur disque local / SE
   - protocole root – open(rfio:/dpm/…)
   - analyse fichiers en série / analyse d'un fichier mergé
2. Accès depuis un WN de fichiers sur SE
   - protocole root – open(rfio:/dpm/…)
   - analyse fichiers en série / analyse d'un fichier mergé
3. Accès depuis une UI de fichiers sur disque local / SE
   - copie locale via commande rfcp
   - analyse fichiers en série

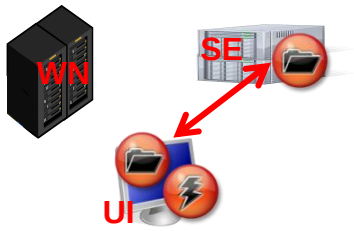
   } parallélisation copie /analyse
4. Accès depuis un WN de fichiers sur SE
   - copie locale via commande rfcp
   - analyse fichiers en série

   } parallélisation copie /analyse

Jobs de type 4 sur un même WN

Conclusion

❶ Accès depuis une UI de fichiers sur disque local / SE



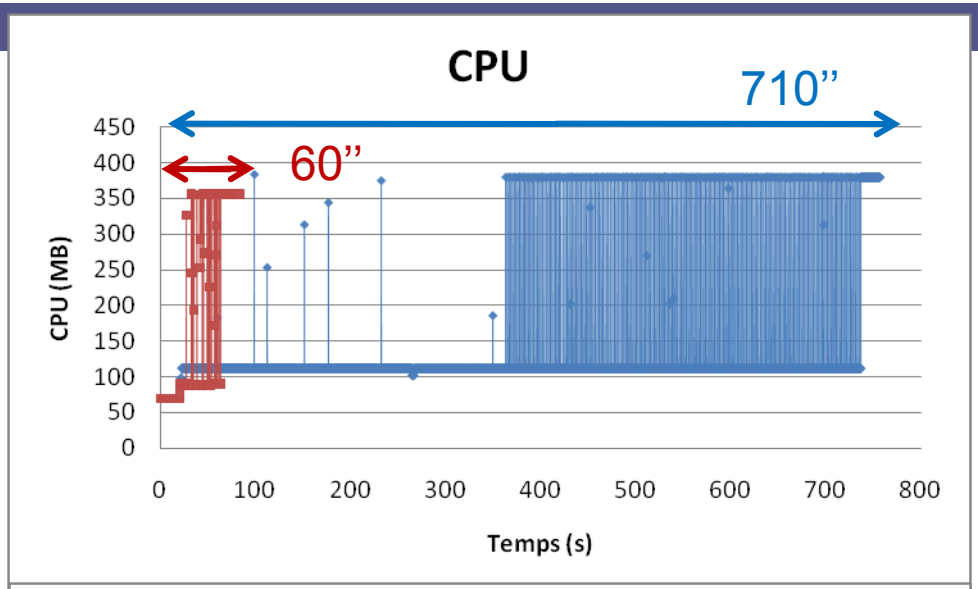
Chainer les fichiers et les ouvrir en même temps avec PyROOT

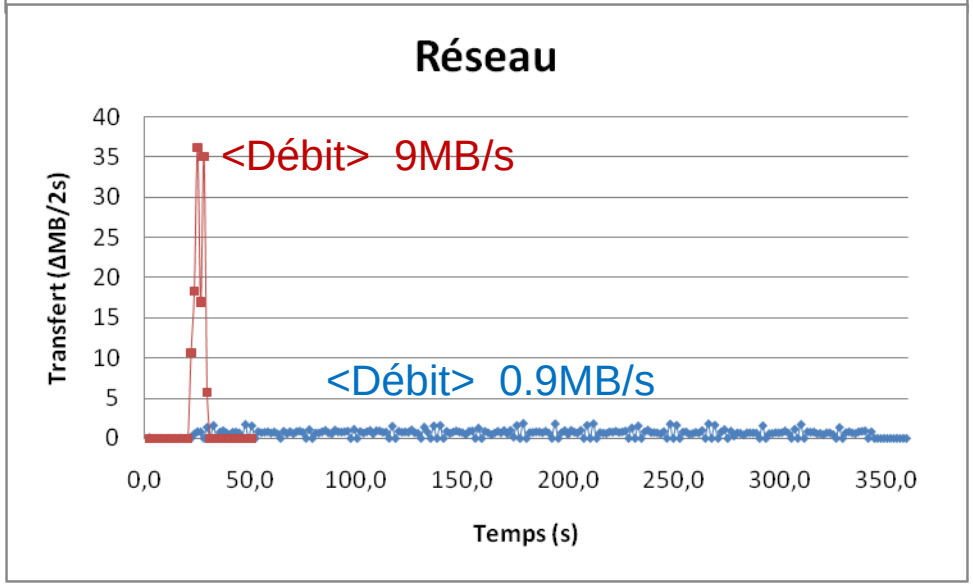
(200 fichiers : tot = 500Mo)

Fichiers sur Data_Local (gpfs) ou sur le SE (rfio) du Lapp

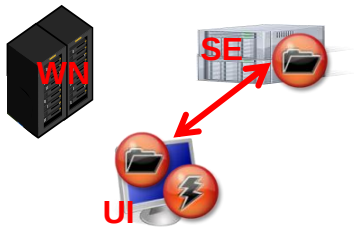
- Accès plus rapide vers Data_Local que SE
- CPU_{max} 380MB
- $Réseau_{max}$ $17MB.s^{-1}$
- $Réseau_{max}$ $2MB.s^{-1}$

} Protocoles ≠





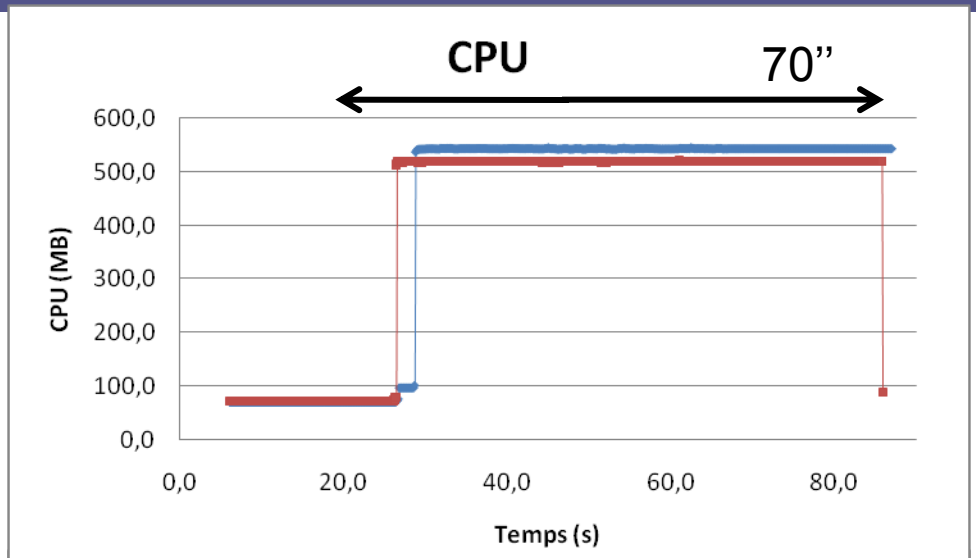
❶ Accès depuis une UI de fichiers sur disque local / SE

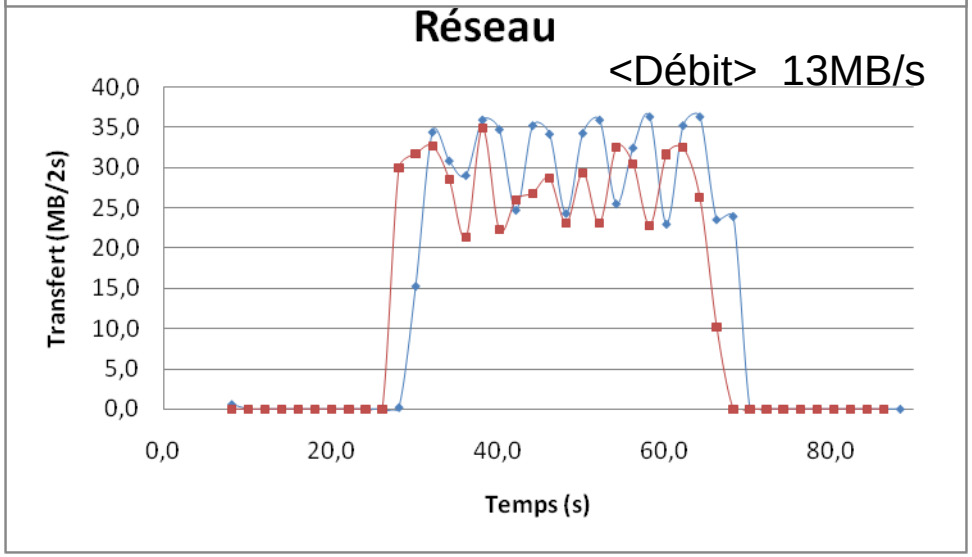


Un seul fichier de 500Mo

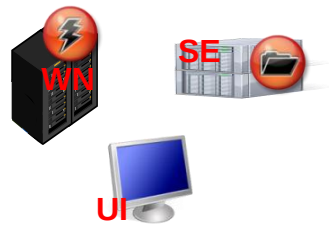
(5int, 1 double, 115 float)

- Accès plus rapide vers Data_Local que SE (rfio) (5" de différence)
- CPU_{max} 530MB
- $Réseau_{max}$ 18MB.s^{-1}
- Protocoles ≠ mais effet atténué car 1 seul fichier



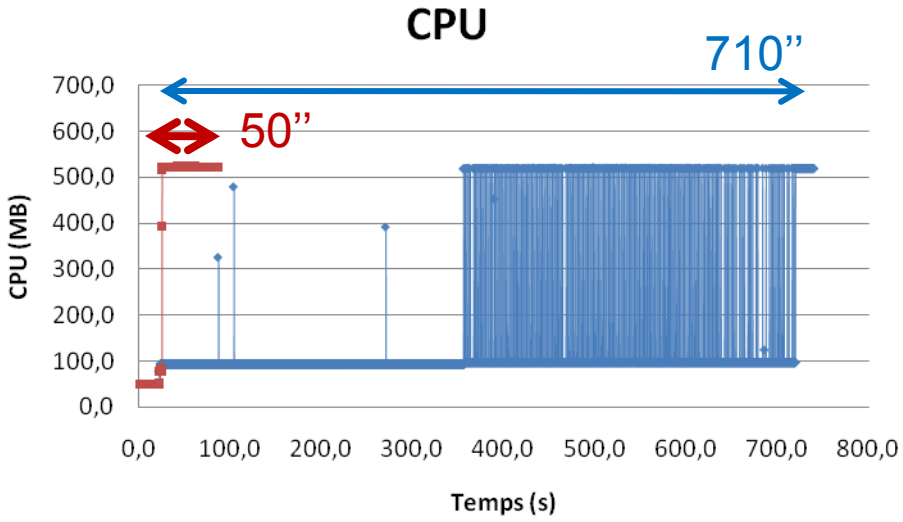


❷ Accès depuis un WN de fichiers sur SE



200 fichiers ou un seul fichier de 500Mo (5int, 1 double, 115 float)

- Accès plus rapide avec un seul fichier que 200
- Temps ~ que depuis une UI
- Temps < que depuis une UI
- CPU_{max} 530MB
- $Réseau_{max}$ $17MB.s^{-1}$
- $Réseau_{max}$ $5MB.s^{-1}$



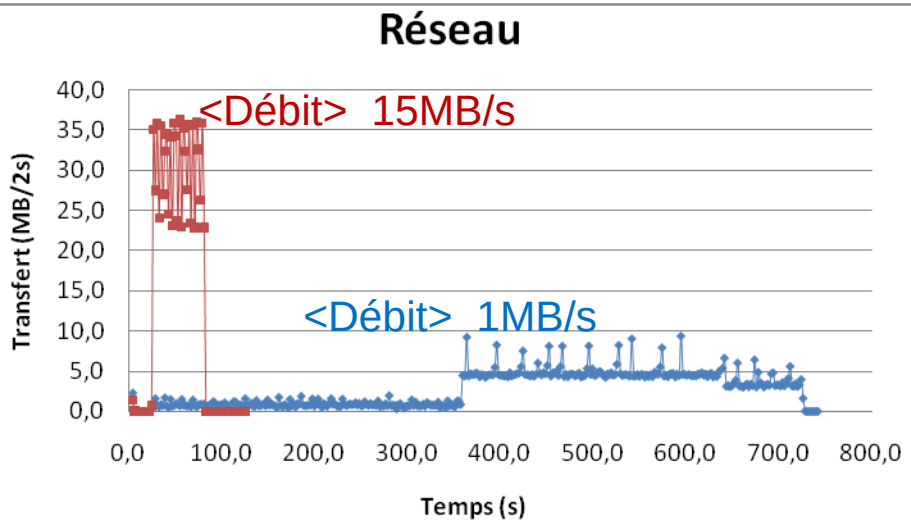
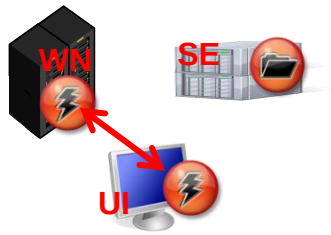


Tableau résumé d'accès à distance

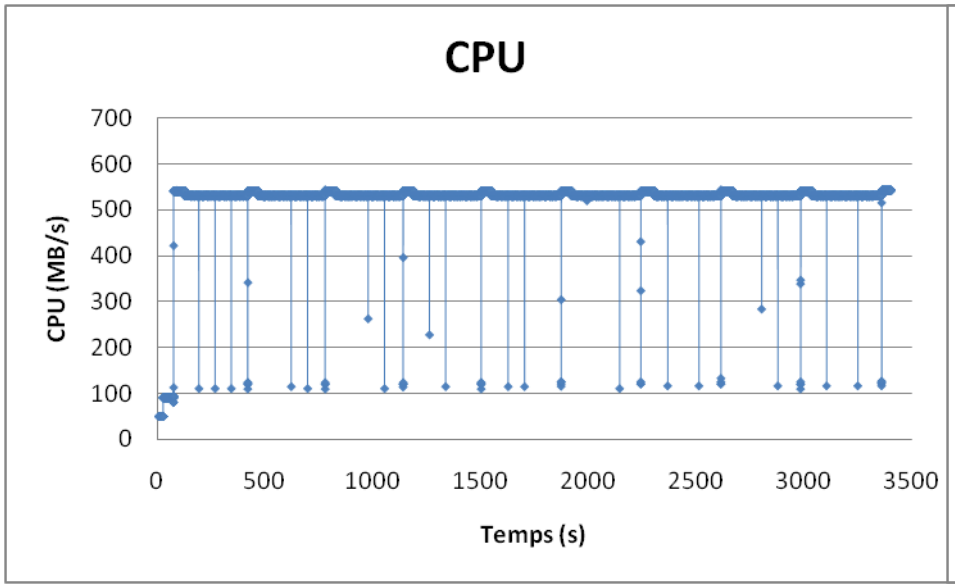
	CPU	Données	Protocole	<NetWork(MB/s)>	Temps (s)
200 fichiers	UI	SE	open rfio	0.9	710
	UI	Data_Local	open	9.1	60
	WN	SE	open rfio	1.0	710
1 fichier mergé	UI	SE	open rfio	13.0	70
	UI	Data_Local	open	13.0	65
	WN	SE	open rfio	15.0	50

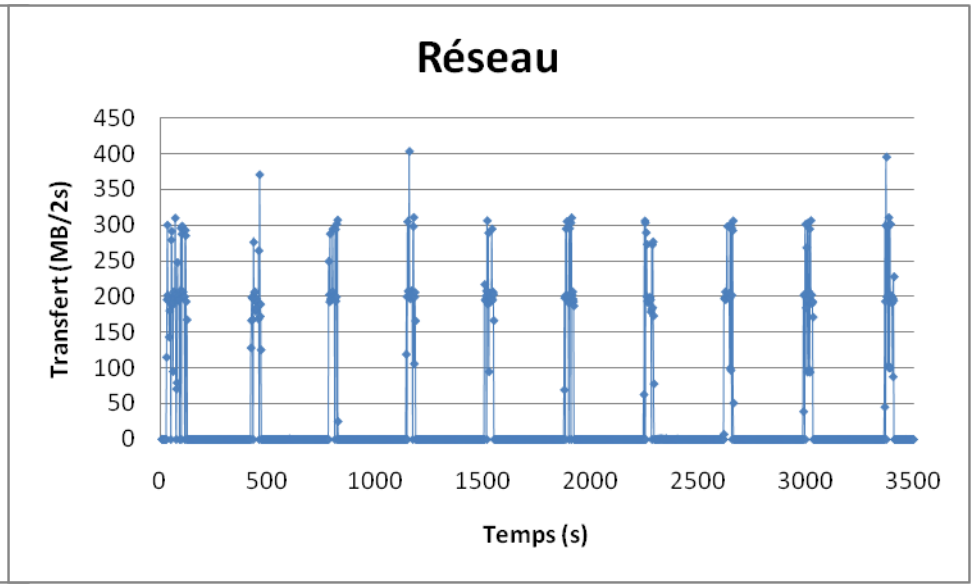
- Merger les fichiers diminue temps total car moins d'I/O
- Pas d'effet de cache car le taux de renouvellement des fichiers contenus dans le cache est très élevé

❸ Accès depuis une UI de fichiers sur disque local / SE

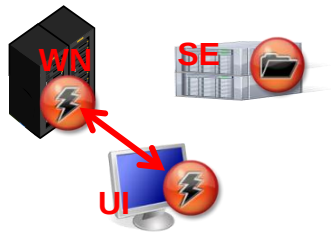


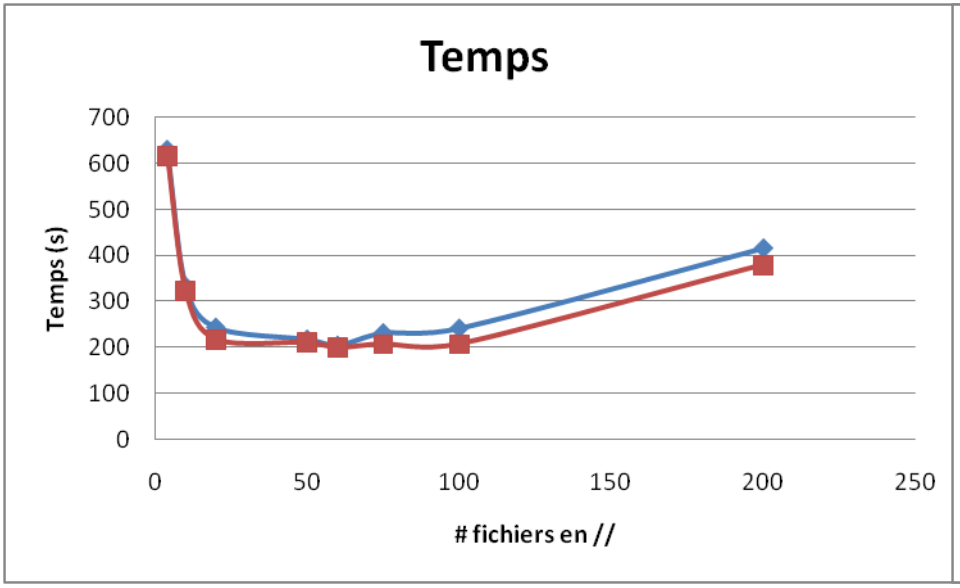
- Trois étapes itératives :
  - copier n fichiers localement (rfcp parallélisés)
  - analyser les n fichiers pendant copie des suivants
  - effacer fichiers analysés

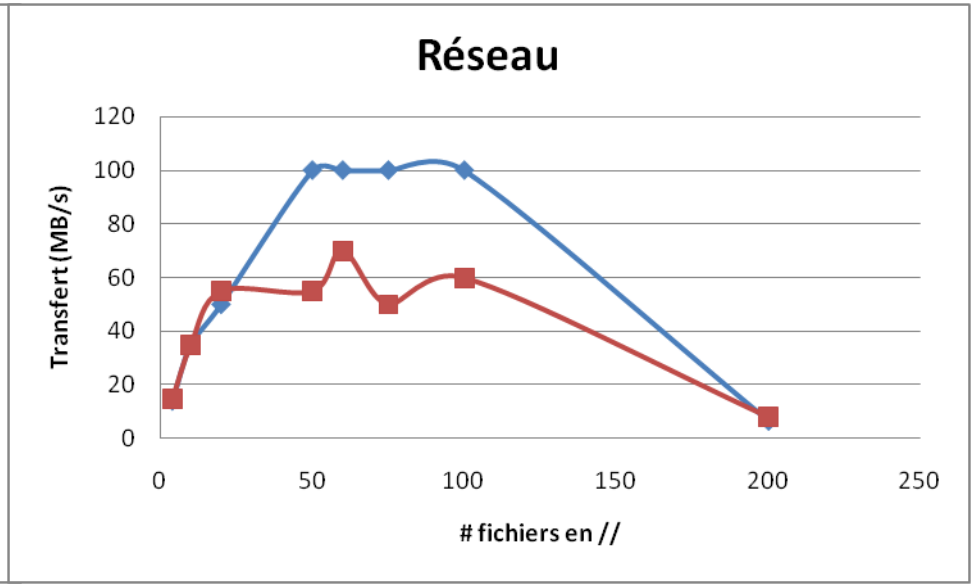




❸ Conclusion





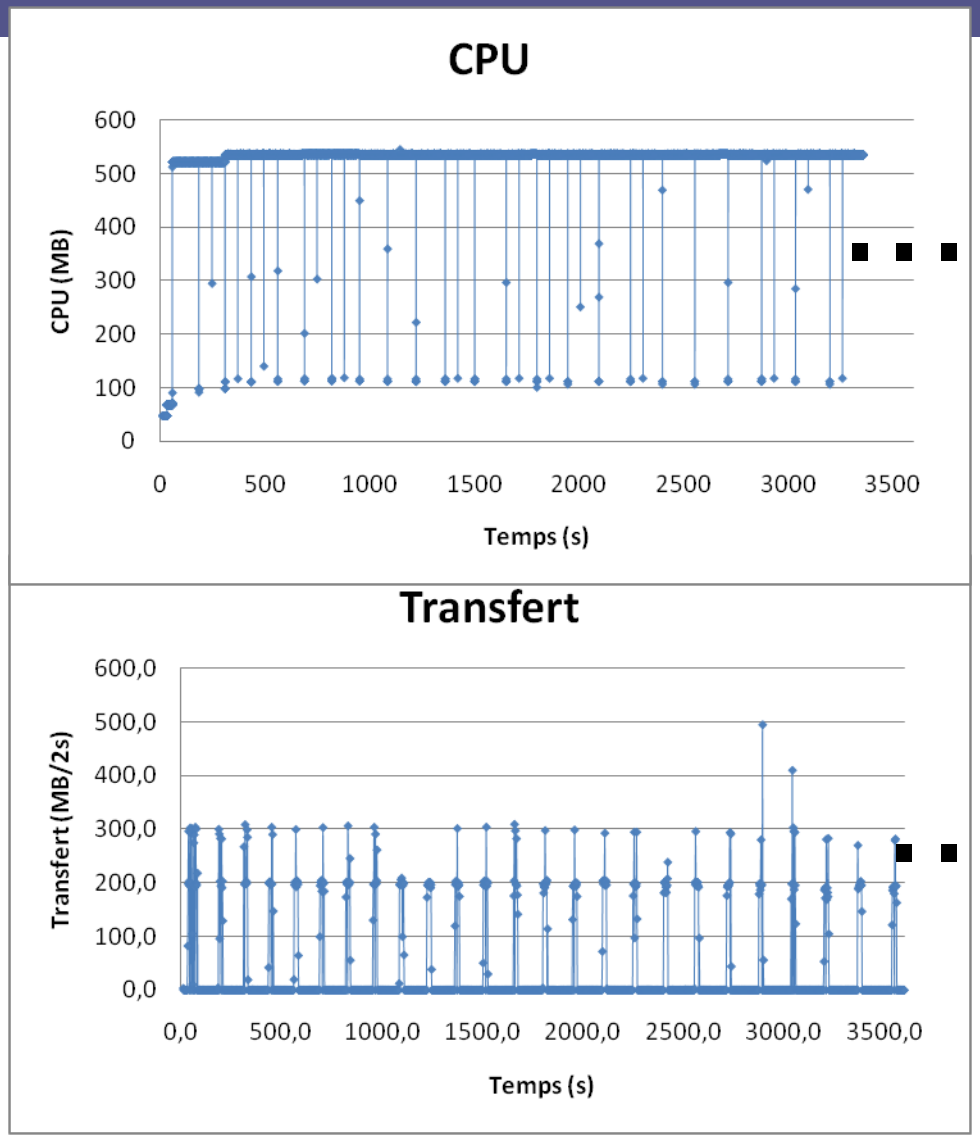


- Temps rfcp >> rfio lorsque le nombre de fichiers augmente
- Réseau utilisé plus optimal avec plusieurs fichiers copiés en même temps
- Max du réseau interne Lapp de 1Gb/s

❹ Accès depuis un WN de fichiers sur SE

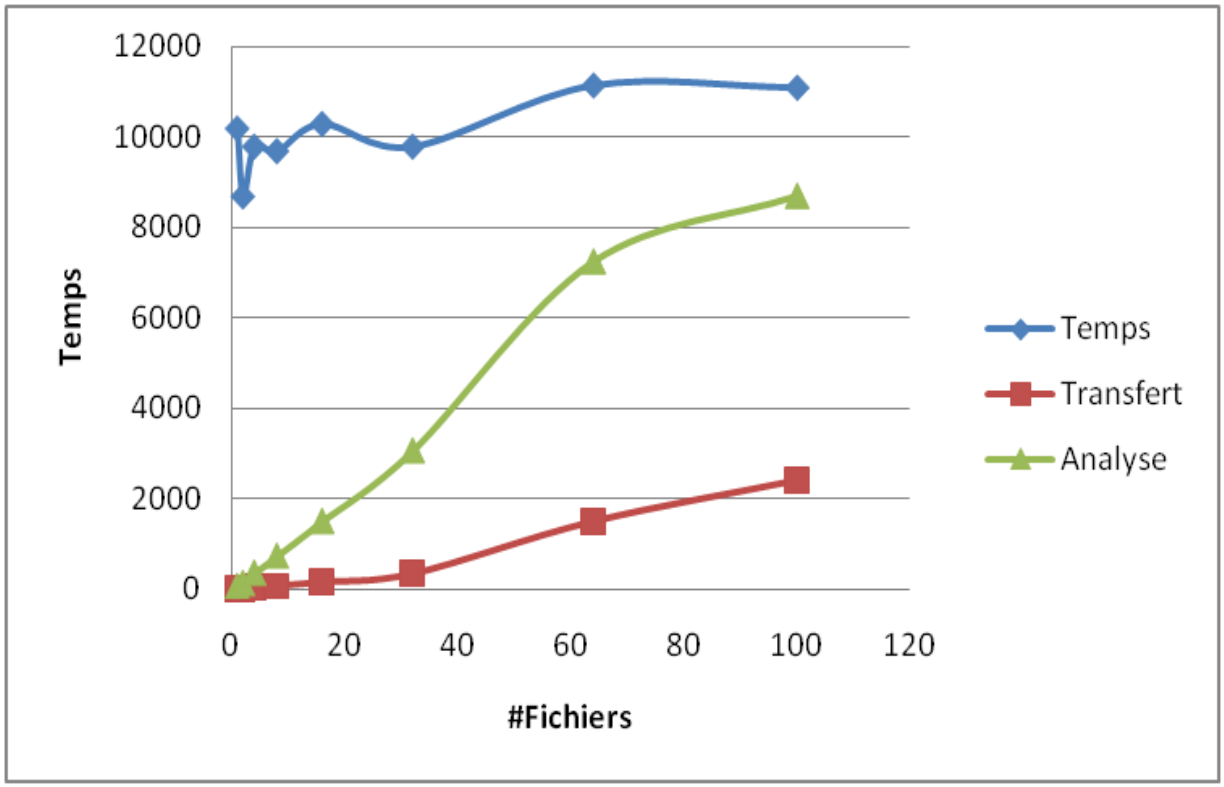
Closer to reality : fichiers de 1G0

- Essais avec N fois X fichiers de 1Go X.N=100
- $t_{calcul} > t_{transfert}$ et t_{calcul} α X $\Rightarrow t_{tot} \approx 100$"/1Go si X<40
- Limitation vient du calcul



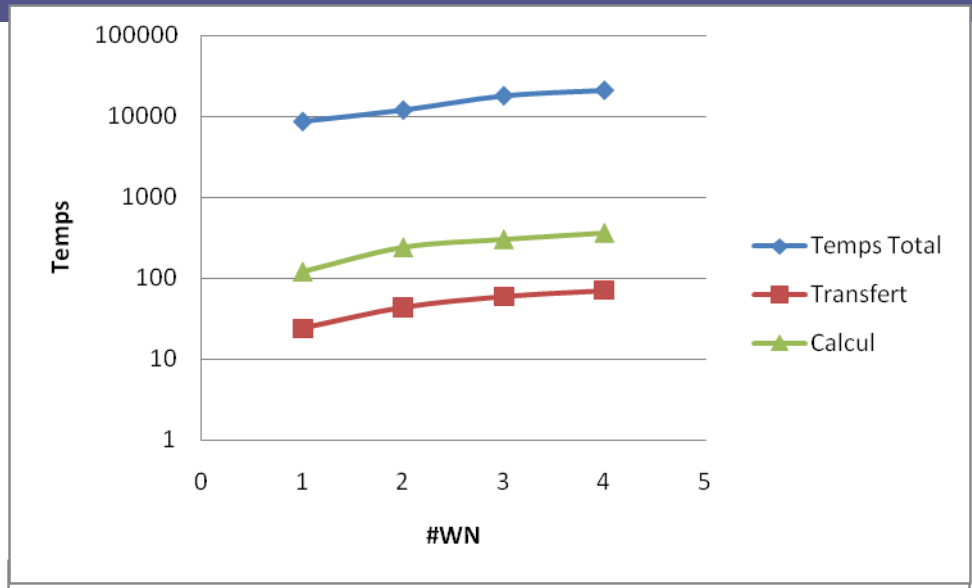
❹ Résumé

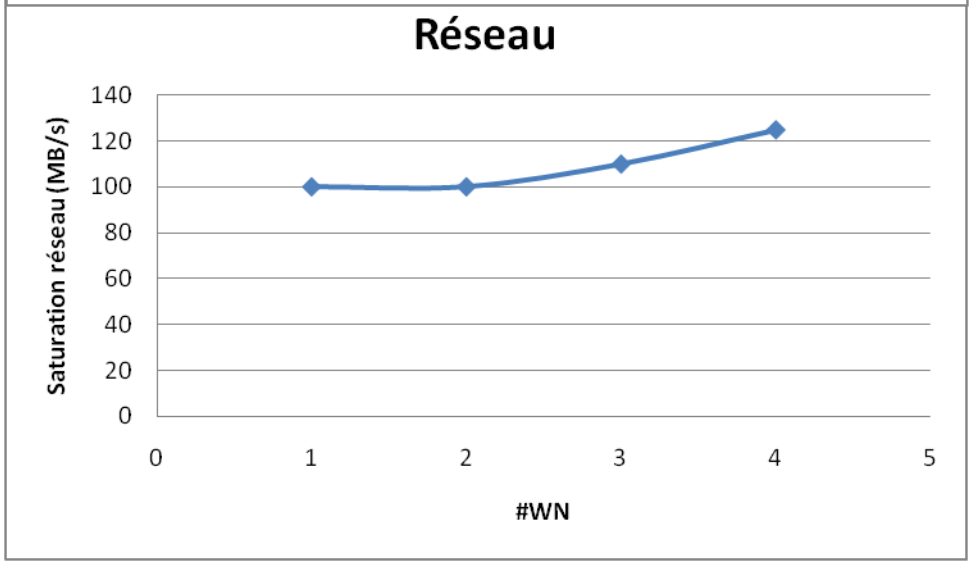
- Temps total pour transfert en // et calcul est équivalent qq soit le nombre de fichiers < 40
- Limitation vient du calcul



4 Accès depuis un WN de fichiers sur SE
Jobs sur un même WN (blade 4 cores)

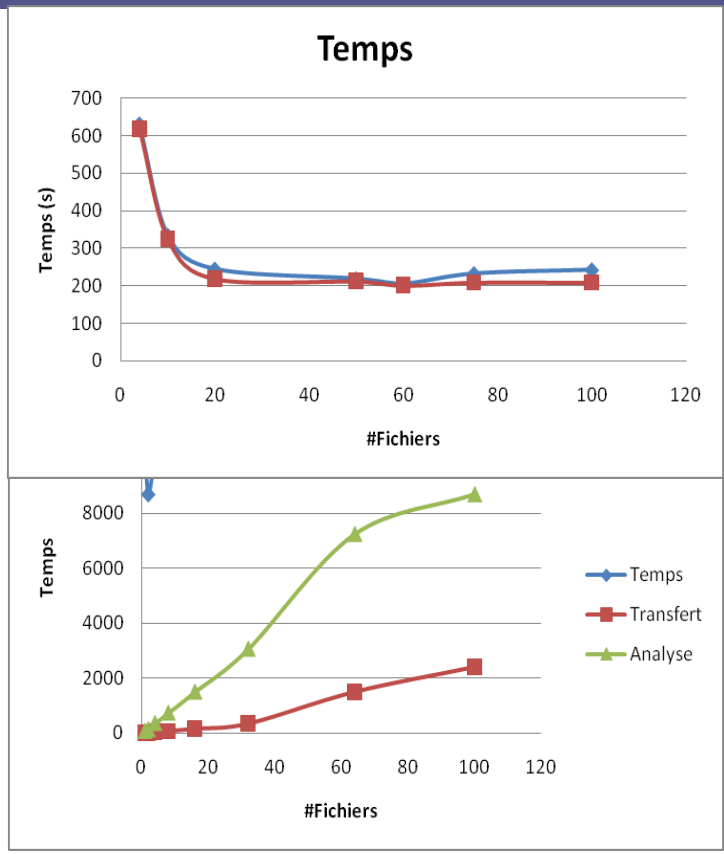
- Blade de 4 WN sortit spécialement pour ces tests (période calme : Atlas_P)
- X jobs mêmes WN (X=1,2,3,4)
- Travail en rfcp //
- Réseau même dans 4 cas
- Les jobs se pénalisent entre eux : la vitesse lecture/écriture des données sur le disque du blade est bloquante.
- Max du réseau cluster WN↔SE 2*1Gb/s





❹ Conclusions

- en “normal” :
  - cpu_{max} 500MB
  - <débit>: 17 et $débit_{max}$: 50MB/s
- en // : $point_{min}$ avec 60fichiers (200MB)
  - UI : $débit_{max}$ = 70MB/s
  - WN : $débit_{max}$ = 110MB/s
- Fichiers 1Go : même effet (2Go en //)
- Sur même WN
  - Transferts fichiers : même débit (100MB/s)
  - CPU partagées => compétition entre jobs
  - Limite d’accès aux données copiées sur disque du blade
  - Pas ce genre de problème avec des open rfio



Conclusions

		CPU	Data	Protocole	<NetWork> (MB/s)	
1 Fichier de 1G0	❶	UI	UI	open	13.0	Résultat site dépendant (LAPP : gpfs + réseau)
	❶	UI	SE	rfio:/dpm/…	13.0	
	❷	WN	SE	rfio:/dpm/…	15.0	Limitation : activité SE et occupation du réseau WN/SE **non prévisibles** (fonction du type des jobs qui tournent à un moment donné)
100 fichiers de 1G0 en //	❸	UI	SE	rfcp local UI	70.0	
	❹	WN	SE	rfcp local WN	110.0	Limitation **: vitesse lecture/écriture** sur disque WN

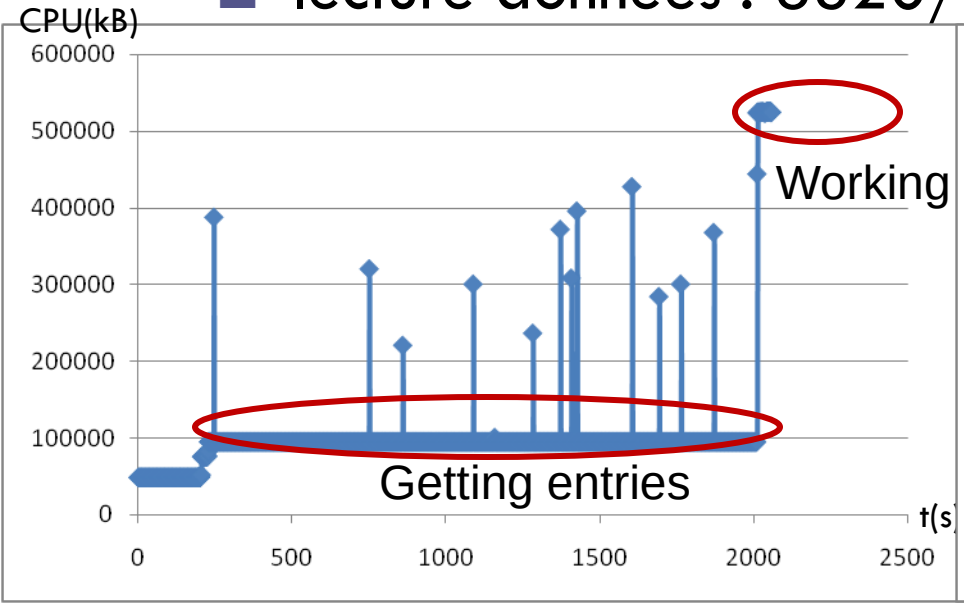
- Fichiers mergés plutôt que série de petits fichiers
- “open rfio” plus performant que rfcp mais dépend de l’activité sur cluster
- Capacités maximum du réseau Lapp interne et cluster atteint (1Gb/s)

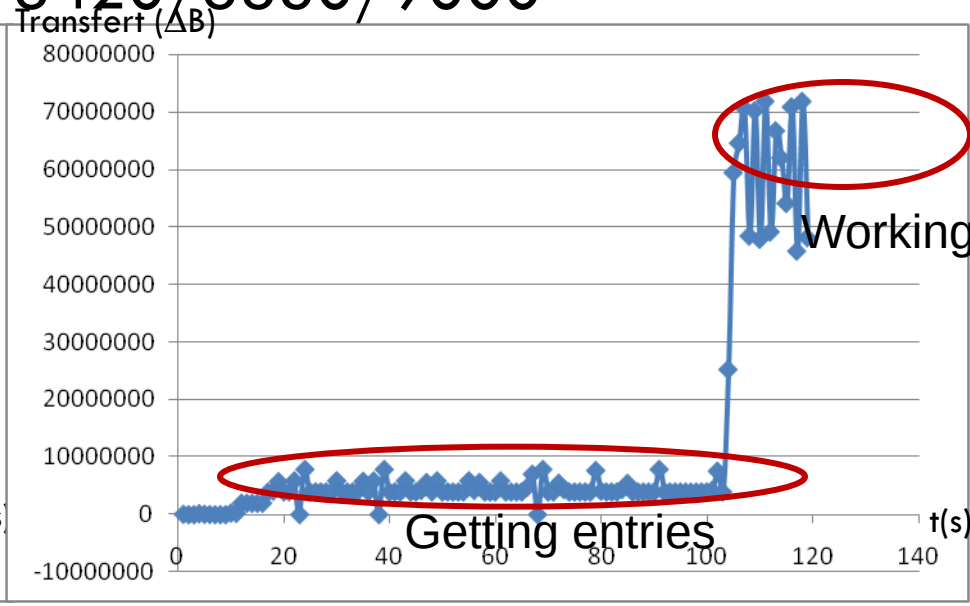
Open rfio des fichier de 1Go

- 1/2/3/4 WN sur le même blade avec 100 open rfio
- 1 WN open_rfio et 1 WN rfcp en // sur même blade
- 2 WN open_rfio et 2 WN rfcp en // sur même blade

1/2/3/4 open rfio sur blade

- Temps :
  - import root et dépendances : 11.6" et 0.4"
  - add chain 0"
  - getting entries : 180"
  - lecture données : 8320/8420/8830/9000"





Plusieurs jobs sur même blade de 4 WN

	Getting entries	rfcp	Lecture données (rfcp)	Lecture données (rfio)	Total (rfcp)	Total (rfio)
1rfio+1rfcp	200”	27”	130”	7920”	8724”	8120”
2rfio+2rfcp	200”	41”	230”	9180”	13300”	9400”
3rfio+1rfcp	204”	26”	130”	9150”	9120”	9400”
1rfio+3rfcp	230”	54”	290”	8700”	17000”	9000”