

ETUDE DE PERFORMANCES

Uniserve - Apache

Résumé : Ce document a pour vocation de présenter les différences de comportements entre le serveur d'application Uniserve, à caractère événementiel, et le serveur Web Apache.

Historique de mise à jour

Version	Date	Auteur	Objet
1.0	01/02/2005	Philippe Le Goff	Version initiale

Documents associés

Titre	Auteur	Référence	Version
Uniserve			

Mots clés

Uniserve, Rebol, Apache, serveur Web

Sommaire

1. OBJET.....	5
2. CONTEXTE	6
2.1 Caractéristiques des machines :	6
2.1.1 Serveur Cible :	6
2.1.2 Serveur Injecteur.....	7
2.2 Tests :	7
2.2.1 Types de tests :	7
2.2.2 Fichiers de tests :	8
2.2.3 Versions logicielles utilisées :	8
3. PARTIE I : INJECTION DEPUIS MACHINE BLADE VERS MACHINE CIBLE ML370....	10
3.1 Tests Uniserve, Apache, Apache + Magic! :	11
3.1.1 Temps de réponse Uniserve 0.9.9 :	11
3.1.1.1 Test Uniserve Local	11
3.1.1.2 Tests Uniserve en Remote.....	13
3.1.2 Temps de réponse Apache 1.3.27 :	15
3.1.3 Tests Apache 1.3.27 seul	15
3.1.4 Temps de réponse Apache 1.3.27 + Magic! 1.2.....	16
3.2 Tests Uniserve R2 avec injecteur BLADE : consommation mémoire et charge réseau	17
3.2.1 Evolution du process starter.r	17
3.2.2 Analyse VmStat :	17
3.2.3 Analyse Réseau :	18
3.2.3.1 Nombre de paquets transmis :	18
3.2.3.2 Nombre d'octets/ s :	18
3.3 Tests Apache 1.3.27 R2 avec injecteur Blade – consommation mémoire et charge réseau	19
3.3.1 Évolution du process httpd	19
3.3.1.1 Consommation CPU (6 tirs)	19
3.3.2 Analyse VMSTAT :	19
3.3.2.1 Détail VMSTAT d'un run :	20
3.3.3 Analyse Réseau :	20
3.3.3.1 Nombre de paquets transmis :	20
3.3.3.2 Détail d'un pic :	21
3.3.3.3 Nombre d'octets/ s :	21
3.4 Synthèse de la partie I	22
4. PARTIE II : INJECTION DEPUIS SERVEUR ML370 VERS LA CIBLE BLADE	23
4.1 TESTS EN LOCAL (L1, L2, L3) et EN REMOTE sur serveur Blade :	24
4.1.1 Description des tests :	24
4.1.2 Tests en local :	25
4.1.3 Test en remote.....	25
4.2 TESTS DE PUISSANCE :	27
4.2.1 Temps de réponse test R99-1 :	27
4.2.2 Temps de réponse test R99-2 avec de 2 à 9 injecteurs :	29

4.2.3 Analyse VMStat du processus starter.r	30
4.3 Synthèse de la partie II	30

5. ANALYSE DES RESULTATS :.....32

5.1 Comparaison Uniserve ML370 versus Blade en local :.....	32
5.2 Comparaison Uniserve ML370 versus Blade en Remote :.....	32
5.3 Tests de puissance :.....	32
5.4 Comparaison Apache / Uniserve :.....	33

6. ANNEXE 1 :.....35

Comparatif de versions pour REBOL	35
---	----

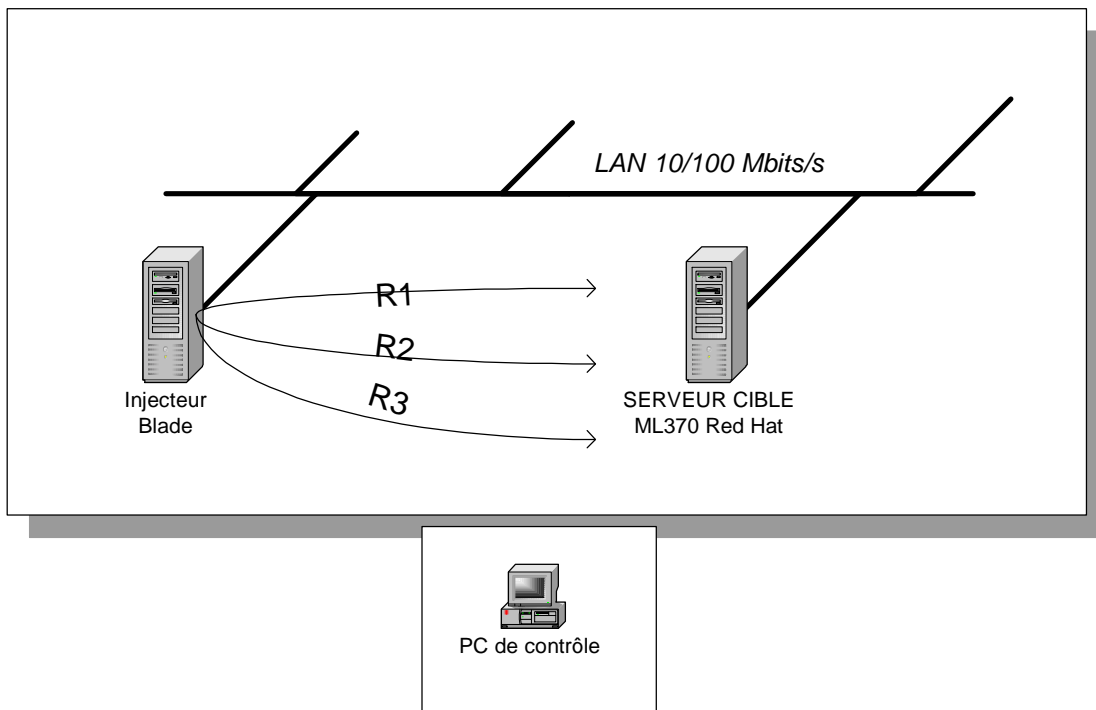
1. OBJET

Ce document a pour objet de présenter les différences de comportement du serveur multi-protocoles et événementiel Uniserve, de la société SOFTINNOV et le serveur Web APACHE de l'Apache Foundation. Des éléments de comparaison sont également fournis pour le couple Apache / Magic! 1.2, ce dernier serveur étant la production d'Olivier Auverlot.

L'étude porte essentiellement pour ce premier lot sur les temps de réponse des différents serveurs. Les évolutions de mémoire, nombre de processus feront l'objet d'une seconde étude.

2. CONTEXTE

Les tests de performance sont réalisés sur la configuration suivante :



2.1 Caractéristiques des machines :

Les caractéristiques des serveurs sur lequel s'est déroulée l'étude sont les suivantes :

2.1.1 Serveur Cible :

MARQUE	COMPAQ
TYPE	ML370
NOM	bergson
OS	Linux 2.4.9-e.27smp ES 2.1 #1 SMP Tue Aug 5 15:49:54 EDT 2003 i686 (RedHat)
RAM	MemTotal: 254440 kB
CPU	vendor_id : GenuineIntel cpu family : 6 model : 11 model name : Intel(R) Pentium(R) III CPU family 1266MHz stepping : 1 cpu MHz : 1263.535 cache size : 512 KB

2.1.2 Serveur Injecteur

MARQUE	IBM
TYPE	BladeCenter HS40
OS	Linux Fedore Core 3.0 – Noyau 2.6.9-1
RAM	MemTotal: 4096 kB
CPU	4 XeonMP cpu : 2.2 GHz

Ces deux serveurs sont situés sur le même réseau LAN, en 10/100 Mbits/s.

2.2 Tests :

Les tests effectués portent sur le service HTTPd du serveur Uniserve, lequel a été adapté pour écouter sur le port 8081. Ils serviront à comparer les performances du serveur Apache 1.3.27 et celles d'Uniserve 0.9.9.

2.2.1 Types de tests :

Trois types de tests sont réalisés : il s'agit de requêtes GET sur des pages statiques. L'injection se fait via des requêtes rebol "read..." .

Rappel : une requête READ correspond à un "open", puis à la lecture des données sur le port ouvert.

Type	Nature	Commande Rebol
L1	Local, 1000 requêtes GET en boucle sur le même fichier	loop 1000 [read <u>http://localhost:8081/index.html</u>]
L2	Local, 1000 requêtes GET sur le fichier index1.html, puis 1000 sur index2.html, ... Pour un total de 5000 requêtes GET.	i: 0 loop 5 [i: i + 1 loop 1000 [read to-url rejoin ["http://localhost:8081/index" i ".html"]]]
L3	1000 requêtes GET accédant de façon aléatoire à l'un des fichiers index1.html à index5.html	loop 1000 [read to-url rejoin ["http://localhost:8081/index" random 5 ".html"]]
R1	Idem L1, mais en remote	Idem L1 local avec IP serveur cible
R2	Idem L2, mais en remote	Idem L2 local avec IP serveur cible
R3	Idem L3, mais en Remote	Idem L3 local avec IP serveur cible
R99	R3 avec 5000 requêtes et 3 injecteurs	loop 5000 [read to-url rejoin ["http://IP-CIBLE/index" random 5 ".html"]]

2.2.2 Fichiers de tests :

Les mêmes fichiers html de tests sont utilisés dans les deux cas.

Type	Taille
Index1.html	8.6 ko
Index2.html	8.6 ko
Index3.html	8.6 ko
Index4.html	8.6 ko
Index5.html	8.6 ko

Les fichiers .html comprennent cependant des liens vers des images de type GIF, JPG, et PNG :

Type	Liens vers ...	Taille Images relatives aux liens
Index1.html	logo.gif	1.9 ko
Index2.html	logo.gif, 2.jpg	1.9 ko, 44 ko
Index3.html	logo.gif, 2.jpg, 3.png	1.9 ko, 44 ko, 66.1 ko
Index4.html	logo.gif, 2.jpg, 3.png, 4.gif	1.9 ko, 44 ko, 66.1 ko, 79.6 ko
Index5.html	logo.gif, 2.jpg, 3.png, 4.gif, 5.png	1.9 ko, 44 ko, 66.1 ko, 79.6 ko, 145 ko

2.2.3 Versions logicielles utilisées :

PRODUIT	VERSION	Remarques
REBOL	Core 2.5.6.4.2	/Pro
APACHE	1.3.27	mod_ssl/2.8.12 OpenSSL/0.9.6b PHP/4.3.2 Timeout 300 KeepAlive Off MaxKeepAliveRequests 100 KeepAliveTimeout 15 MinSpareServers 5 MaxSpareServers 20 StartServers 8 MaxClients 150 MaxRequestsPerChild 1000
UNISERVE	0.9.9	Service HTTPd avec log as-Apache
STAT_Dim	6.0	Version Linux
MAGIC!	1.2	

Remarques :

1/ Le time-out par défaut Réseau Rebol n'a pas été modifié.

2/ Le service HTTPd d'Uniserve a été modifié de manière à tracer les accès dans des logs identiques à ceux obtenus avec le serveur Apache .

Soit par exemple des lignes :

```
127.0.0.1 - - [9/Dec/2004:13:27:58.719748 +0000 ] "GET wwwroot/index1.html" 200 8599 19002
127.0.0.1 - - [9/Dec/2004:13:27:58.72312 +0000 ] "GET wwwroot/index2.html" 200 8617 19003
```

Les logs Uniserve indiquent aussi le nombre d'accès (19002 et 19003 dans l'exemple ci-dessus).

3. PARTIE I : INJECTION DEPUIS MACHINE BLADE VERS MACHINE CIBLE ML370

3.1 Tests Uniserve, Apache, Apache + Magic! :

3.1.1 Temps de réponse Uniserve 0.9.9 :

Les tableaux suivants synthétisent les résultats pour la durée moyenne des tests .
Les tests en locaux servent à caractériser la réponse d'une machine spécifique, en dehors des aspects réseaux.

3.1.1.1 Test Uniserve Local

Uniserve 0.9.9 – L1 Local

Type Test et Run	Temps total (s)	Temps moy. (s)	Temps min (s)	Temps max (s)
L1 – 1	4,093	0,004097	0,0031089	0,012290001
L1 – 2	4,2263	0,0042305	0,00331115	0,014839
L1 – 3	5,2005	0,0052058	0,003097	0,78480
L1 – 4	4,1659	0,0041701	0,0030803	0,013899
L1 – 5	5,66163	0,0056673	0,003040	1,5919
Moyenne	4,669466	0,00467414	0,00312747	0,4835456

Remarques :

Si le temps minimum moyen par requête reste constant (4 – 5 ms), le temps maximum moyen varie entre 120 ms et 1,6 s.

A titre de comparaison, le même test mené avec Magic! 1.2 conduit à un temps global de 45 secondes en moyenne.

Uniserve 0.9.9 – L2 Local

Type Test et Run	Temps total (s)	Temps moy. (s)	Temps min (s)	Temps max (s)
L2 – 1	22,3143	0,0044637	0,00309753	1,17969
L2 – 2	28,7882	0,0057588	0,003097	5,0277
L2 – 3	21,3715	0,0042751	0,0030975	0,018798
L2 – 4	29,7408	0,005949	0,003097	5,63960
L2 – 5	21,4128	0,0042834	0,003299	0,01599
Moyenne	24,72552	0,004946	0,003137606	2,3763556

Remarques :

L2 – 1 : 1 point à plus d'une seconde (no 4910 / 1.179 s) (get sur index5.html)

L2 – 2 : 3 points dont 1 à 5 secondes

L2 – 4 : 3 points singuliers dont 1 à 5.6 s (940-941) (get sur index1.html)

Uniserve 0.9.9 – L3 Local

Type Test et Run	Temps total (s)	Temps moy. (s)	Temps min (s)	Temps max (s)
L3 – 1	4,07190	0,0040759	0,0030975	0,01160
L3 – 2	4,1871	0,0041913	0,0030994	0,01200
L3 – 3	4,24557	0,004249	0,0031395	0,014490
L3 – 4	4,211	0,004215	0,0030975	0,0122
L3 – 5	4,3759	0,00438	0,0030975	0,01290
Moyenne	4,218294	0,00422224	0,00310628	0,012638

Remarques :

Dans le cas d'un accès aléatoire, les temps moyens sont très stables (min 3.1 ms, max 130 ms).

⇒ On utilise les ressources machines (noyau) . La charge en locale correspond à celle du traitement de la demande réseau par le service Uniserve Httpd plus celle générant la requête (read). Dans les deux cas, c'est VM Rebol qui est sollicitée.

3.1.1.2 Tests Uniserve en Remote

Uniserve 0.9.9 – R1 Remote avec injecteur PC, hors LAN

```
loop 1000 [read http://IP-CIBLE/index.html ]
```

Type Test et Run	Temps total (s)	Temps moy. (s)	Temps min (s)	Temps max (s)
R1 – 1	65,8318	0,0659	0,0539	3,0583
R1 – 2	66,0654	0,06613	0,0546	0,33
R1 – 3	64,0067	0,06407	0,05380	0,445
R1 – 4	77,98	0,07806	0,0548	0,3847
R1 – 5	65,4622	0,06553	0,05390	0,4227
Moyenne	67,8692	0,0679	0,0542	0,9281

Uniserve 0.9.9 – R2 Remote avec injecteur PC, hors LAN

```
i: 0
```

```
loop 5 [ i: i + 1
```

```
loop 1000 [ read to-url rejoin ["http://IP-CIBLE/index" i ".html" ] ]
```

Type Test et Run	Temps total (s)	Temps moy. (s)	Temps min (s)	Temps max (s)
R2 – 1	331,14	0,0662	0,0430	0,5889
R2 – 2	334,4	0,0669	0,0539	3,2548
R2 – 3	361,07	0,0722	0,0533	1,157
R2 – 4	298,9	0,0598	0,0529	0,49
R2 – 5	334,731	0,067	0,0542	0,5842
Moyenne	332,05	0,0664	0,0515	1,2150

Uniserve 0.9.9 – R3 Remote avec injecteur PC, hors LAN

```
loop 1000 [read to-url rejoin ["http://IP-CIBLE/index" random 5 ".html"]]
```

Type Test et Run	Temps total (s)	Temps moy. (s)	Temps min (s)	Temps max (s)
R3 – 1	63,89	0,0639	0,0543	0,3284
R3 – 2	72,096	0,0721	0,0351	3,277
R3 – 3	62,7079	0,06277	0,0534	0,3307
R3 – 4	75,866	0,0759	0,0498	6,074
R3 – 5	66,071	0,0661	0,0535	0,37470
Moyenne	68,1262	0,0682	0,0492	2,0770

Remarques :

Une requête prend en moyenne 68 ms depuis un PC situé hors LAN. (configuration 10/100 Mbs côté PC).
Les temps maximum varient fortement 300 ms et 3-6 secondes.

Les fortes variations de temps maximum sont sans doute liées au fait que le PC est situé dans un réseau 10 MBts, à la différence du serveur cible, et qu'un switch est en intermédiaire.

Uniserve 0.9.9 – R1 Remote depuis l'IBM HS40 Blade Injecteur, en LAN

```
loop 1000 [read http://IP-CIBLE/index.html ]
```

Les négociations sont forcées en 100Mbps/s full duplex.

Type Test et Run	Temps total (s)	Temps moy. (s)	Temps min (s)	Temps max (s)
R1 – 1	3,566	0,0035	0,00279	0,00949
R1 – 2	3,57	0,00357	0,00279	0,00949
R1 – 3	3,5583	0,00356	0,00279	0,0138
R1 – 4	3,480	0,00348	0,00279	0,0093
R1 – 5	3,458	0,00346	0,00279	0,00949
Moyenne	3,5265	0,0035	0,00279	0,01031

Uniserve 0.9.9 – R2 Remote depuis l'IBM HS40 Blade Injecteur, en LAN

```
i: 0  
loop 5 [ i: i + 1  
    loop 1000 [ read to-url rejoin ["http://IP-CIBLE/index" i ".html"] ] ]
```

Type Test et Run	Temps total (s)	Temps moy. (s)	Temps min (s)	Temps max (s)
R2 – 1	17,428	0,00349	0,00279	0,0131
R2 – 2	17,467	0,00349	0,00279	0,0136
R2 – 3	17,493	0,00349	0,00270	0,0135
R2 – 4	17,468	0,00349	0,00279	0,0143
R2 – 5	17,484	0,00349	0,00279	0,0135
Moyenne	17,46800	0,00349	0,00277	0,01360

Uniserve 0.9.9 – R3 Remote depuis l'IBM HS40 Blade Injecteur, en LAN

```
loop 1000 [read to-url rejoin ["http://IP-CIBLE/index" random 5 ".html"]]
```

Type Test et Run	Temps total (s)	Temps moy. (s)	Temps min (s)	Temps max (s)
R3 – 1	3,51	0,0035	0,00279	0,013
R3 – 2	3,527	0,0035	0,00279	0,0178
R3 – 3	3,518	0,0035	0,00279	0,0127
R3 – 4	3,498	0,0035	0,00279	0,0094
R3 – 5	3,506	0,00351	0,00279	0,0095
Moyenne	3,51180	0,00350	0,00279	0,01248

3.1.2 Temps de réponse Apache 1.3.27 :

3.1.3 Tests Apache 1.3.27 seul

Apache 1.3.27 – R1 Remote depuis le Blade Injecteur, en LAN

```
loop 1000 [read http://IP-CIBLE/softinnov/index.html ]
```

Type Test et Run	Temps total (s)	Temps moy. (s)	Temps min (s)	Temps max (s)
R1 – 1	3	0,0030	N/A	N/A
R1 – 2	3	0,0030	N/A	N/A
R1 – 3	2	0,0020	N/A	N/A
R1 – 4	2	0,0020	N/A	N/A
R1 – 5	3	0,0030	N/A	N/A
Moyenne	2,6	0,0026		

Remarques :

Dans la version du noyau utilisé pour la cible, il n'est pas possible de définir pour Apache une précision inférieure à la seconde. Les temps totaux sont donc à prendre dans cette perspective.

Apache 1.3.27 – R2 Remote depuis le Blade Injecteur, en LAN

```
i: 0  
loop 5 [ i: i + 1  
    loop 1000 [ read to-url rejoin ["http://IP-CIBLE/softinnov/index" i ".html" ] ]
```

Type Test et Run	Temps total (s)	Temps moy. (s)	Temps min (s)	Temps max (s)
R2 – 1	12	0,0024	N/A	N/A
R2 – 2	13	0,0026	N/A	N/A
R2 – 3	12	0,0024	N/A	N/A
R2 – 4	12	0,0024	N/A	N/A
R2 – 5	12	0,0024	N/A	N/A
Moyenne	12,2	0,0024		

Apache 1.3.27 – R3 Remote depuis le Blade Injecteur, en LAN

```
loop 1000 [read to-url rejoin ["http://IP-CIBLE/softinnov/index" random 5 ".html"]]
```

Type Test et Run	Temps total (s)	Temps moy. (s)	Temps min (s)	Temps max (s)
R2 – 1	2	0,0020	N/A	N/A
R2 – 2	2	0,0020	N/A	N/A
R2 – 3	2	0,0020	N/A	N/A
R2 – 4	2	0,0020	N/A	N/A
R2 – 5	2	0,0020	N/A	N/A
Moyenne	2	0,002		

3.1.4 Temps de réponse Apache 1.3.27 + Magic! 1.2

Apache 1.3.27 + Magic! 1.2 – R1 Remote depuis le Blade Injecteur, en LAN

```
loop 1000 [read http://IP-CIBLE/suivi/wwwroot/index.rhtml ]
```

Type Test et Run	Temps total (s)	Temps moy. (s)	Temps min (s)	Temps max (s)
R1 – 1	39	0,039	N/A	N/A
R1 – 2	39	0,039	N/A	N/A
R1 – 3	39	0,039	N/A	N/A
R1 – 4	39	0,039	N/A	N/A
R1 – 5	39	0,039	N/A	N/A
Moyenne	39	0,039	N/A	N/A

Remarques :

On voit que l'ajout du traitement de la requête après Apache par le serveur Magic! est préjudiciable aux performances avec un temps moyen par requête de 39 ms.

Apache 1.3.27 + Magic! 1.2 – R2 Remote depuis le Blade Injecteur, en LAN

i: 0

```
loop 5 [ i: i + 1
```

```
loop 1000 [ read to-url rejoin ["http://IP-CIBLE/suivi/wwwroot/index" i ".rhtml" ] ]
```

Type Test et Run	Temps total (s)	Temps moy. (s)	Temps min (s)	Temps max (s)
R2 – 1	196	0,039	N/A	N/A

Apache 1.3.27 + Magic! 1.2 – R2 Remote depuis le Blade Injecteur, en LAN

```
loop 1000 [read to-url rejoin ["http://IP-CIBLE/suivi/wwwroot/index" random 5 ".rhtml" ]]
```

Type Test et Run	Temps total (s)	Temps moy. (s)	Temps min (s)	Temps max (s)
R3 – 1	39	0,039	N/A	N/A

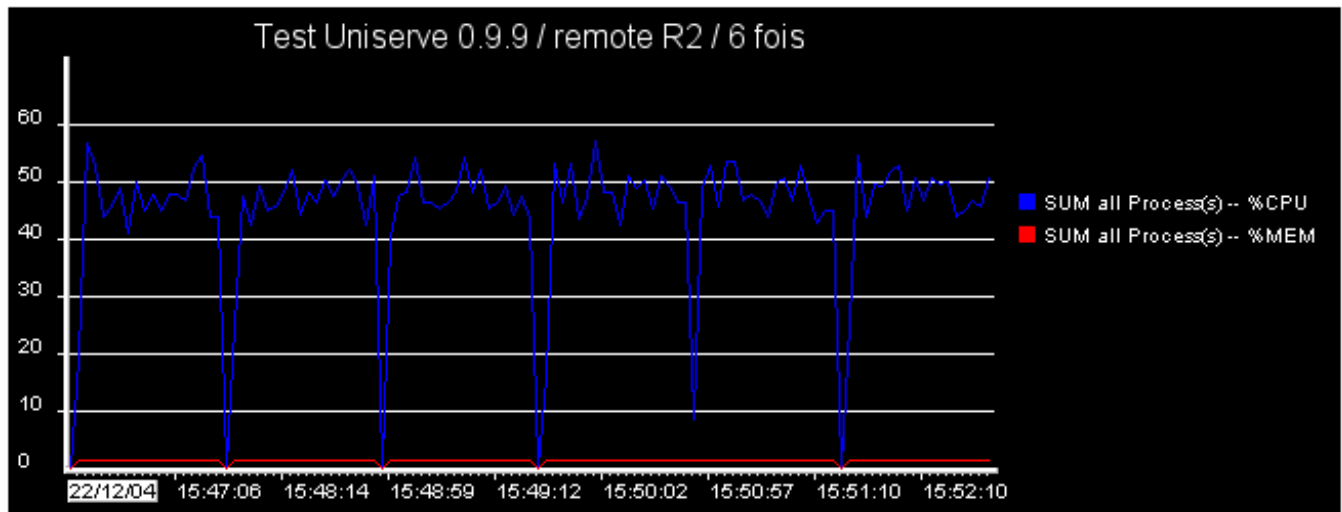
Remarques :

Quoique stable dans sa durée le test, là encore le temps de traitement est sans commune mesure avec celui d'Apache seul ou Uniserve.

3.2 Tests Uniserve R2 avec injecteur BLADE : consommation mémoire et charge réseau

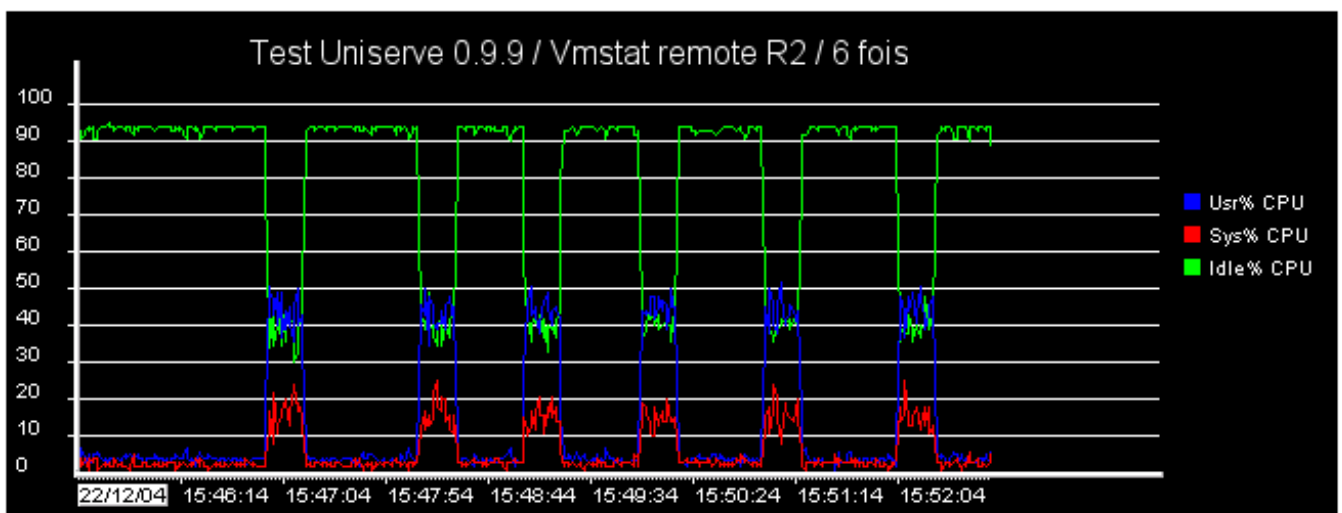
3.2.1 Evolution du process starter.r

Analyse du processus sur starter.r (process uniserve)



⇒ 45 % de CPU pour le processus starter.r, et moins de 5 % de consommation mémoire.

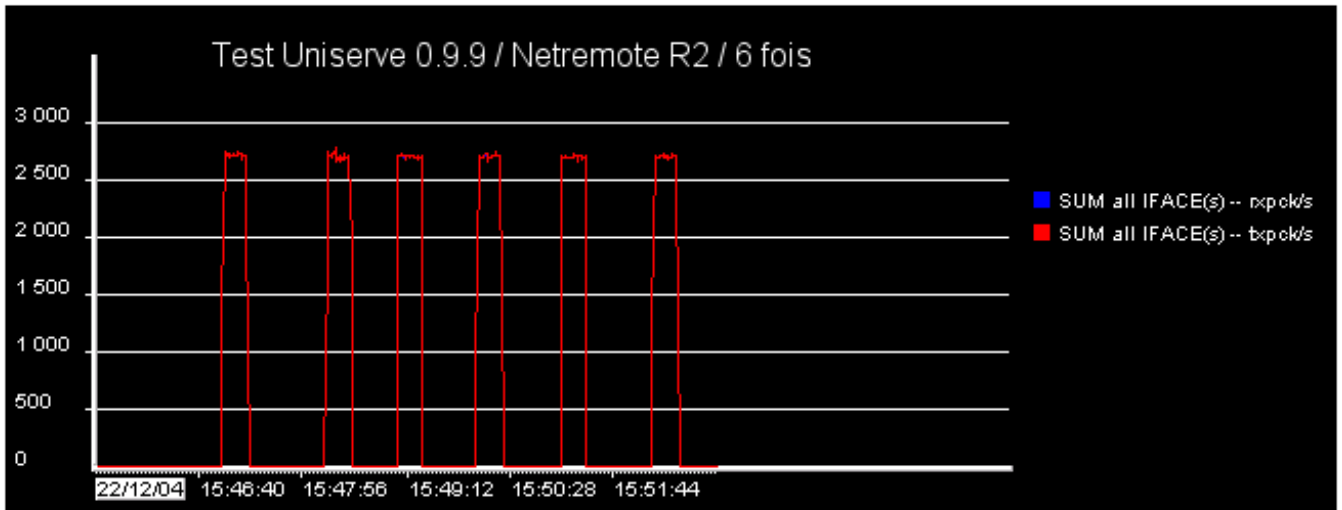
3.2.2 Analyse VmStat :



⇒ On peut remarquer que chaque run entraîne un pic d'activité CPU à 40 - 45 % pour le processus starter.r et 20% pour le système.

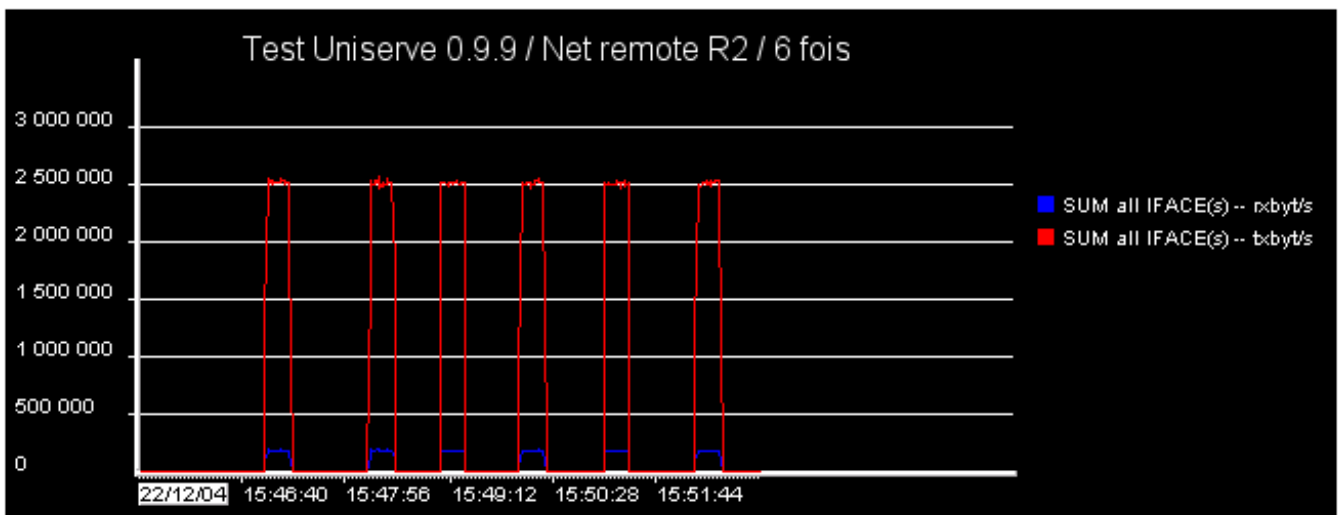
3.2.3 Analyse Réseau :

3.2.3.1 Nombre de paquets transmis :



⇒ environ 2700 – 2750 paquets sont transmis.

3.2.3.2 Nombre d'octets/s :

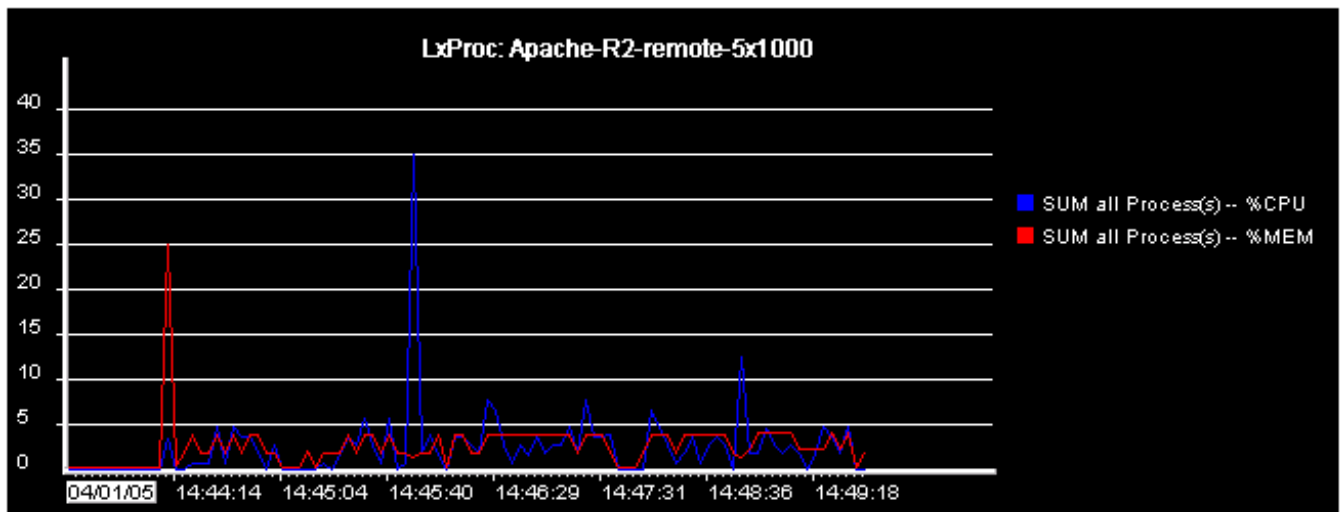


⇒ 2,5 Mo/s sont transmis au maximum, soit 5 % de la bande passante.

3.3 Tests Apache 1.3.27 R2 avec injecteur Blade – consommation mémoire et charge réseau

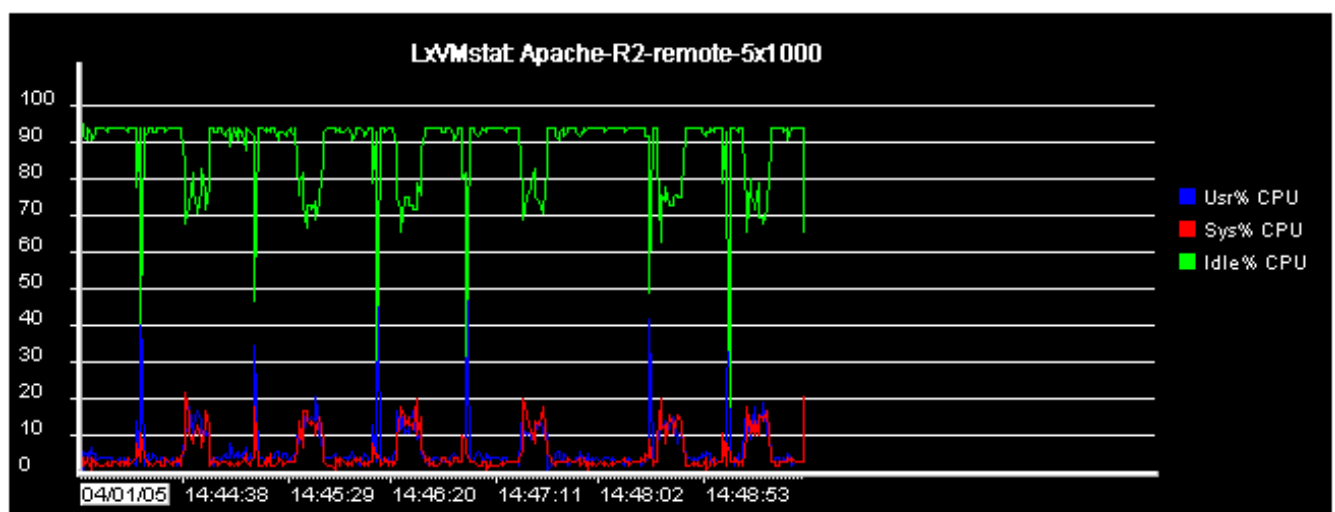
3.3.1 Évolution du process httpd

3.3.1.1 Consommation CPU (6 tirs)



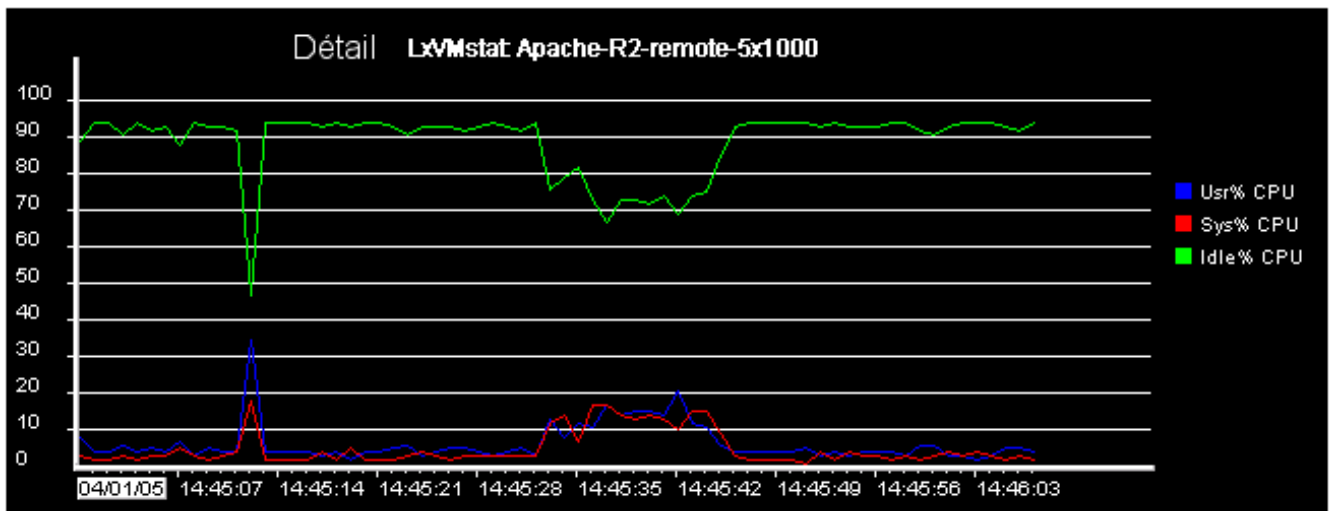
⇒ L'allure des courbes est moins régulière que pour Uniserve. On distingue les enveloppes globales des 5 tirs, avec les phases d'arrêt-relance, mais,

3.3.2 Analyse VMSTAT :



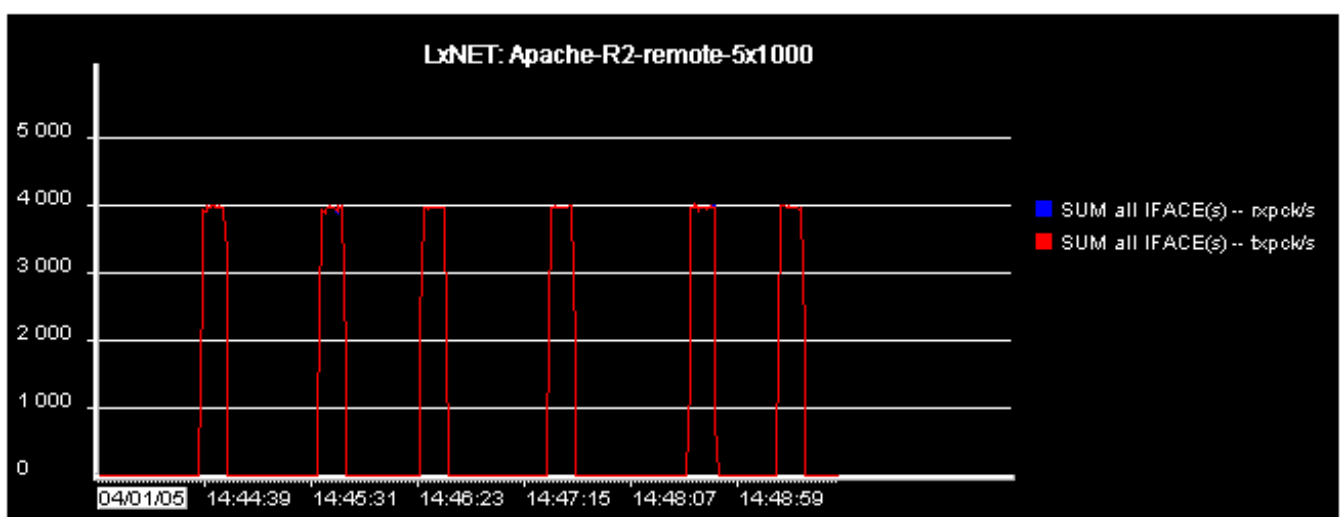
- ⇒ On note que le processus présente une courbe en deux parts, un pic étroit pouvant aller jusqu'à 40 % de CPU (pulse), et un ensemble plus large à une moyenne de 15 % de CPU.
- ⇒ Le pic étroit correspond à la phase d'arrêt-relance du serveur Web entre deux runs. C'est donc le pic large qui correspond au test R2.

3.3.2.1 Détail VMSTAT d'un run :



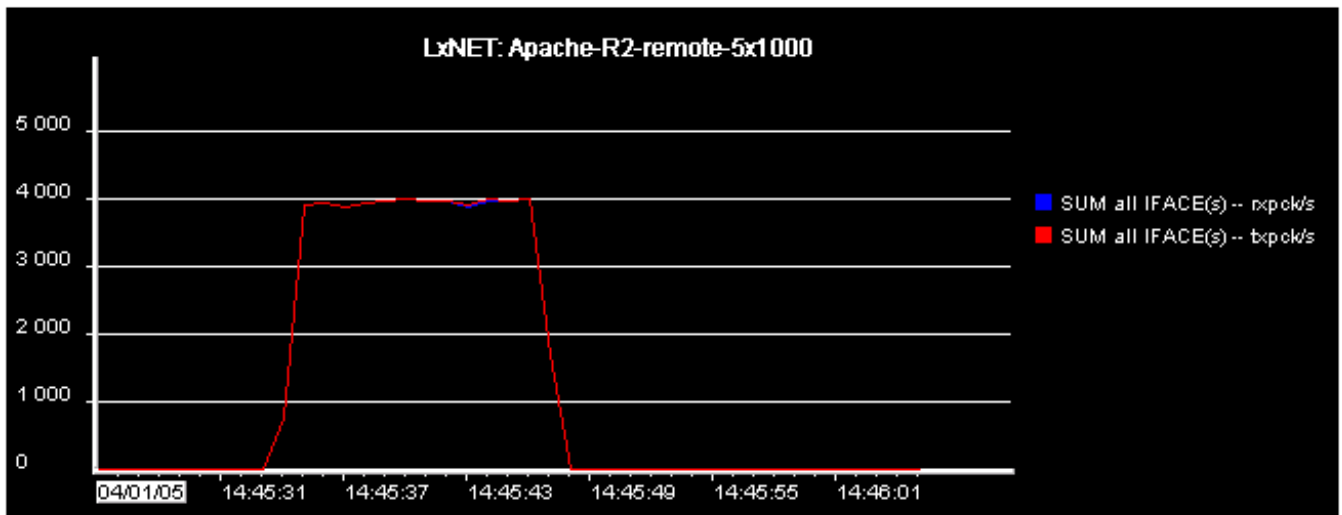
3.3.3 Analyse Réseau :

3.3.3.1 Nombre de paquets transmis :

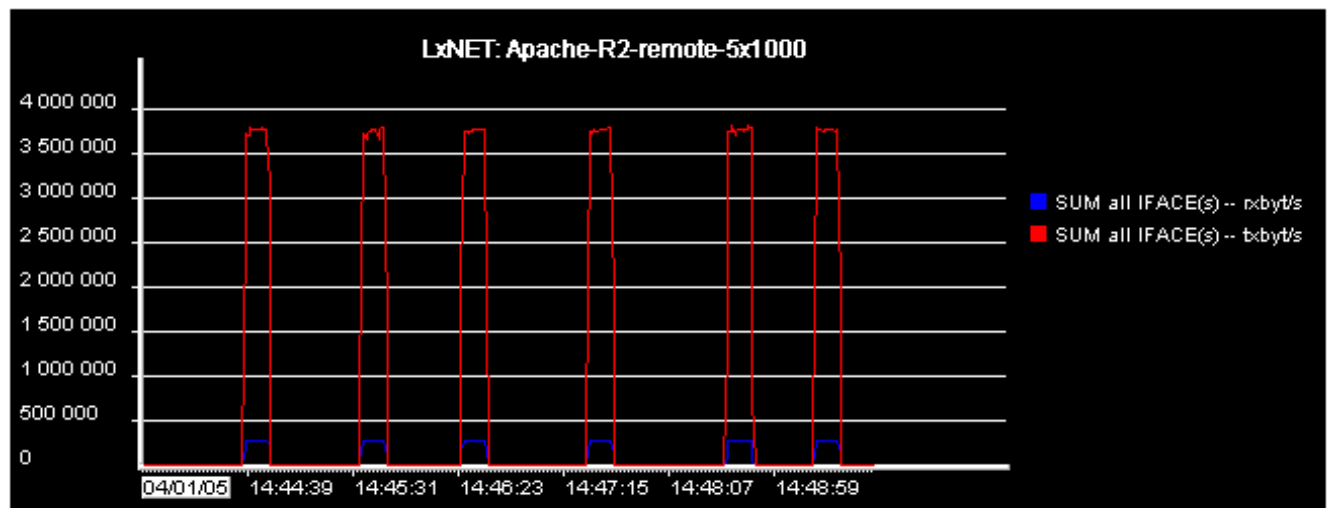


⇒ 50 % de paquets en plus sont transmis par-rapport à la solution Uniserve.

3.3.3.2 Détail d'un pic :



3.3.3.3 Nombre d'octets/ s :



⇒ On observe le même ratio (1.5) entre Apache et Uniserve.

3.4 Synthèse de la partie I

Si on compare les données, une première synthèse se dégage :

- En local (L1, L2, L3), une requête prend environ entre 4 et 5 ms, tandis que pour les test équivalents en remote (R1, R2, R3), le temps moyen est de 67-68 ms avec l'injecteur PC et de 3.5 ms avec l'injecteur Blade. Les meilleurs temps moyens obtenus en Remote (Blade) qu'en local peuvent s'expliquer par les caractéristiques monolithique du noyau Linux : lors d'accès extérieurs, l'adaptateur réseau pré-traite les instructions transmises au kernel.
- Sur les tirs en Remote avec injecteur Blade, il est possible de dresser la classification suivante pour les temps **moyens** :

Type de test	Serveur Apache 1.3.27	Serveur Uniserve/HTTPd	Serveur Apache + Magic!
R1	2.6 ms	3.5 ms	39 ms
R2	2.4 ms	3.5 ms	39 ms
R3	2.0 ms	3.5 ms	39 ms

- Sur les tirs en Remote avec injecteur Blade, il est possible de dresser la classification suivante pour les temps **maximums** :

Type de test	Serveur Apache 1.3.27	Serveur Uniserve/HTTPd
R1	2.6 s	3.5 s
R2 (5000 req.)	12.2 s	17.5 s
R3	2.0 s	3.5 s

⇒ Le serveur Apache est environ 50% plus efficace en moyenne que le serveur Uniserve / HTTPd.

- En termes de ressources machine :

Serveur	%CPU user	%Mem	%CPU sys	txpcks/s	txbits/s
R2 Remote Uniserve	45	2-3	17	2700	2 2500 000
R2 Remote Apache	4	3-4	13	4000	3 700 000

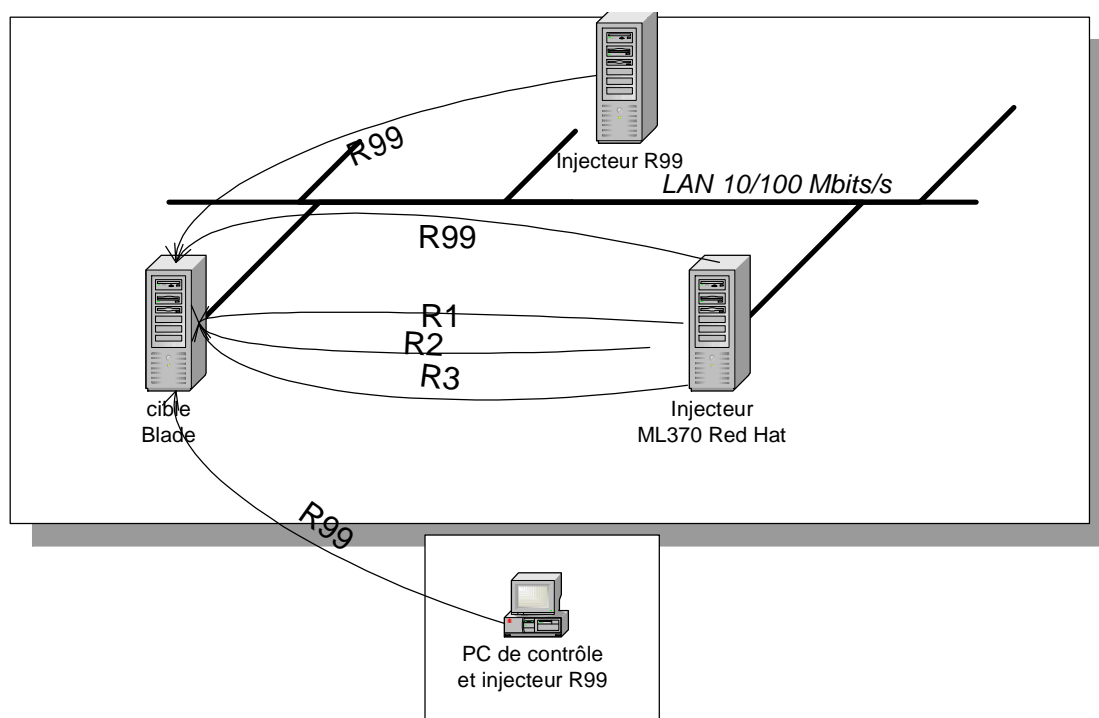
⇒ Là encore, Apache est moins consommateur en ressources que le processus starter.r d'Uniserve. Le nombre de paquets transmis par seconde est supérieur d'environ 50 % à celui pour Uniserve.

4. PARTIE II : INJECTION DEPUIS SERVEUR ML370 VERS LA CIBLE BLADE

4.1 TESTS EN LOCAL (L1, L2, L3) et EN REMOTE sur serveur Blade :

4.1.1 Description des tests :

Les mêmes tests que précédemment ont été réalisés, en utilisant cette fois la machine ML370 comme injecteur,



et la machine Blade HS40 comme cible. Par ailleurs, deux machines supplémentaires ont été utilisées dans le cadre du test de puissance.

Les tableaux suivants donnent une vision synthétique des résultats.

4.1.2 Tests en local :

Uniserve 0.9.9 – R1, R2, et R3 Local sur Blade

Type Test et Run	Temps total (s)	Temps moy. (s)	Temps min (s)	Temps max (s)
L1	2,7087	0.002708	0.0021	0.0060
L2	13,788	0.00276	0.0021	0.0064
L3	2,7539	0.00276	0.0022	0.0060

Remarques :

⇒ Une requête prend en moyenne 2.7 ms ce qui correspond à un gain brut de 66 à 70 %, par-rapport aux tests équivalents faits sur le serveur ML370.

La partie logicielle (rebol, uniserve) étant rigoureusement identique, on peut y voir l'effet du changement de matériel, éventuellement la différence de noyau (ici, 2.6.9-1) pour Linux.

4.1.3 Test en remote

Uniserve 0.9.9 – R1, R2, et R3 sur Blade Remote depuis ML370

Type Test et Run	Temps total (s)	Temps moy. (s)	Temps min (s)	Temps max (s)
R1 – 1	3,76	0,00376	0,00299	0,0118
R1 – 2	3,90	0,00391	0,0029	0,0100
R1 – 3	3,895	0,0039	0,003	0,0115
R1 – 4	3,756	0,00375	0,003	0,0117
R1 – 5	4,01	0,0040	0,003	0,0122
Moyenne	3,864	0,00386	0,003	0,0114

(Rappel résultat sur ML370 : Moyenne 3,5265 0,0035 0,00279 0,01031)

Uniserve 0.9.9 – R2 Remote sur Blade depuis ML370

Type Test et Run	Temps total (s)	Temps moy. (s)	Temps min (s)	Temps max (s)
R2 – 1	20,8315	0,00416	0, 0029	0,0118
R2 – 2	19,6487	0,0039	0,0029	0,0118
R2 – 3	20,5785	0,0041	0,003	0,0117
R2 – 4	20,3631	0,0040	0,003	0,0124
R2 – 5	19,964	0,0039	0,003	0,012
Moyenne	20,277	0,0040	0,00296	0,0119

(rappel résultat R2 sur ML370 : Moyenne 17,46800 0,00349 0,00277 0,01360)

Uniserve 0.9.9 – R3 Remote sur Blade depuis ML370

Type Test et Run	Temps total (s)	Temps moy. (s)	Temps min (s)	Temps max (s)
------------------	-----------------	----------------	---------------	---------------

R3 – 1	4,1553	0,0041	0,003	0,0117
R3 – 2	4,097	0,0041	0,003	0,0116
R3 – 3	4,17	0,0041	0,003	0,0116
R3 – 4	4,287	0,0043	0,003	0,0116
R3 – 5	4,05	0,0040	0,003	0,0099
Moyenne	4,15	0,0041	0,003	0,0113
(rappel résultat R3 sur ML370 : Moyenne 3,51180 0,00350 0,00279 0,01248)				

Remarques :

Les résultats obtenus sont légèrement en deçà (de 0.4 ms en moyenne par requête) de ceux obtenus pour les tests équivalents effectués sur le ML370, alors que les tests en local sont nettement meilleurs. La puissance du ML370 (PIII) étant inférieure à celle du serveur Blade, on peut considérer que l'injecteur ML370 n'est pas assez performant pour la cible.

Pour valider cette hypothèse, un test rapide a été effectué avec une machine de performances proches de celle du serveur Blade :

Uniserve 0.9.9 – R2 Remote sur Blade depuis serveur iso-Blade

Type Test et Run	Temps total (s)	Temps moy. (s)	Temps min (s)	Temps max (s)
R2 – 1	18,7736	0,003756	0,0032	0,0082
R2 – 2	18,989	0,0038	0,0032	0,0081
R2 – 3	18,991	0,0038	0,0032	0,0080
Moyenne	18,92	0,00378	0,0032	0,0081

Dans ce cas, les performances se rapprochent nettement de celles observées lorsque le serveur Blade est injecteur et la cible est le ML370.

⇒ On peut donc raisonnablement penser qu'avec une machine très haut de gamme, on tendrait vers des temps moyens par requêtes de l'ordre de 3.5 ms, soit un équivalent de : 285 requêtes traitées en moyenne par seconde, pour une machine injectrice.

4.2 TESTS DE PUISSANCE :

Une série supplémentaire de tests (nommée R99) a été réalisée, à l'issue des premiers résultats.

Il s'agit de tests en accès concurrentiels : 3 machines vont générer des requêtes concurrentielles sur la machine cible, pour le serveur Uniserve.

4.2.1 Temps de réponse test R99-1 :

L'analyse des temps de réponses doit être mise en parallèle de l'observation des tirs : 2 machines, en full 100 Mb/s dans le même LAN que la cible Blade, envoient leurs requêtes, ainsi qu'un PC situé hors du LAN des trois autres.

L'écart entre le début des tirs pour chaque machine est de l'ordre d'une seconde.

On constate visuellement dans les différentes consoles que les deux premières machines terminent leurs tirs bien avant le PC.

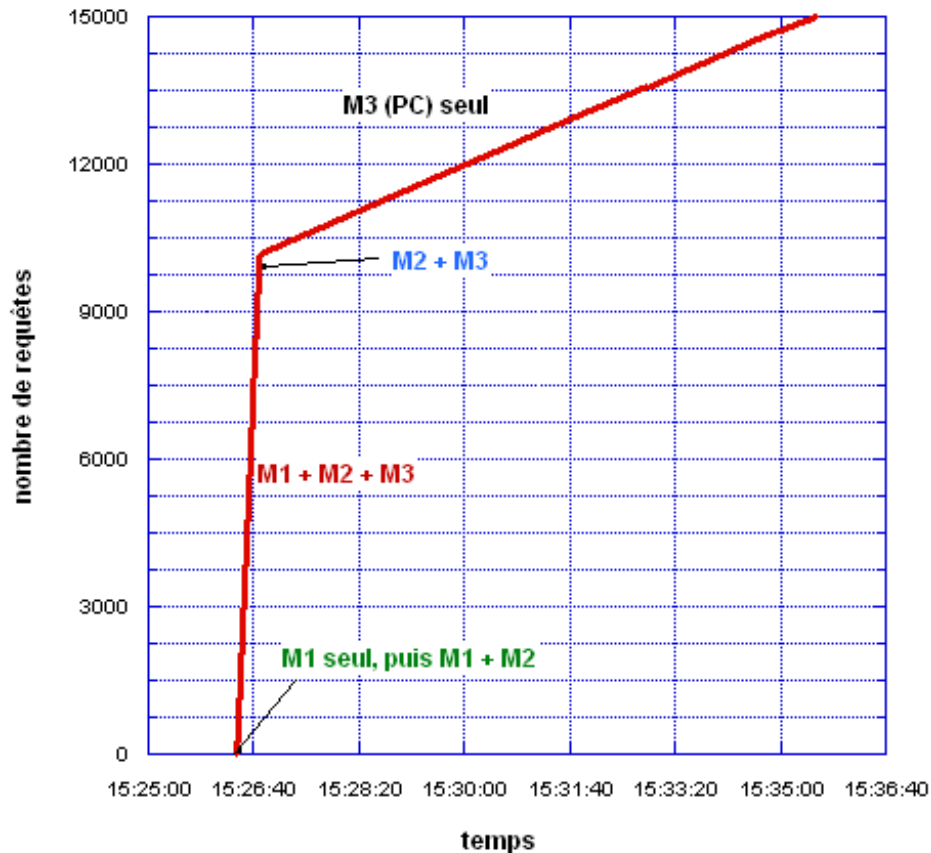
Au niveau des logs Uniserve, la première machine à injecter ses requêtes place environ 260 requêtes avant que la deuxième ne commence, puis les traces s'intercalent régulièrement.

Enfin les traces du PC apparaissent, les deux premières machines terminent leurs injections, tandis que le PC va continuer pendant plusieurs dizaines de secondes.

Le tableau suivant résume les différentes phases :

Phase	Temps total (s)	Temps moy. (ms)	Temps min (ms)	Temps max (s)	Nbre requêtes traitées par phase	Nbre Req/s
<i>R99-I M1 seule</i>	0.99	3.8	1.6	93	260	263
<i>R99-II M1 + M2</i>	1.8	2.1	N/A	63	853	473
<i>R99-III M1 + M2 + M3</i>	19	2.2	N/A	88	8901	468
<i>R99-IV M2 + M3</i>	0.6	2.4	N/A	52	147	245
<i>R99-V M3 seule</i>	528	109	101	335	4839	9,1
	550,4				15000	

R99 sur machine Blade



Remarques :

- è Les temps moyens sont déterminés sur le nombre de points (requêtes) dans la phase considérée.
- è les 15 000 requêtes sont traitées en un peu plus de 9 minutes, à la moyenne globale de 27,3 req/s.
- è Tant que les machines I et II injectent (et même si la machine M3 = PC envoient aussi des requêtes), ce sont elles qui génèrent le plus de trafic. On montrerait que dans la phase V où le PC reste seul injecteur, il a 96% de son injection encore à réaliser. D'où l'importance de la maîtrise du débit réseau et de la puissance de la machine.
- è Durant la phase II, Uniserve traite indifféremment et alternativement les requêtes de M1 et M2.
- è Durant la phase III où les trois machines sont en concurrence, le fait d'avoir une injection plus lente diminue à peine le nombre de requêtes traitées par seconde. En effet, très peu de requêtes provenant du PC hors LAN sont traitées par Uniserve.
- è Les 10 000 requêtes des machines M1 (5000) et M2 (5000) sont traitées en 22.4 : soit un temps moyen de 2.2-2.3 ms, et un gain brut de 50%. Des essais complémentaires doivent être menés.

4.2.2 Temps de réponse test R99-2 avec de 2 à 9 injecteurs :

D'autres tests de puissance ont été menés sur le principe précédent.

Le serveur Uniserve a été déployé sur chaque injecteur (3 précédents) et un service créé, dupliqué et customisé pour déclencher sur l'événement `on-new-client` une injection de 5000 requêtes aléatoires sur la cible Blade HS40.

```
install-service [  
  name: 'runtime1965  
  port-id: 1965  
  scheme: 'tcp  
  stop-at: none  
  
  on-new-client: does [  
    loop 5000 [read to-url rejoin ["http://IP-CIBLE/index" random 5 ".html"]]  
      write-client "OK"  
      close-client  
  ]  
]
```

Un service va écouter sur le port 1965, un autre sur le port 1966, etc jusqu'à 1968.

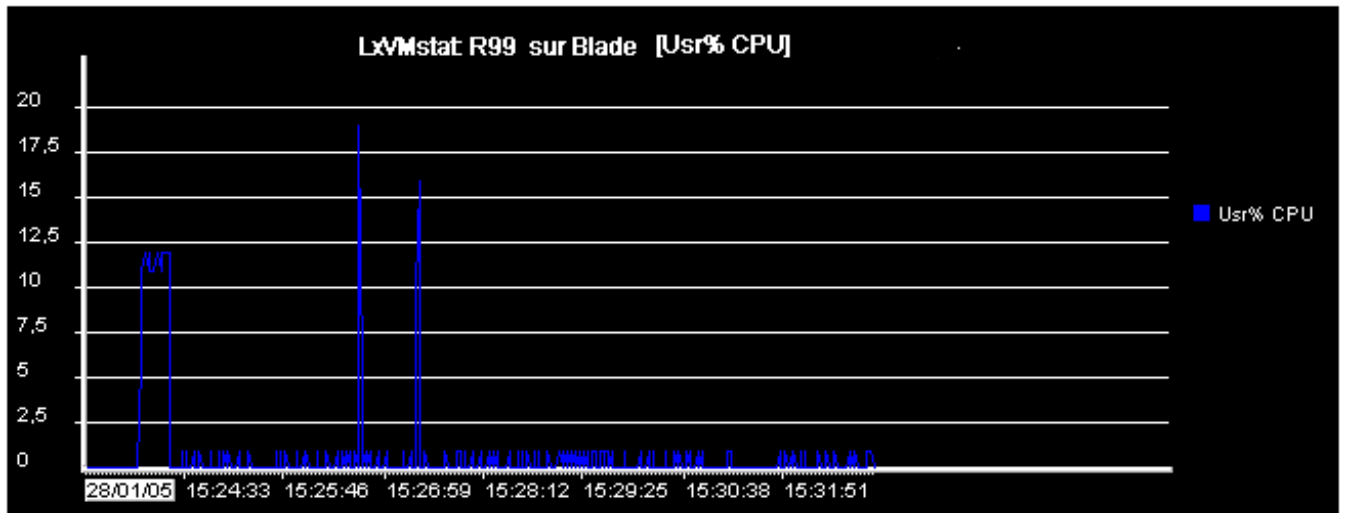
Le déclenchement des événements **on-new-client**, et donc des 3 injections, est réalisé par l'appel (via un PC externe) d'une requête tcp sur chaque service d'Uniserve par serveur (en utilisant une commande : **close open** sur le port).

Le tableau suivant résume les résultats :

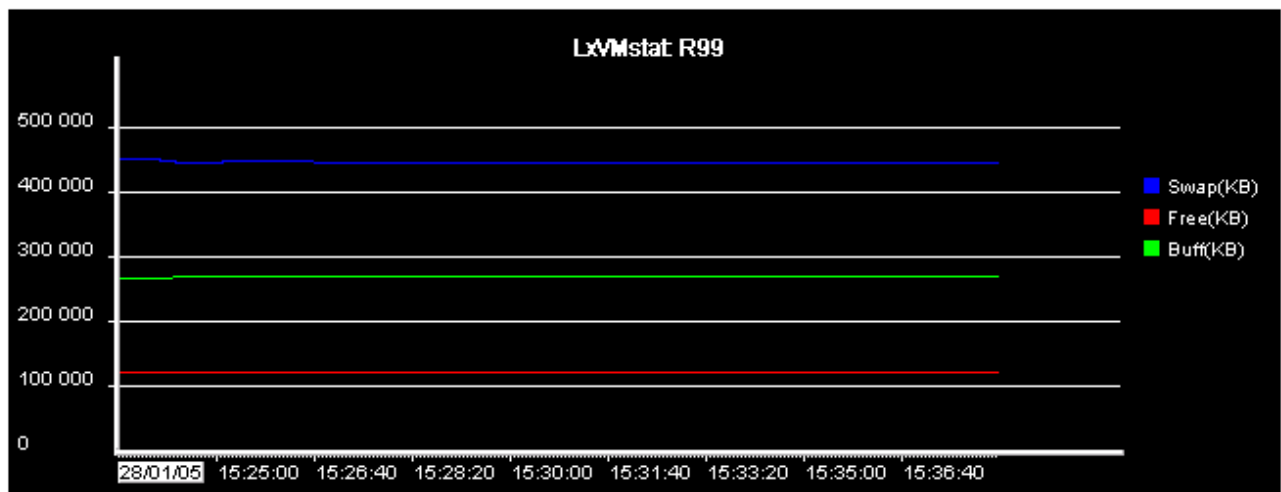
N° injecteur	Test	Temps total (s)	Temps moy. (ms)	Temps min (ms)	Temps max (ms)	Nbre requêtes traitées	Nbre moy. Req/s
1	-	15.8	3.15	2.59	7.59	5000	317.5
2	R99	19.37	1.94	0.8	10.7	10000	515.4
3	R99 - 3	23.8	1.58	0.87	49.6	15000	630
4	R99 - 4	38.8	1.94	0.87	6.8	20 000	515.5
5	R99 - 4	42.9	1.79	0.87	19.8	25 000	558.6
6	R99 - 6	47.2	1.57	0.87	9.4	30 000	636.9
9	R99 - 9	71.6	1.59	0.87	21.9	45 000	628.9
12	R99-12	94.8	1.58	0.87	22.8	60 000	633

è Les logs Uniserve sur la cible Blade montre que le compteur ne s'incrémente pas régulièrement au début des logs pendant quelques secondes.

4.2.3 Analyse VMStat du processus starter.r



On remarque que la consommation mémoire n'est pas correcte. A part le massif initial et les deux pics à 12.5 et 15 % de consommation CPU, la mesure reste en-dessous des 5 %. Mesure non probante.



4.3 Synthèse de la partie II

Si on compare les données, une première synthèse se dégage :

- En local (L1, L2, L3), une requête prend environ entre 2.7 ms, tandis que pour les tests équivalents en remote (R1, R2, R3), le temps moyen est de 4.0 ms avec injecteur ML370, et 3.7 ms avec un injecteur plus puissant.
- Sur le tir de puissance (R99) où trois machines envoient 5000 requêtes sur le serveur cible Blade, Uniserve se comporte très bien : il absorbe ainsi les 2/3 de la charge en un peu plus de 22 secondes, et atteint donc les performances de l'Apache 1.3.27 en configuration un pour un.

Type de test	Serveur Apache 1.3.27 1 pour 1	Serveur Uniserve/HTTPd 1 pour 1	Serveur Uniserve/HTTPd 2 pour 1
R1	2.6 ms	3.8 ms	
R2	2.4 ms	4.0 ms	
R3	2.0 ms	4.1 ms	
R99			2.2 ms (sur partie du test)

- En termes de ressources machine : il n'a pas été possible d'effectuer des mesures probantes avec les outils utilisés auparavant (partie I).

5. ANALYSE DES RESULTATS :

5.1 Comparaison Uniserve ML370 versus Blade en local :

Sur les temps de réponse des tirs L1, L2, L3, le gain est de l'ordre de 70% : on passe en changeant de machines de 4.2-4.9 ms à 2.7 ms. Les configurations logicielles (Rebol, Uniserve) étant les mêmes, le gain s'explique par les différentes performances des machines et des noyaux. Le kernel 2.6.9-1 du serveur Blade a en effet été optimisé à la compilation pour permettre une meilleure gestion de la couche TCP/IP (paramètre).

5.2 Comparaison Uniserve ML370 versus Blade en Remote :

En situation un pour un (une cible, un injecteur), les temps de réponse varient fortement selon la puissance de la machine injectrice, mais aussi du réseau.

Pour le ML370, le temps de réponses moyen varie entre 3.5 ms et 68 ms suivant la nature de l'injecteur et sa situation (LAN 100/hors LAN et 10 Mbs)

Test	ML370/inj PC	ML370 / inj. Blade	Blade /inj ML370	Blade / inj M2
R1	67,9	3.5	3.8	
R2	66,4	3.5	4.0	3.78
R3	68,2	3.5	4.1	

Au sein du même LAN, seule la puissance de la machine injectrice conduit à une différence, et encore minime.

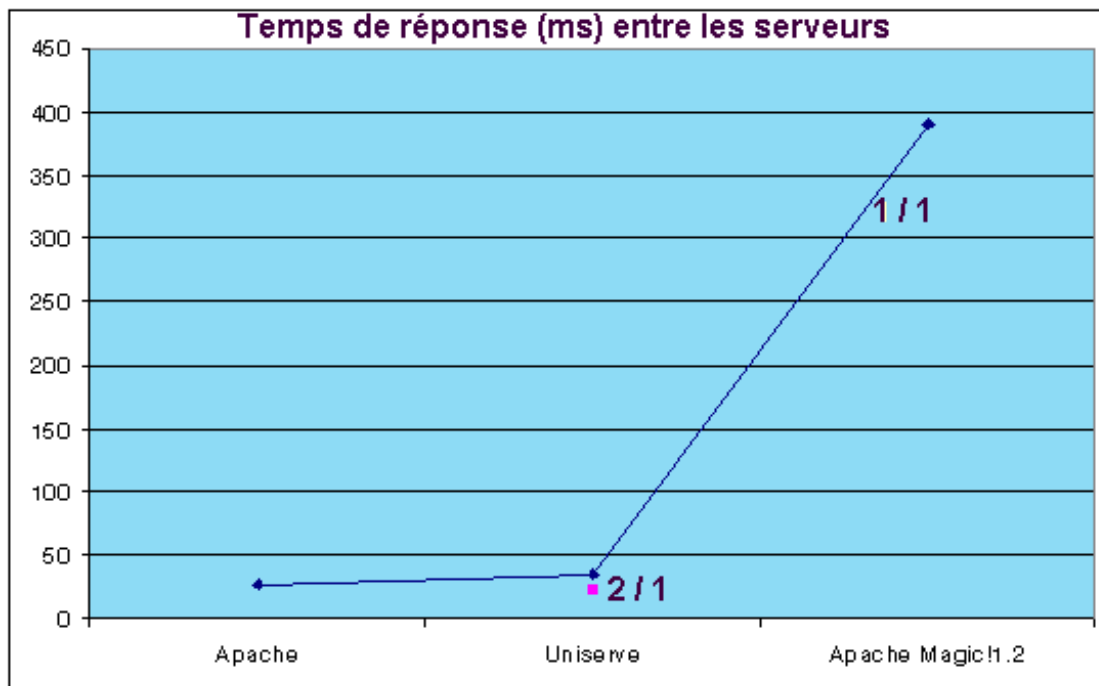
5.3 Tests de puissance :

On montre qu'en accès concurrentiels, le temps moyen de traitement d'une requête se situe entre 1.5 et 1.9 ms.

Uniserve traite les requêtes entrantes de manière séquentielle, de sorte que le temps total suit une loi linéaire avec le nombre de requêtes.

5.4 Comparaison Apache / Uniserve :

Le schéma suivant montre les différents temps de réponse :



Les différences entre les serveurs peuvent s'expliquer ainsi :

- Apache est compilé et génère au démarrage un certain nombre de processus fils, 8 dans notre cas, chacun d'entre eux ayant la charge de traiter une requête unique. Chacun des processus fils peut cependant être amené à servir successivement plusieurs requêtes pendant sa durée de vie. Potentiellement, cette distribution de processus permet au serveur Apache de répondre à de fortes sollicitations. (pre-forking model server).
- Uniserve 0.9.9 est servi par l'interpréteur REBOL, et une fois rentré dans sa boucle événementielle attend les requêtes pour les traiter. La gestion des ressources et notamment de la mémoire est assurée par l'interpréteur REBOL et par l'OS. Ce fonctionnement s'appuie donc fortement sur les performances intrinsèques de la machine, et de la VM Rebol (partie réseau (fonction awake)).
- Magic!1.2 traite les requêtes effectuées que lui passe le serveur Apache. A chaque demande, il y a un temps de chargement de Rebol, significatif, qui pénalise les performances du serveur.

Conclusion :

Cette étude permet de donner des éléments qualitatifs et des métriques concernant l'usage d'un serveur multi-protocole tel que Uniserve de SoftInnov, relativement à Apache.

Environ 300 000 requêtes ont été lancées au total entre la phase de préparation et les campagnes de tirs.

Il sera sans aucun doute nécessaire de procéder à une évaluation complémentaire mettant en compétition une version plus évoluée d'Uniserve, voire le serveur Cheyenne, et une version 2.x d'Apache, sur une machine de performances équivalentes au serveur Blade HS40 utilisé dans cette étude.

ANNEXES

6. ANNEXE 1 :

Comparatif de versions pour REBOL

Ici le comparatif porte sur le serveur Blade

Uniserve 0.9.9 – Tests en Local avec version Rebol : REBOL/Core 2.5.6.4.2

Type Test et Run	Temps total (s)	Temps moy. (s)
L1	2.71	0.0027
L2 (5000 requêtes)	13.8	0.00275
L3	2.76	0.00276

Uniserve 0.9.9 – Tests en Local avec version Rebol/core 2.5.58.4.2 (alpha)

Type Test et Run	Temps total (s)	Temps moy. (s)
L1	2.806	0.0028
L2 (5000 requêtes)	14.10	0.00282
L3	2.78	0,00278

è il n'y a pas de différences perceptibles entre les deux versions de REBOL, ce qui est normal dans la mesure où c'est dans la phase de chargement de la VM Rebol que peut exister une différence de temps. Une fois chargée en mémoire, la VM se comporte identique quelque soit sa version (du moins ici).