



# Livre blanc

## Haute disponibilité sous Linux

Nicolas Ferre <Nicolas.Ferre@alcove.fr>

29 septembre 2000

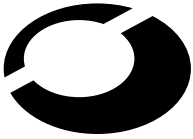
### Résumé

Ce livre blanc décrit une solution informatique à haute disponibilité. Les technologies mises en oeuvre permettent d'augmenter la fiabilité d'un système informatique de type Linux : elles maintiennent en permanence au moins une machine opérationnelle.

*Avec les livres blancs d'Alcôve, bénéficiez de l'expérience de la première société européenne d'expertise sur les logiciels libres.*

### Copyright

Alcôve, tous droits réservés.



## Table des matières

<b>1</b>	<b>Les enjeux de la haute disponibilité</b>	<b>1</b>
1.1	Sécuriser le fonctionnement de l'entreprise . . . . .	1
1.2	Définition du besoin . . . . .	2
<b>2</b>	<b>Solutions pour une haute disponibilité</b>	<b>3</b>
2.1	Ressources critiques d'un système informatique . . . . .	3
2.2	Outils et matériels . . . . .	4
2.2.1	Surveillance et répartition de charge . . . . .	4
2.2.2	Mécanismes de redondance . . . . .	5
2.2.3	Tolérance aux pannes . . . . .	6
<b>A</b>	<b>Références</b>	<b>9</b>



# 1 Les enjeux de la haute disponibilité

## 1.1 Sécuriser le fonctionnement de l'entreprise

Les ordinateurs composent le système nerveux de l'entreprise. Ils sont indispensables à son bon fonctionnement 24 heures sur 24 car des clients, collaborateurs, commerciaux ont besoin en permanence de se connecter au système d'information. Habités aux services que peut rendre le système d'information interne, les employés ont besoin d'une bonne disponibilité de leur outil de travail. De même, les clients utilisent régulièrement le portail commerce électronique de l'entreprise présent sur la toile...

Donc une panne peut causer une perte de productivité considérable et coûter beaucoup d'argent.

Si le système informatique est chargé de contrôler un accès aux bâtiments, de vérifier le bon fonctionnement de processus industriels ou tout autre tâche critique, la haute disponibilité est ici indispensable pour des questions de sécurité.

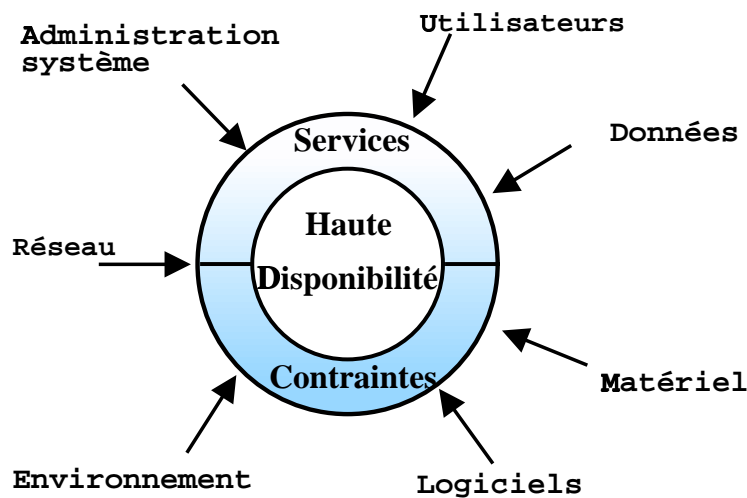


FIG. 1.1 – Eléments pouvant altérer la disponibilité d'un système informatique.

Bien que les logiciels libres soient très implantés dans le monde du service Internet pour les entreprises (Apache est le premier serveur web de l'Internet), ils se développent aussi dans d'autres domaines : serveurs d'applications, interrogation de bases de données, réseau privé virtuel (VPN<sup>1</sup>), machine de contrôle de processus industriels, sécurité électronique des bâtiments, etc.

Les logiciels libres doivent donc proposer des systèmes hautes disponibilités de qualité comparable à ceux des éditeurs propriétaires.

<sup>1</sup>VPN (Virtual Private Network) : Réseau privé qui utilise les infrastructures publiques de communication tout en maintenant la confidentialité des données. Une entreprise peut utiliser cette technique pour construire un réseau intranet entre différents sites distants, sans utiliser de liaisons spécialisées.



## 1.2 Définition du besoin

Une configuration haute disponibilité est fortement dépendante du besoin de l'entreprise : de la distribution du travail entre plusieurs machines à la duplication permanente des données dans des bâtiments géographiquement séparés, la solution technologique, la mise en oeuvre et le coût sont différents.

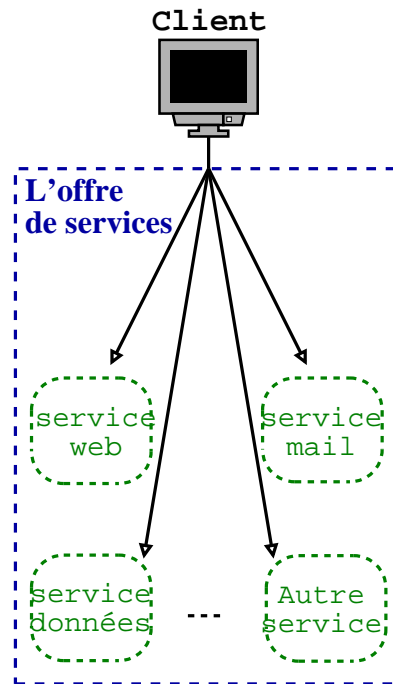


FIG. 1.2 – Services exigeant une haute disponibilité.

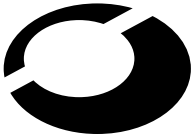
Une grappe d'ordinateurs (plus couramment appelée cluster<sup>2</sup>) est employée pour fournir des services différents :

- la construction d'une machine de calcul parallèle (cluster scientifiques) ;
- la mise en place d'un système haute disponibilité ;
- la répartition de charge entre plusieurs machines (load balancing).

Il est fréquent de demander à un système de proposer une répartition de la charge de travail et, en cas de panne, d'avoir un comportement haute disponibilité. Ces deux derniers points seront donc souvent associés pour garantir une qualité de service optimale.

En revanche, notre propos n'est pas ici d'augmenter la puissance de calcul mais de sécuriser le système : la mise en place d'un agrégat de machines parallèles, dédiées au calcul, n'est pas l'objet de ce livre blanc.

<sup>2</sup>Cluster : Ordinateurs en grappe qui se partagent le travail et/ou peuvent prendre le relais les uns des autres. Une des ces machines constitue un *noeud* du cluster.



## 2 Solutions pour une haute disponibilité

### 2.1 Ressources critiques d'un système informatique

Identifier les faiblesses d'un système informatique est l'action initiale à mener pour pouvoir proposer un fonctionnement fiable. La solidité du système d'exploitation et des applications a souvent fait l'objet d'une étude approfondie, avant le choix d'implémentation d'une solution d'entreprise. Le système GNU/Linux<sup>1 2</sup> et les applicatifs implantés au coeur d'une société ont assurément déjà fait leurs preuves. Ils ne sont que très rarement en cause lors d'un problème informatique.

Les causes d'un dysfonctionnement logiciel viennent essentiellement d'une charge trop importante de travail. Il convient dans ce cas de supprimer le «goulet d'étranglement» en employant les méthodes de répartition de charge.

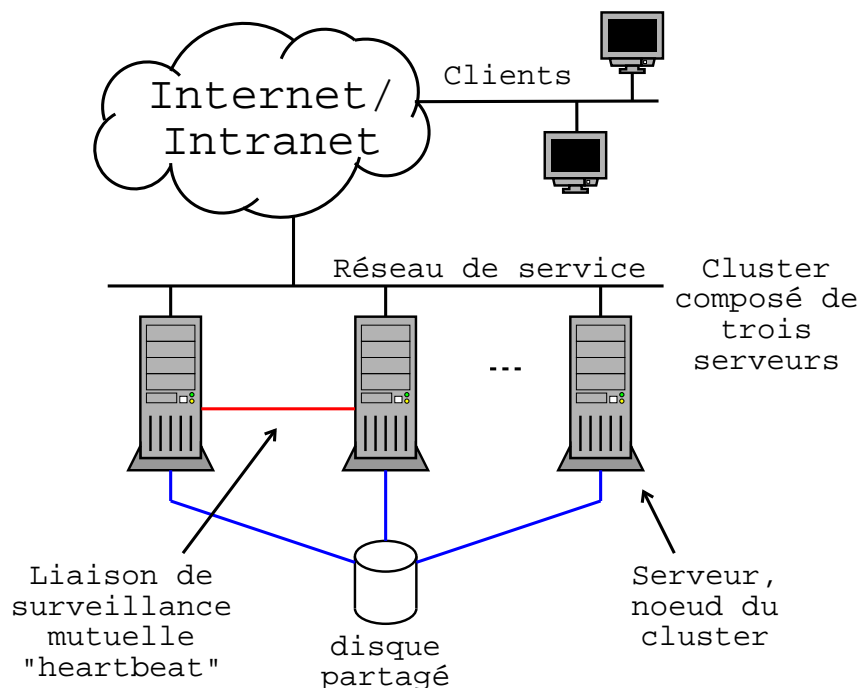
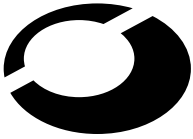


FIG. 2.1 – Terminologie.

En revanche, le matériel nous met en face d'une toute autre problématique : un composant physique du système peut tomber en panne. La chute d'une seule de ces ressources critiques (Single Point Of Failure) met

<sup>1</sup>GNU (GNU's Not Unix (GNU N'est pas Unix)) : nom du projet initié par Richard Stallman en 1984 qui consiste à reprogrammer un système compatible Unix sous une licence qui en permet la libre distribution (GPL).

<sup>2</sup>GPL (General Public License) : licence d'utilisation des logiciels du projet GNU qui permet entre autre la libre distribution, et impose que le code source des binaires rendus publics doit être accessible.



hors service la totalité du système informatique :

- processeur ;
- carte mère ;
- alimentation ;
- interface réseau ;
- disque dur, contrôleur et câble des systèmes de stockage de données .

En faisant fonctionner plusieurs de ces éléments au sein d'un même système informatique, on supprime leur caractère critique. L'intérêt de la redondance apparaît alors pour éviter l'arrêt total du service. On peut, soit multiplier de tels composants à l'intérieur d'un seul boîtier ou rack, soit former un ensemble de machines classiques communiquant entre elles. La seconde solution permet d'utiliser un matériel standard donc bon marché et aisément disponible ; elle permet aussi de modifier une installation existante.

Si le service peut être rendu (simultanément ou non) par plusieurs machines, le problème d'arrêt du système ne se pose plus, pour des raisons de réparation, mise à jour ou maintenance. Les autres machines du cluster prennent le relais en effectuant une commutation ou une nouvelle répartition de la charge de travail. La synchronisation des données stockées doit alors être prise en charge avec soin, c'est un point délicat à résoudre lors de la construction de clusters.

## 2.2 Outils et matériels

C'est sur l'architecture matérielle que peut se jouer l'efficacité mais aussi le coût de la solution haute disponibilité sous GNU/Linux. Il est important de choisir une configuration de son système adaptée aux besoins.

La combinaison des outils provenant de divers projets libres permet de répondre au cas par cas à chacune des exigences de l'entreprise.

On peut généralement classer les outils et les architectures suivant trois grands axes : surveillance et répartition de charge, mécanismes de redondance et tolérance aux pannes.

### 2.2.1 Surveillance et répartition de charge

Ce sont des clusters employés généralement dans le monde des services Internet et du commerce électronique. Ils permettent de répartir la charge de travail entre les différentes machines. Cette charge peut être celle engendrée par l'exécution d'une application ou par un important trafic réseau. Un tel système est souvent mis en place lorsqu'un nombre important d'utilisateurs demande le même type de service au système. Une requête peut alors être distribuée au noeud du cluster le moins occupé à un moment donné. Certains systèmes permettent même de réaffecter dynamiquement, à un autre noeud, une demande en cours de traitement.

Souvent, les serveurs d'applications réseaux doivent faire face à de nombreuses connections simultanées. Ceci les empêche de répondre assez rapidement pour offrir le service attendu par l'utilisateur, le client. Le trafic est alors dérivé vers un noeud proposant un service équivalent, dans le cluster. L'ensemble est souvent géré par un ordinateur dédié ou par un programme particulier présent sur toutes les machines ; cette gestion est configurable par l'administrateur. La plupart des solutions fonctionnent sur un cluster composé de machines présentes sur un réseau local mais certaines solutions, moins performantes, peuvent emprunter des liaisons distantes.



Les outils disponibles dans le monde du logiciel libre prennent en charge cette répartition à différents niveaux :

- noyau du système d'exploitation (MOSIX<sup>3</sup>, LVS<sup>4</sup>) ;
- espace utilisateur (Mon<sup>5</sup>, GNUQueue) ;
- facilités proposées par certaines applications (Apache, Sendmail).

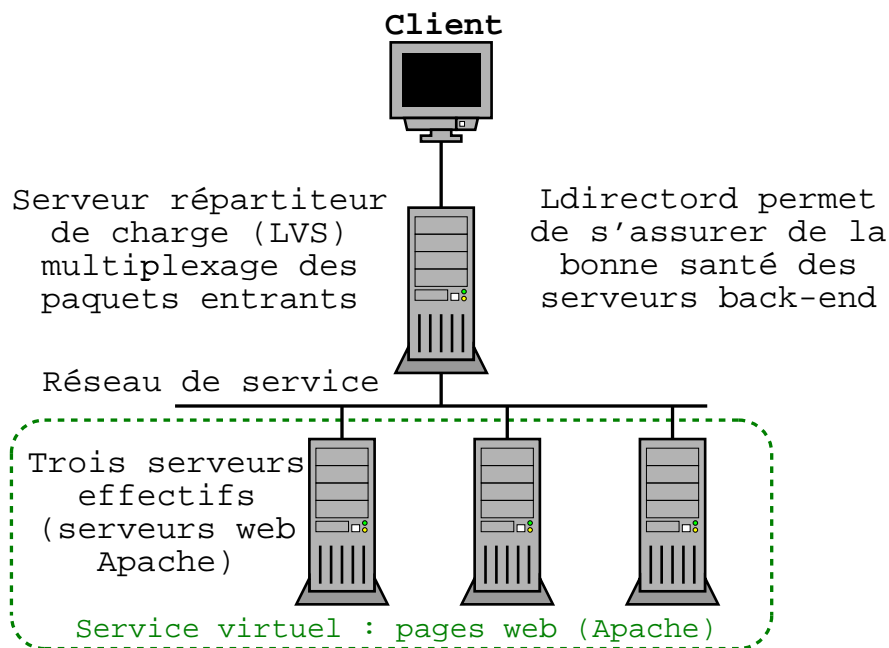


FIG. 2.2 – Exemple de répartition de charge entre plusieurs serveurs web.

Exemple de configuration permettant de surveiller le trafic réseau et de répartir la charge de travail entre plusieurs serveurs effectifs offrant le même service virtuel : pages web servies par **Apache**. **Ldirectord** est un outil permettant de surveiller la bonne santé des serveurs du back-end. Une autre solution aurait été d'utiliser le superviseur de services **Mon**. **Linux Virtual Server (LVS)** répartit les requêtes entre les différents serveurs web.

### 2.2.2 Mécanismes de redondance

La mise en place de plusieurs occurrences d'un composant critique du système permet de supprimer les pannes fatales. On emploie souvent le terme de redondance lorsque les applications critiques et le matériel qui les exécute sont instanciés plusieurs fois et prennent le relais les uns des autres (on parle en anglais de

<sup>3</sup>**MOSIX** : Ensemble logiciel permettant au noyau Linux d'avoir un comportement coopératif dans un cluster ; chaque machine est alors une partie d'un seul système. La répartition de la charge de travail est dynamique, optimisée et totalement transparente pour l'utilisateur.

<sup>4</sup>**LVS** (Linux Virtual Server) ou **IPVS** (Internet Protocol Virtual Server) : Applicatif permettant le multiplexage des paquets IP destinés à des serveurs. Ce dispositif permet de faire de la répartition de charge.

<sup>5</sup>**Mon** : Contrôleur de ressources système qui peut être utilisé pour surveiller la disponibilité d'un service réseau, de conditions de fonctionnement (température), etc. Si un évènement inattendu survient, une alerte est déclenchée.



Failover services : FOS). Le cluster se résume alors à une ou plusieurs paires de machines, principales et de secours.

A l'aide d'un système de communication, chaque ordinateur surveille son ou ses «jumeaux» par l'intermédiaire d'un canal dédié. Il peut se présenter sous la forme d'un lien série (avec ou sans protocole PPP <sup>6</sup>), d'un lien Ethernet ou simplement d'une liaison spécialisée (watchdog <sup>7</sup>). Cette surveillance rapprochée est la garantie de vie d'une machine paire (pouls ou heartbeat). Si un noeud du cluster est amené à tomber, son second prend le relais dans la seconde (ou même plus rapidement), il s'approprie son identité et se charge d'apporter le service demandé par l'utilisateur, sans laisser transparaître la faiblesse passagère du serveur maître. Il s'agit ici de minimiser le temps de commutation. L'ordinateur ayant subi la panne est réinitialisé ou réparé pour reprendre, au plus vite, la surveillance attentive d'un «jumeau» en service.

On peut donc aisément comprendre que le temps de redémarrage d'un serveur doit être assez court et qu'une procédure de récupération des données doit être exécutée.

Dans une redondance efficace, le partage des données stockées sur disque (s'il y en a) est un point à considérer attentivement lors du développement de l'architecture matérielle.

Deux solutions permettent de maintenir l'unicité (intégrité) des données tout en permettant de les préserver si un incident survient :

- Partage du périphérique de stockage (RAID <sup>8</sup>, bus SCSI <sup>9</sup> partagé pour une solution à moindre coût) ;
- Mise en reflet (mirroring) des disques, c'est à dire copie de leur contenu à intervalles réguliers.

L'exemple concret représente une paire de serveurs de mail (utilisant **Sendmail**) redondants. Le client s'adresse en temps normal à la machine principale. Si cette dernière tombe en panne, son pouls (heartbeat) cesse et la machine de secours s'en aperçoit. En effet, le processus **Heartbeat** <sup>10</sup> de la machine de sauvegarde interroge, à travers la liaison série, son processus pair. Ne répondant pas, l'adresse de la machine principale est attribuée par Heartbeat à la machine de secours via le processus **Fake** <sup>11</sup>, elle peut ainsi prendre le relais. Les données sont partagées sur un disque NFS <sup>12</sup> monté alternativement par l'une puis l'autre des machines.

### 2.2.3 Tolérance aux pannes

La tolérance à la panne est une notion beaucoup plus exigeante car elle implique un fonctionnement normal, quelle que soit la nature de la panne.

<sup>6</sup>**PPP** (Point-to-Point Protocol) : Protocole de communication entre deux équipements utilisant une ligne série. C'est le protocole utilisé, par exemple, entre un utilisateur et son fournisseur d'accès à Internet. Dans un contexte de machines redondantes, il peut être utile pour mettre en place un mécanisme de surveillance mutuelle sur un support RS-232.

<sup>7</sup>**Watchdog** : Chien de garde logiciel ou matériel du système qui donne l'alerte si on ne le caresse pas assez souvent ! Typiquement, une panne matérielle ou logicielle empêche de réarmer un compteur qui, arrivé à échéance, déclenche une action de secours.

<sup>8</sup>**RAID** (Redundant Array of Independent Disks) : Moyen de stocker des données à différents endroits sur plusieurs disques durs. L'ensemble apparaît au système sous la forme d'un seul périphérique de stockage. Selon la configuration (RAID-1, RAID-5), la vitesse de lecture, la tolérance aux pannes, la correction d'erreurs peut être améliorée.

<sup>9</sup>**SCSI** (Small Computer System Interface) : Standard décrivant une interface parallèle permettant aux ordinateurs de communiquer avec leurs périphériques (de stockage notamment).

<sup>10</sup>**Heartbeat** : Application permettant à un ordinateur de prendre le pouls (heartbeat) d'autres machines. Si l'une d'entre elles ne répond pas à un message envoyé, elle est considérée comme défaillante ; une mesure de secours est alors prise.

<sup>11</sup>**Fake** : Commutateur de serveurs redondants, Fake permet, à un système de secours, de prendre l'adresse IP d'une machine tombée en panne dans le réseau local.

<sup>12</sup>**NFS** (Network File System) : Application client / serveur permettant d'accéder au périphérique de stockage d'un ordinateur distant. Le système de fichiers ainsi *monté* sur sa machine est vu comme n'importe quel disque local.

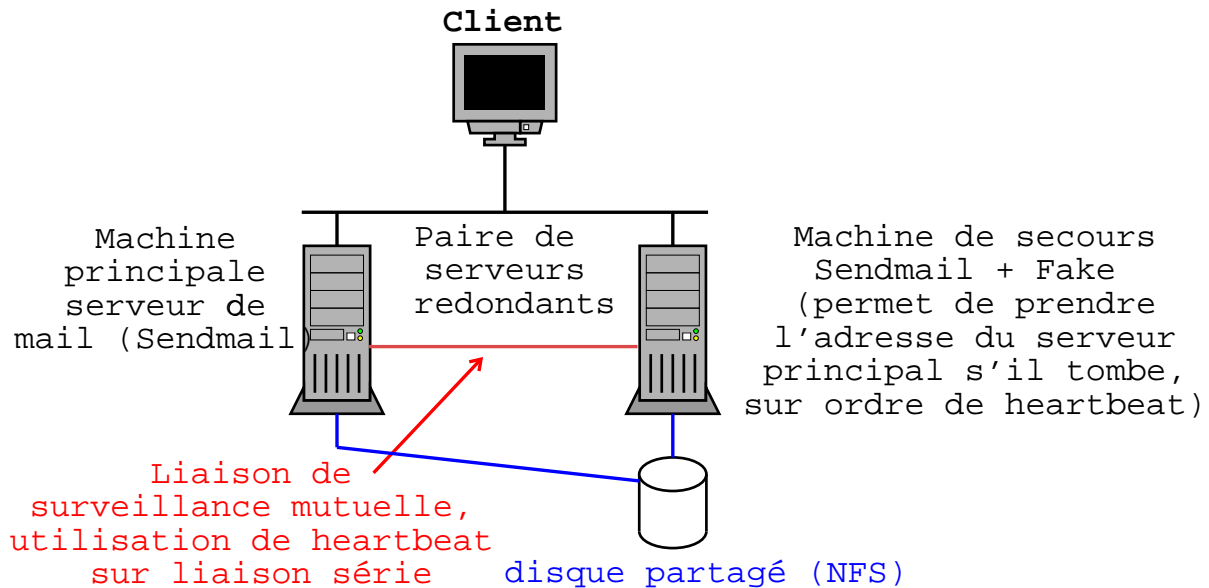


FIG. 2.3 – Exemple de mise en oeuvre de la redondance entre plusieurs serveurs de mail.

Les systèmes tolérants aux pannes sont typiquement employés dans des applications critiques (transport, aérospatiale, etc. ..). Certaines des techniques employées en haute disponibilité proviennent ces environnements.

Il faut tout de même garder à l'esprit que le coût matériel d'un cluster haute disponibilité à base de matériel standard est beaucoup moins cher qu'un serveur tolérant aux pannes. De tels systèmes sont caractérisés par :

- la mise en place de matériel spécialisé (communications entre sites distants par fibre optique, FDDI<sup>13</sup>, Fibre Channel<sup>14</sup>, système de commutation réseau contrôlé par ligne série<sup>15</sup>);
- un fonctionnement possible en mode dégradé (système de fichiers tolérant aux pannes);
- redémarrage et remise en état rapide du système après panne.

En plus d'une architecture massivement redondante, supprimant tous les points critiques du matériel, la mise en place d'un système de fichiers réseau tolérant aux pannes est un composant essentiel d'une telle solution. Bien que de tels systèmes sous licence libre soient encore très jeunes, ils apportent déjà des services de haute qualité :

- copie conforme des données sur plusieurs serveurs;
- conservation des données en cas de dégradation du réseau;

<sup>13</sup>**FDDI** (Fiber-Distributed Data Interface) : Standard de transmission de données par fibre optique sur un réseau local pouvant s'étendre sur 200 km. Le protocole FDDI est basé sur le protocole token ring. Il met en place deux anneaux qui peuvent être employés en anneau primaire / anneau de secours ou conjointement, ceci augmentant le débit jusqu'à 200 Mo/s.

<sup>14</sup>**Fibre Channel** : Technologie de transmission de données entre ordinateurs à haut débit (jusqu'à 1 Go/s). Elle est particulièrement employée pour connecter des unités de stockages aux serveurs. On peut utiliser le Fibre Channel sur plusieurs supports physiques : fibre optique (pour les grandes distances (10 km)), câble coaxial, paire torsadée.

<sup>15</sup>**Ligne série** : Moyen de communication entre équipements électroniques qui utilise un mode de transmission de données les unes à la suite des autres (par opposition au mode de transmission en parallèle). Sur un PC par exemple, la communication peut se faire par l'intermédiaire du port série suivant la norme RS-232.



- système sécurisé, encodage ;
- connections/déconnections «à chaud» ;
- journalisation des transactions<sup>16</sup>.

CodaFS<sup>17</sup>, ReiserFS, NFS, l'utilisation de Enhanced Network Block Device<sup>18</sup>, de RAID mettent à la disposition de la grappe de telles technologies. Le volume de stockage ainsi constitué est robuste et fournit de nombreuses facilités de gestion.

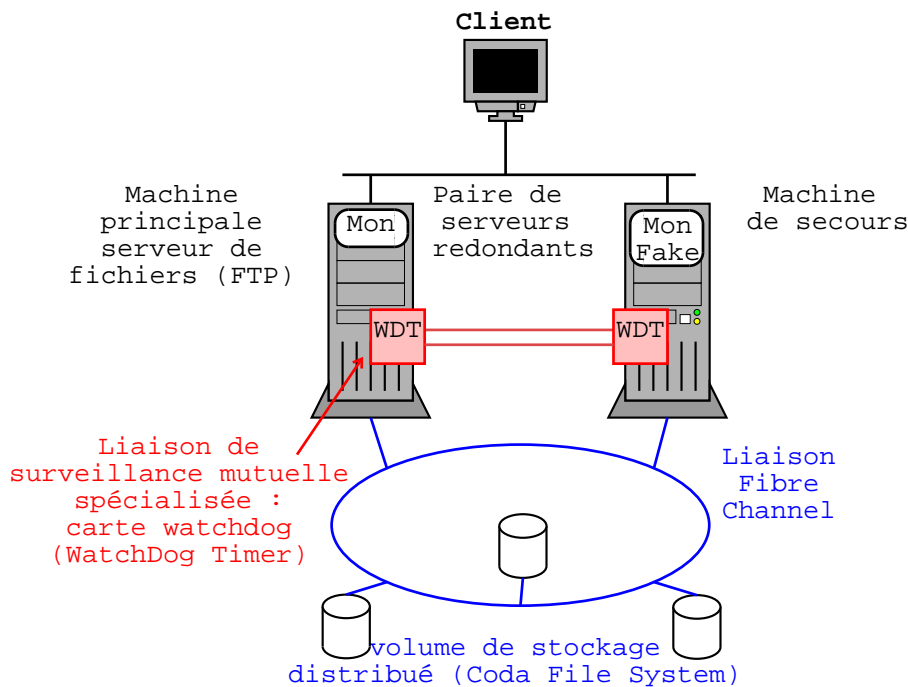


FIG. 2.4 – Serveurs de fichiers redondants tolérants aux pannes.

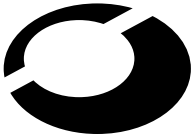
Cet exemple met en oeuvre plusieurs techniques de tolérance aux pannes pour sécuriser un ensemble de serveurs FTP<sup>19</sup>. La redondance agit ici comme dans l'exemple précédent : permettre de disposer d'un serveur de secours si la machine principale tombe en panne. Dans cet exemple, la surveillance locale et de la machine paire est confiée à une carte spéciale **WatchDog Timer ICS WDT501-P** qui se charge d'alerter le système en cas de panne. Le stockage est distribué sur un réseau **Fibre Channel (carte Qlogic QLA2x00)** et est géré par le système de fichiers **Coda**.

<sup>16</sup>**Système de fichiers journalisé** : système de fichiers assurant que toute mise à jour des données est stockée dans un journal de transactions avant d'être écrite sur le disque. Un tel système de fichiers permet de retrouver les données intactes, après un crash, et il réduit le temps de redémarrage du système crashé.

<sup>17</sup>**Coda File System** : Système de gestion de fichiers réseau, distribué. Permet d'implémenter un volume de stockage tolérant aux pannes, redondant et sécurisé.

<sup>18</sup>**ENBD** (Enhanced Network Block Device) : Module du noyau Linux permettant de voir un ensemble de blocs (disque dur, partition ou simplement fichier) distants comme faisant partie de la machine locale. Les échanges de données sont journalisés, peuvent être sécurisés et peuvent reprendre après une panne.

<sup>19</sup>**FTP** (File Transfer Protocol) : Protocole standard de l'Internet permettant d'échanger des fichiers entre machines. C'est un protocole applicatif qui utilise la pile TCP/IP.



## A Références

- Article présentant une solution haute disponibilité dans Linux Journal (<http://www2.linuxjournal.com/lj-issues/issue64/3247.html>)
- Article présentant les différents types de cluster (<http://www.linuxworld.com/linuxworld/lw-2000-03/lw-03-clustering.html>)
- HOWTO haute disponibilité (<http://metalab.unc.edu/pub/Linux/ALPHA/linux-ha/High-Availability-HOWTO.html>)
- Projet mettant en oeuvre diverses configuration de serveurs haute disponibilité (<http://ultramonkey.sourceforge.net>)
- Guide d'installation de Red Hat High Availability Server (<http://www.redhat.com/support/manuals/RHHAS-1.0-Manual/>)
- Linux Virtual Server (LVS) (<http://www.linuxvirtualserver.org>)
- Linux High Availability (<http://www.linux-ha.org>)
- GNU Queue (<http://queue.sourceforge.net>)
- Mon : Service Monitoring Daemon (<http://mon.sourceforge.net/>)
- Fake : Redundant Server Switch (<http://fake.sourceforge.net/>)
- Coda File System (<http://www.coda.cs.cmu.edu/>)
- ENBD (<http://www.it.uc3m.es/~ptb/nbd/>)