

00623  
25



# UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE CONTADURIA Y ADMINISTRACION

USO DEL SOFTWARE LIBRE EN LOS SISTEMAS  
DE ALTA DISPONIBILIDAD

TESIS PROFESIONAL QUE PARA  
OBTENER EL TITULO DE:  
LICENCIADO EN INFORMATICA

PRESENTA:

VICTOR AUGUSTO SANCHEZ RAMIREZ

ASESOR: D. R. RICARDO RIVERA SOLER

MEXICO, D. F.

2003



A



Universidad Nacional  
Autónoma de México



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

## **DEDICATORIA**

*Dedico este trabajo con mucho cariño a mi madre  
Patricia Ramirez Ortiz*

B

## **AGRADECIMIENTOS**

*A mi madre y a mi padre:*

Por que nunca dejaron de apoyarme y siempre creyeron en mí.

*A mis hermanos:*

Que siempre me mostraron su apoyo, aun cuando no se los pidiera.

*A la Universidad Nacional Autónoma de México*

Que me brindo todos los medios para desarrollarme tanto  
como persona y como profesionista

*Al Dr. Ricardo Rivera Soler:*

Que con su paciencia y enseñanzas me ayudaron a  
llegar al termino de este trabajo y de mi carrera.

*A mis amigos: Kristel, Javier, José, José Luis, Ivonne, Luis, Leslie*

Por todos los momentos que compartimos en la escuela, los cuales  
han sido los mejores de mi vida.

*Al Maestro en Ciencias Longino Jácome Pérez y a mi compañero  
de trabajo y, próximamente colega Victor Manuel Razo Soutyo:*

Que me brindaron consejos y apoyo durante el  
tiempo que estuve desarrollando este trabajo.

C

# Índice General

INTRODUCCIÓN	1
<b>MARCO PROBLEMÁTICO</b>	
Antecedentes	3
Identificación del problema	4
Demarcación del fenómeno	5
Opiniones profesionales	6
Opiniones no profesionales	7
Cuestionario para determinar la viabilidad del tema	8
Repuestas de los cuestionarios	11
Conclusiones de los cuestionarios	15
Hipótesis preliminar	15
Objetivos	16
<b>MARCO TEÓRICO</b>	
Libros	19
Lecturas de estudio	19
Lectura ligera	26
Tesis	27
Artículos de revistas	29
Sitios de Internet	30
Conferencias	31
<b>MARCO CONCEPTUAL</b>	
Definiciones	32
Derivación etimológica	32
Definiciones del diccionario	34
Definiciones de autores	35
Logrando la alta disponibilidad	36
Tipos de sistemas de alta disponibilidad	38
Causas de la caída de un sistema	40
Hardware	40
Ambiente y Fallas físicas	40
Fallas en la red	41
Fallos en los sistemas de bases de datos	41
Fallas en los servidores web	41
Clusters de alta disponibilidad	42
Tipos de cluster de alta disponibilidad	43
Componentes de un sistema de alta disponibilidad	46
Software Libre	57
¿Qué es el software libre?	57

D

¿Cómo surgió?	57
Ventajas y desventajas del uso del Software Libre	58
Licencias	60
Software para proteger el sistema	62
Monografía del tema	64
Alta disponibilidad con Linux	67
Proyecto Ultramonkey	69
Cuadro de clasificación del tema	72

## MARCO METODOLÓGICO

Variables	73
Variables de control	74
Hipótesis definitiva	74
Definición del Universo	74
Muestra	75
Definición del método de la investigación	75
Costos de la investigación	76
Colaboradores y apoyos	76
Cuestionario preliminar	76
Cuestionario Definitivo	80
Realización de la encuesta	82
Captura de la información	82
Análisis de los resultados	89
Conclusiones de la encuesta	90

## MARCO INSTRUMENTAL

Propuestas de acción	92
Plan de acción para la realización de la página en Internet	93
Plan de acción para la participación en el CONSOL 2003	95
Plan de acción para artículos en el suplemento enter@te	95
Plan de acción para proponer a la Jefa de la carrera de Informática la celebración de una conferencia	95
Plan de acción para proponer a la Jefa de la carrera de Informática, la inclusión de temas relacionados con el software libre, en materias de la carrera	96

## CONCLUSIONES

Conclusiones	97
Consideraciones finales	98

E

## ANEXOS

Glosario	100
Licencias OpenSource	103
Oficio entregado en la Jefatura de Informática	104
Epígrafe	106

## BIBLIOGRAFÍA

108

f

# Introducción

Software libre es aquel que se encuentra bajo un esquema de desarrollo cooperativo; es decir, que todo aquel que este interesado en desarrollar, corregir o mejorar alguna aplicación, ya sea propia o de alguien más, puede hacerlo. Para que esto se logre es necesario que todas las personas tengan acceso al código fuente del programa. Esto da como resultado la creación de aplicaciones robustas y confiables, ya que al contar con la participación de personas de todo el mundo, el programa cuenta con las ideas de todos ellos.

En el presente trabajo abordo el tema del software libre aplicado a los sistemas de alta disponibilidad. Esta tesis no pretende ser un documento exhaustivo acerca del tema ya que este es muy extenso y diverso, más bien se enfoca a explicar conceptos básicos de alta disponibilidad presentar algunas herramientas libres que pueden ser utilizadas en uso del software libre en esta clase de sistemas.

En el primer capítulo (**Marco Problemático**) hago el planteamiento del problema, además de que analizo las causas y consecuencias del mismo. Para determinar la viabilidad del tema entrevisté a varias personas tanto profesionales como no profesionales para saber su opinión acerca del uso del software libre en los sistemas de alta disponibilidad.

En el **Marco Teórico** presento las distintas fuentes que ofrecen información acerca del tema de la alta disponibilidad. No presento todas las existentes debido a que son muy variadas en extensión y forma; simplemente presento las que proporcionaron más información para la investigación.

En el capítulo que conforma el **Marco Conceptual** se presenta la explicación de los conceptos teóricos, y se trata a profundidad el tema de los sistemas de alta disponibilidad. Se presenta la descripción de algunos productos y lo que es el software libre y las herramientas que existen para lograr sistemas altamente disponibles.

En el **Marco Metodológico** presento nuevamente las variables que están

---



involucradas en el uso del software libre en los sistemas de alta disponibilidad y la realización que tienen entre ellas; además en este capítulo presento la hipótesis definitiva, la cual presenta ligeros cambios respecto a la preliminar. Al final de este capítulo se habrá hecho la aprobación o desaprobación de la hipótesis de acuerdo como el análisis de resultados de las encuestas aplicados lo indiquen.

El **Marco Instrumental** presenta las acciones que emprendí una vez que comprobé la veracidad de la hipótesis, estas acciones buscan principalmente la difusión del trabajo realizado con el fin de que sea aprovechado por quien lo necesite.

En el último capítulo de esta tesis hago las **Conclusiones** pertinentes a las que pude llegar a través de la realización de la investigación.

Este trabajo viene acompañado por varios anexos que enriquecen el contenido teórico de los capítulos que conforman esta tesis.

## Capítulo 1

# MARCO PROBLEMÁTICO

*"no sería paradójico afirmar que el hombre que plantea un problema no es enteramente el mismo que lo resuelve"*

**Ramón Y Cajal**

## Antecedentes

Desde los últimos 3 años he sido usuario del software libre (SL), principalmente del sistema operativo Linux, por lo cual he tenido contacto con la ideología sobre la cual se basa este movimiento. También he sido testigo de la calidad de los programas que se han desarrollado bajo esta filosofía de desarrollo. Es por eso que estoy convencido de que el SL es tan bueno como el software propietario<sup>1</sup>, y que puede ser utilizado en muchos ámbitos de la vida tanto laboral como académica.

Uno de los campos donde considero que el software libre puede ser utilizado, es en el de los sistemas de alta disponibilidad (tema por el que también tengo interés), esto es debido a que las herramientas desarrolladas como software libre poseen características técnicas similares al software propietario, es decir, robustez, seguridad, utilidad. He sido usuario de sistemas que debido a su

---

<sup>1</sup> Software propietario es aquel cuyo código fuente sólo está a disposición de los programadores de la empresa que desarrolló el software, y que no se permite su modificación o redistribución

importancia necesitan alguna clase de protección; sin embargo, no se les implementa (la mayoría de las veces por desconocimiento de las herramientas que se pueden utilizar) y creo que el software libre puede ser una buena opción para hacerlo.

Un ejemplo de esto es un sistema de inscripciones escolares. Este sistema debe de estar disponible para que los alumnos puedan realizar sus trámites, y si mientras ellos se están inscribiendo el sistema falla, pues se creará un gran descontento entre los usuarios, y las fechas de realización del trámite de inscripción deberán de ampliarse, lo que conlleva a un problema ya que habrá que invertir más recursos (por parte de la dirección) para realizar el trámite.

Mi interés en desarrollar el tema del software libre aplicado a los sistemas de alta disponibilidad es debido a que creo que de esta manera contribuyo a darle difusión a herramientas que son poco conocidas, y que con ellas se puede proteger a los sistemas contra fallos; además de que me permite conocer el tema para posteriormente aplicarlo en mi área de trabajo.

## **Identificación del problema**

Como administrador de sistemas estoy consciente de la importancia de tener el servicio (o servicios) disponible cuando los usuarios lo requieran. Además de que yo mismo he sido usuario de sistemas que, en mi opinión, debería de contar con protección en caso de fallas y que sin embargo no lo tienen.

Al momento de documentarme sobre el tema he descubierto que la mayoría de las soluciones creadas para los sistemas de alta disponibilidad son proporcionadas por grandes compañías consultoras con el uso de hardware caro, y en muchos casos con costos excesivos para pequeñas empresas, es por eso que creo que el software libre es una buena opción para proteger los sistemas sin realizar gastos enormes.

Por esto este trabajo muestra las causas por las cuales el SL no es utilizado en los sistemas de alta disponibilidad

Sin embargo cuando el problema no es atendido las consecuencias son variadas:

- Cuando un servicio no cuenta con la protección adecuada contra fallas se corre el peligro de perder tiempo y dinero cada vez que el servicio no está disponible, siendo que se puede implementar de una manera relativamente sencilla ya que a las herramientas libres se puede tener

- acceso muy fácilmente.
- Las organizaciones que tengan servicios de alta disponibilidad probablemente están gastando más dinero utilizando soluciones basados en software propietario que si utilizaran software libre.
- Las compañías que proporcionan soluciones, basadas en software libre están perdiendo oportunidades de negocio debido a que los clientes buscan soluciones y al no contar con una oferta por parte de empresas que se dedican al software libre.
- No se le está dando una adecuada difusión al software libre debido a la falta de oportunidades para demostrar la clase de soluciones que se pueden desarrollar utilizándolo.

## **Demarcación del fenómeno**

El tema de la alta disponibilidad es muy extenso, requiere un tratamiento mucho mayor que el que un trabajo de tesis puede dar, esto es debido a que los distintos niveles de disponibilidad puede ser requeridos para toda clase de sistemas e involucrar el uso de tecnologías desde las muy sencillas hasta algunas en extremo complejas.

No es mi objetivo hablar sobre los grandes y caros sistemas de alta disponibilidad que algunas compañías (como los bancos o las compañías áreas) implementan. Sino más bien sobre aquellos que buscan la disponibilidad y protección de la información a un nivel suficiente para servidores de archivos, web o correo, sistemas que algunas compañías utilizan diariamente para desempeñar sus funciones.

Además, esta tesis no pretende cubrir a gran profundidad todos los aspectos que involucran los sistemas de alta disponibilidad (hardware, tecnologías de conectividad, ambiente físico del centro de datos, etc.), debido a que cada punto requiere un tratamiento muy extenso y no puede ser cubierto en un trabajo de tesis. En esta tesis sólo trataré el tema relacionado al software utilizado en la instalación de sistemas de alta disponibilidad. Para mayor información de los puntos no tratados en esta tesis puede referirse a la bibliografía presentada al final del trabajo.

## Opiniones profesionales

En esta sección presento a las personas que, con sus respuestas a un cuestionario que les aplique, me ayudaron a determinar la viabilidad del tema. Su opinión es considerada como profesional ya que, además de haber implantado por lo menos un par de sistemas de alta disponibilidad tienen conocimiento sobre el software libre.

Nombre:	Tony Bourke
Ocupación:	Consultor privado, especialista en administración de UNIX
Razón de la encuesta:	Tony Bourke ha diseñado he implementado servidores de balanceo de carga para algunos sitios web con alto tráfico. Ha publicado artículos en revistas como Sys Admin Magazine, Hostintech y Network World. Escribió además el libro llamado "Server Load balancing" publicado por la editorial O'reilly, ISBN: 0596-00050-2. Puede ser contactado mediante la dirección electrónica: <a href="mailto:tony@vegan.net">tony@vegan.net</a> .

**Tabla 1.1:** OP: Tony Bourke

Nombre:	Steve Blackmon
Ocupación:	Administrador de sistemas
Razón de la encuesta:	Junto con John Nguyen escribí un artículo llamado "High Availability File Server with Heartbeat" para la revista Sys Admin Magazine (Vol. 9 No. 10, septiembre 2001) en el que habla de la alta disponibilidad en los sistemas Linux. Steve ha sido desarrollador de software y administrador de sistemas por 14 años. Actualmente proporciona consultoría en el área de la alta disponibilidad y SAN en la ciudad de Atlanta. Su correo electrónico es: <a href="mailto:steve.blackmon@transtech.com">steve.blackmon@transtech.com</a> .

**Tabla 1.2:** OP: Steve Blackmond

Nombre:	Victor Manuel Razo Soutuyo
Ocupación:	Administrador de sistemas
Razón de la encuesta:	Trabaja como administrador de sistemas en la Dirección General de Estadística y Desarrollo Institucional, ha implantado sistemas de alta disponibilidad utilizando tanto sistemas SUN como Windows NT. Tiene conocimiento en el área de Software Libre y maneja el sistema operativo Linux. Su correo electrónico es: <a href="mailto:vmrazo@estadistica.unam.mx">vmrazo@estadistica.unam.mx</a>

Tabla 1.3: OP: Victor Razo Soutuyo

## Opiniones no profesionales

Las personas que se encuentran en este grupo poseen conocimientos en administración de sistemas pero no necesariamente conocen o han utilizado el software libre.

Nombre:	Gunnar Eyal Wolf Iszaevich
Ocupación:	Administrador de sistemas
Razón de la encuesta:	Él, es una de las personas que mejor conocen las posibilidades y limitaciones del software libre. Ha dado varias conferencias acerca de software libre y seguridad en informática. Actualmente trabaja como administrador de la red en la Facultad de Estudios Superiores campus Iztacala de la UNAM, y es miembro externo del área de seguridad en cómputo de DGSCA, UNAM. Puede ser contactado a través de su correo electrónico: <a href="mailto:gwolf@gwolf.cx">gwolf@gwolf.cx</a> .

Tabla 1.4: Gunnar Eyal Wolf Iszaevich

TESIS CON  
FALLA DE ORIGEN

Nombre: Alejandro G. Sánchez Martínez

Ocupación: Gerente de e-compugraf S.A

Razón de la encuesta: Es usuario de software libre y fundó una empresa llamada e-compugraf que brinda soluciones basadas en software libre. Su correo electrónico es asanchezm@e-compugraf.com.

Tabla 1.5: Alejandro G. Sánchez Martínez

Nombre: Felipe Javier Rangel Pompa

Ocupación: Web master

Razón de la encuesta: Es administrador de sistemas Windows NT, fue elegido para esta encuesta debido a que él no es usuario de software libre y su opinión será importante ya que representa un sector que aun es usuario de software propietario. felipe@2-bweb.com

Tabla 1.6: Felipe Javier Rangel Pompa

## Cuestionario para determinar la viabilidad del tema

El siguiente es el cuestionario que se les aplicó a estas personas para determinar la viabilidad del tema.

1. ¿Cuántos sistemas de alta disponibilidad ha implantado?

**Razón de la pregunta:** Saber cual es la experiencia que tiene el entrevistado acerca del tema.

**Respuesta esperada:** Algún número mayor a 1.

2. ¿Cuáles cree usted que son las fuentes de información más apropiadas para investigar acerca del tema?

**Razón de la pregunta:** Saber a que tipo de fuentes debo recurrir para investigar el tema.

**Respuesta esperada:** Nombre de algún libro o alguna dirección de Internet

3. ¿ Conoce qué es el software libre y su filosofía de trabajo?

**Razón de la pregunta:** Saber si el entrevistado conoce el software libre para las siguientes preguntas.

**Respuesta esperada:** Sí. **Relación:** Pregunta 4.

4. ¿ Cree que el software libre es una opción viable para establecer un sistema de alta disponibilidad?

**Razón de la pregunta:** Saber si el entrevistado comparte la idea de que el software libre puede aplicarse en sistemas de alta disponibilidad.

**Respuesta esperada:** Sí.

5. ¿ Ha implantado algún sistema de alta disponibilidad utilizando software libre? ¿ Qué herramientas utilizó?

**Razón de la pregunta:** Saber si ha tenido éxito implantando un sistema con software libre.

**Respuesta esperada:** Sí.

6. ¿ Podría mencionar algunas ventajas y desventajas de utilizar el software libre en estos sistemas?

**Razón de la pregunta:** Comparar las ventajas y desventajas del uso de esta clase de software.

**Respuesta esperada:** Más ventajas que desventajas.

7. ¿ Qué herramientas o programas que sean software libre conoce que se utilicen para lograr la alta disponibilidad en el servicio web. ?

**Razón de la pregunta:** Conocer en que herramientas puedo enfocarme para su estudio.

**Respuesta esperada:** Heartbeat, LinuxHA, etc.

8. ¿ Cuánto es el tiempo promedio que ha invertido en implementar un sistema de alta disponibilidad?. Por favor considere desde el análisis hasta cuando el sistema ya esta en producción.

**Razón de la pregunta:** Conocer cuanto es el tiempo promedio de implantación de esta clase de sistemas.

**Respuesta esperada:** Un periodo entre 1 semana y un mes.

9. Describa brevemente los problemas a los que se enfrenta al momento de



instalar un sistema de alta disponibilidad.

**Razón de la pregunta:** Conocer los principales problemas que se suceden en la implantación de estos sistemas.

**Respuesta esperada:** La que el entrevistado proporcione.

## Repuestas de los cuestionarios

Opiniones Profesionales			
Número de pregunta	Tim Bourke	Steve Blackmond	Victor Manuel Razo Soutuyo
¿Cuántos sistemas de alta disponibilidad ha implantado?	Un número considerable, me atrevería a decir que como 100 en los últimos 5 años.	5 - 6	2
¿Cuales cree usted que son las fuentes de información mas apropiadas para investigar acerca del tema?	La alta disponibilidad es un tema muy extenso, puede cubrir una variedad de instalaciones, redes, balanceo de servidores, aplicaciones de hosting, entrega de contenido, etc. Network magazine es un lugar, algunas otras revistas comerciales otro, mi libro es otro.	Internet, deja.com, etc	www.linux-ha.org, Answer book de Solaris; HW, Admin, Sis Op
¿Conoce que es el software libre y su filosofía de trabajo?	Si, opensource, software libre, GNU, GPL y BSD todos son términos para una variedad de proyectos de software. Tan variados son, que completas infraestructuras han sido construidas sobre estos tipos de productos. La mayoría de las compañías utilizan estas tecnologías en algún grado, y hay otras que dependen de ellas exclusivamente.	Si	Si

TESIS CON  
 FALLA DE ORIGEN

Opiniones Profesionales			
Número de pregunta	Tim Bourke	Steve Blackmond	Victor Manuel Razo Soutuyo
¿Ha implantado algún sistema de alta disponibilidad utilizando software libre? ¿Que herramientas utilizó?	Bueno, de sistema operativo uso Linux, servidor de web apache 1.0.x o 2.x dependiendo. MySQL para bases de datos, y Php para el desarrollo de aplicaciones, aunque no tengo mucha experiencia en la programación de las aplicaciones, yo solo corro los sistemas; Para administración uso Openssh el cual me permite la administración de servidores UNIX/Linux.	Si, use Heartbeat sobre Linux junto con abundante software libre.	No
¿Podría mencionar algunas ventajas y desventajas de utilizar el software libre en estos sistemas?	La ventaja del software libre es, por supuesto, que es libre. Esto no significa que es gratis o sin costo, le cuesta a alguien instalarlo y trabajar sobre el software pero esto quiere decir ahorro significativo sobre aplicaciones comerciales	Junto con el soporte (ver pregunta 5), los productos comerciales están en ocasiones más "pulidos", esto una vez más tiende a ser un gran elemento para las personas que son menos habilidosas.	Ventajas: No estar atado a contratos y la posibilidad de adaptar el programa a mis necesidades, facilidad de distribución, rendimiento en costos. – Desventajas: Convencer a los jefes

TESIS CON  
FALLA DE ORIGEN

Opiniones no profesionales			
Pregunta	Gunnar Eyal Wolf Iszaevich	Alejandro G. Sánchez Martínez	Felipe Javier Rangel Pompa
¿Conoce que es el software libre y su filosofía de trabajo?	Sí	Sí	Sí
¿Cree que el software libre es una opción viable para establecer un sistema de alta disponibilidad?	Completamente, pues ofrece todas las alternativas necesarias, y permite ajustarla a las necesidades específicas de cada sitio	Sí. Por su alta calidad, seguridad y posibilidad de adecuación a las necesidades de cada requerimiento,	No. Por que las empresas que manejan sistemas de alta disponibilidad requieren de un soporte que garantice la funcionalidad de su aplicación y el software libre actualmente no cuenta con dicho respaldo.
¿Podría mencionar algunas ventajas y desventajas de utilizar el software libre en estos sistemas?		<p>Ventajas:</p> <ul style="list-style-type: none"> <li>- Disponibilidad,</li> <li>- seguridad, fiabilidad,</li> <li>- corrección rápida de errores, adecuación sencilla a las necesidades de cada cual gracias a que se cuenta con el código.</li> </ul> <p>Desventajas: Ninguna</p>	<p>Ventajas:</p> <ul style="list-style-type: none"> <li>- Reducción de costos en licencias</li> <li>- Alta Funcionabilidad</li> <li>- Alta Disponibilidad</li> <li>- Alta Seguridad</li> </ul> <p>Desventajas:</p> <ul style="list-style-type: none"> <li>- No cuenta con soporte técnico</li> <li>- Es difícil encontrar gente capacitada</li> <li>- Tiene una difícil implementación</li> </ul>

TESIS CON  
FALLA DE ORIGEN

Opiniones no profesionales			
Pregunta	Gunnar Eyal Wolf Iszaevich	Alejandro G. Sánchez Martínez	Felipe Javier Rangel Pompa
¿Que herramientas o programas que sean software libre conoce que se utilicen para lograr la alta disponibilidad en el servicio web. ?	Dependiendo del esquema, se puede hacer de muchas maneras. Mosix, round-robin de DNS, enmascaramiento, etc.	No son paquetes en si, sino implementaciones de diferentes cosas	<ul style="list-style-type: none"> <li>- MySql</li> <li>- PHP</li> <li>- Linux</li> </ul> Como desarrollo: <ul style="list-style-type: none"> <li>- Acer maneja su soporte técnico con una aplicación bajo el desarrollo de PHP y MySql.</li> <li>- El sistema de garantías de Sharp se maneja con perl y postgres</li> </ul>

**TESIS CON  
FALLA DE ORIGEN**

## Conclusiones de los cuestionarios

Después de haber analizado los cuestionarios aplicados, se pueden obtener las siguientes conclusiones de acuerdo con las opiniones de las personas:

- El uso del software libre en los sistemas de alta disponibilidad está determinado por varios factores: el tipo de sistema del que se trata y la habilidad de la persona que administrará el sistema.
- La ventaja de utilizar software libre es que se puede adaptar a las necesidades de cada quien; sin embargo no se obtiene soporte como el que se puede obtener con productos comerciales.

## Hipótesis preliminar

Para ofrecer una respuesta preliminar al problema es necesario revisar que es lo que lo causa; es decir enumerar las razones por las que el software libre no se utiliza en los sistemas de alta disponibilidad:

**Causa:** Hardware incompatible.

**Efecto:** El hecho de no disponer controladores para el hardware con el que se cuenta obliga a los administradores a buscar soporte con alguna otra aplicación que brinden soporte a dicho hardware.

**Causa:** Aplicaciones difíciles de utilizar.

**Efecto:** Descontento por parte de los administradores del sistema.

**Efecto:** Renuencia al uso del software libre en sistemas de alta disponibilidad.

**Causa:** Personal no capacitado.

**Efecto:** No se pudo emprender un proyecto nuevo utilizando las herramientas libres debido a que no se cuentan con los conocimientos adecuados.

**Causa:** Políticas internas.

**Efecto:** No se busca en otras tecnologías mejores herramientas de software.

**Causa:** Costumbre.

**Efecto:** Se pierden oportunidades de probar nuevas herramientas, al no contar con iniciativa para probar otras tecnologías

**Causa:** No hay respaldo por parte de empresas.

**Efecto:** Desconfianza por parte de los administradores de sistemas (y varias personas encargadas de tomar la decisión de que software utilizar), que piensan que no hay seriedad en el desarrollo del software libre.

**Efecto:** Las compañías están perdiendo oportunidades de negocio, para el mercado de los sistemas de alta disponibilidad.

**Causa:** Software propietario es mejor.

**Efecto:** Las empresas buscan en otras compañías de software lo que en las herramientas libres no pueden encontrar.

Como se puede ver existen varias causas por las cuales el software libre no es utilizado, sin embargo para efectos de esta tesis se tomarán las que considero más adecuadas. Por lo que la hipótesis preliminar, queda como sigue:

***El uso del software libre para la implantación de los sistemas de alta disponibilidad no es muy frecuente, debido a que no hay empresas que respalden la tecnología en caso de que ocurran problemas y que los administradores del sistema no están enterados de la existencia de dichas herramientas.***

En la figura 1.1 se pueden observar un diagrama de causa-efecto que esquematiza de manera clara las causas por las que el software libre no es utilizado para la protección de los servicios.

## Objetivos

### Personales

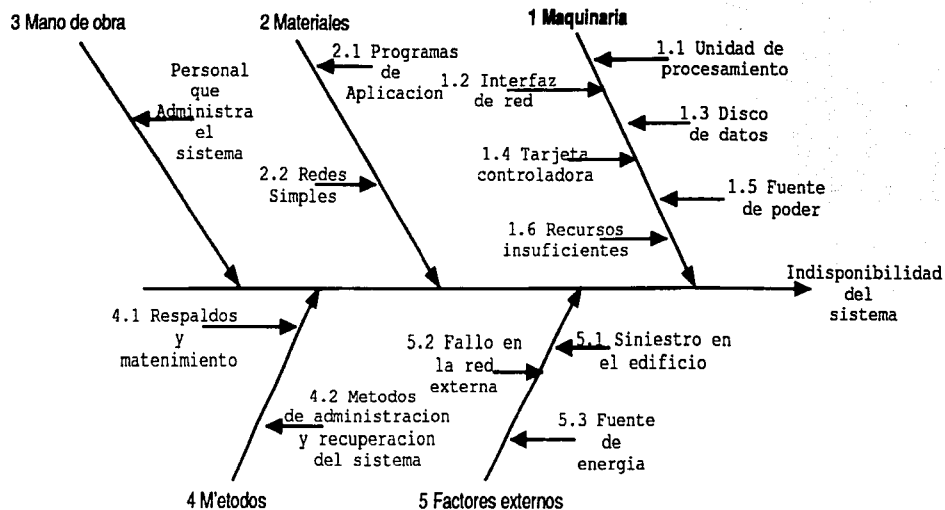
- Obtener el título de licenciado en informática conforme lo establece el "Nuevo reglamento general de exámenes profesionales" de la Facultad de Contaduría y Administración de la UNAM. Título I artículo 3, Título II artículos 7 inciso e, 9 inciso g y artículos 44 al 56.

- Dominar el tema de la alta disponibilidad de manera que sea capaz de aplicarlo en mi trabajo en alguna aplicación de misión crítica que se requiera.

#### Generales

- Dejar una fuente de información para todo aquel que tenga interés en el tema del software libre y la administración de sistemas.
- Dejar el trabajo como una prueba documentada de que el software libre tiene un gran futuro y puede ser utilizado para esta clase de sistemas.





**Figura 1.1:** Causas de la no-utilización del software libre en los sistemas de alta disponibilidad

TESIS CON  
FALLA DE ORIGEN

## Capítulo 2

# MARCO TEÓRICO

*"Los libros no dan sabiduría donde antes no la había,  
pero donde algo hay, la lectura la incrementa"*

**SIR John Harington**

### Libros

#### Lecturas de estudio

Bajo esta categoría se encuentran todos los libros que aportan gran información acerca del tema.

Libro: Blueprints for high availability  
Autores: Evan Marcus y Hal Stern.  
Editorial: Wiley.  
Primera edición.  
ISBN: 0-471-35601-8.

Este es un gran libro que proporciona amplia información y muy completa acerca de las formas que existen para proteger a los sistemas.

---

**EXPLICACIÓN SINTÉTICA DE CADA CAPÍTULO**

---

**Capítulo 1. Introducción:** En este capítulo se presenta la introducción al libro; los autores explican sus objetivos; y mencionan las razones que los llevaron a escribir el libro.

**Capítulo 2. ¿Qué es resistencia?** : Este segundo capítulo se refiere a los conceptos básicos de disponibilidad; explica como medirla; también explica algunas de las fallas más comunes que pueden presentar los sistemas (ya sea de hardware o software. Además en este capítulo se hace un análisis de los costos que puede provocar un fallo. Por último se explican los distintos niveles de disponibilidad que existen.

**Capítulo 3. 20 principios para el diseño de sistemas:** Aquí los autores brindan lo que ellos consideran los 20 principios a seguir al momento de diseñar un sistema. Algunos de estos principios son: Eliminar los puntos únicos de fallo, documentar todo, usar hardware y software confiable, etc.

**Capítulo 4. Administración de datos altamente disponible:** Se presenta una descripción de las distintas tecnologías en hardware que se utiliza para proporcionar mayores niveles de protección del sistema: RAID, LVM, etc.

**Capítulo 5. Diseñando servidores redundantes:** En este capítulo se explican todos los aspectos relacionados para la instalación de un servidor protegido mediante la redundancia de sus componentes.

**Capítulo 6. Manejo de una "failover":** En este capítulo se analizan los distintos métodos de manejar las fallas en el sistema; además de que se dan algunas recomendaciones para monitorear los elementos que componen a un sistema crítico.

**Capítulo 7. Configuraciones para el failover:** Aquí se analizan algunas configuraciones que pueden ser usadas para permitir un "failover" se mencionan 3 clases de grupos de configuraciones: el primero es el que utiliza un cluster de dos nodos, el segundo, es aquel con más de 2 nodos y por último el tercero que se refiere a configuraciones no tan comunes.

**Capítulo 8. Servicios de red redundantes:** Se analizan algunas razones por las cuales falla una red, además de que se mencionan métodos para enfrentar estas fallas.

**Capítulo 9. Confiabilidad del servicio de datos:** Este es un capítulo que habla de cuestiones técnicas, se examinan algunos servicios como el NFS, web, o las bases de datos, también se da una guía para poder elegir entre varias opciones

para la protección del sistema.

**Capítulo 10. Técnicas de replicación:** Aquí se trata el tema de la replicación de los datos, y la relación que este tiene con los niveles de disponibilidad de los sistemas. Trata desde la replicación de un sistema de archivos hasta el de una base de datos de gran tamaño.

**Capítulo 11. Recuperación de la aplicación:** Aquí se analizan los aspectos relacionados con el fallo de las aplicaciones que están corriendo dentro del sistema de alta disponibilidad.

**Capítulo 12. Respaldos y restauraciones:** Este capítulo se refiere a la importancia que tiene el realizar respaldos, además de que muestra algunas características de dispositivos de almacenamiento como cintas, o discos.

**Capítulo 13. Sistema de operaciones:** Se presentan algunas cuestiones relacionadas a la operación del sistema; se explican temas como la ubicación física del centro de datos, aspectos relacionados a la correcta elección de proveedores, etc. Los aquí tratados no son temas completamente técnicos pero son de gran importancia para el diseño de un buen sistema.

**Capítulo 14. Recuperación de desastres:** Presenta el nivel mayor de disponibilidad (y el más difícil de lograr); el prepararse para enfrentar desastres significa hacer al sistema resistente a situaciones o daño total ocasionadas por guerras, actos de terrorismo o desastres naturales; este nivel de protección es obligatoria para sistemas realmente críticos como servicios financieros, salud, etc.

**Capítulo 15. Parting shot:** En este capítulo sólo se hacen las precisiones finales y las conclusiones del libro.

**Libro: Clusters for high availability.**

**Autor: Peter S. Weygant.**

**Editorial: Prentice Hall.**

**Segunda edición.**

**ISBN: 0-013-0893552.**

Este libro es una guía imprescindible para el tema de la alta disponibilidad ya que trata los conceptos fundamentales del tema, así como varias tecnologías de clusters de alta disponibilidad aplicadas a diferentes sistemas operativos.

EXPLICACIÓN SINTÉTICA DE CADA CAPÍTULO

---

**Capítulo 1** Conceptos básicos de alta disponibilidad: En este capítulo el autor explica los principales conceptos relacionados con la alta disponibilidad; explica los niveles de disponibilidad y como medirla; da algunas recomendaciones para aplicarlo en las empresas.

**Capítulo 2** Clustering para eliminar los SPOF<sup>1</sup>: En el segundo capítulo muestra una lista de los diferentes tipos de puntos de falla y como eliminarlos. También en la última parte del capítulo se trata el tema de la implementación de un cluster de alta disponibilidad.

**Capítulo 3** Componentes de un cluster de alta disponibilidad: En este capítulo se da el panorama de como está compuesto un cluster de alta disponibilidad; además explica como es su dinámica de funcionamiento. Se da el nombre de algunas herramientas para cada tipo de cluster y se explica su funcionamiento. También presenta soluciones para otros subsistemas del cluster como son: fuente de poder, sistema de archivos, monitor de transacciones.

**Capítulo 4** Monitoreo y administración del cluster: Aquí se presentan unas técnicas útiles para monitorear el comportamiento del sistema. Este capítulo se centra en el enfoque de la administración y monitoreo proactivos<sup>2</sup>.

**Capítulo 5** Sistemas de alta disponibilidad tolerantes a desastres: Este capítulo presenta una forma de llevar a los sistemas de alta disponibilidad a un nivel mayor de protección contra fallos, mediante una arquitectura tolerante a desastres. Este nivel se logra mediante la redundancia del centro de datos completo y el enlace de estos a diferentes distancias ya sea, local, a nivel de campus, metropolitana o intercontinental.

**Capítulo 6** Soluciones de alta disponibilidad para las empresas grandes: Aquí se aborda algunas soluciones aplicables a las empresas con respecto a los clusters de alta disponibilidad, estas soluciones se enfocan en el almacenamiento. Discute tecnologías como SAN, cintas compartidas, entre otras. También aquí se explica el enfoque llamado "5 *nueves:5 minutos*" de HP.

**Capítulo 7** Ejemplo de soluciones de alta disponibilidad: Este capítulo presenta algunos ejemplos de configuraciones de clusters de alta disponibilidad que resuelven problemas reales de empresas.

**Capítulo 8** Glosario de terminología de alta disponibilidad: Aquí se presenta el

---

<sup>1</sup>Single Point Of Failure – Punto único de fallo (ver Glosario)

<sup>2</sup> Es decir, detectar y solucionar las fallas antes que ocurran.

significado de los términos técnicos que fueron utilizados a lo largo del libro.

Este es un buen libro que se enfoca en soluciones basadas en hardware y software de Hewlett-Packard. La mayoría de las soluciones que presenta se basan en hardware especializado y por lo tanto caros para la clase de sistemas en los que se enfoca esta tesis.

Libro: HIGH AVAILABILITY desing, techniques and processes.

Autores: Floyd Piedad y Michael Hawkins.

Editorial: Prentice Hall.

Primera edición.

ISBN: 0-13-096288-0.

Este libro presenta el tema de los sistemas de altamente disponibles, no desde un punto de vista exclusivamente técnico (como en el libro anterior); sino más bien desde un punto de vista estratégico. Desarrolla de manera muy entendible los temas que contiene el libro y está enfocado tanto para el personal técnico como para los jefes del área de sistemas.

---

#### EXPLICACIÓN SINTÉTICA DE CADA CAPÍTULO

---

**Capítulo 1: Ambiente computacional actual:** En este capítulo los autores hacen una comparación a cerca del panorama actual de las tecnologías del cómputo con décadas anteriores, cuando se usaban los Mainframes y un esquema de procesamiento central. Aquí explican lo que ellos denominan las 6 principales áreas en donde el campo de la tecnología se ha vuelto más complejo con respecto a años anteriores; y estas son: tecnologías y protocolos, proveedores, usuarios, localización de los sistemas de cómputo, cambio constante en las innovaciones tecnológicas y por último aumento en los requerimientos de los negocios. La complejidad de estas áreas de la tecnología radica principalmente en la amplia gama de productos, servicios y proveedores que existen en el mercado. También analizan el TOC (Total Cost of Ownership) o costo total de pertenencia que se refiere al costo total de adquirir y mantener un sistema.

**Capítulo 2: Alcanzando mayor disponibilidad:** La primera parte de este capítulo se refiere a los requerimientos de disponibilidad; es importante acordar, entre los

usuarios y el encargado de diseñar el sistema, un nivel de disponibilidad que les sea suficiente a los usuarios y que sea lo suficientemente realista como para alcanzarlo. También se presenta un pequeño cuestionario (Págs. 14-15) que se puede aplicar para contar con elementos para decidir el nivel de disponibilidad que requieren los usuarios de un sistema. En este capítulo también se explican los distintos niveles de disponibilidad que se pueden lograr (*alta disponibilidad, nivel de operación continua, disponibilidad continua*) en donde este último es el más difícil de lograr y el más costoso.

**Capítulo 3: Planeando para la disponibilidad del sistema:** Este capítulo se dedica a identificar los componentes de un sistema de alta disponibilidad, los cuales pueden ser clasificados como: hardware, software, ambiente, procesos y procedimientos y personal; una vez identificados los elementos que componen al sistema es necesario identificar los "*elementos críticos*" con el fin de evitar o reducir los fallos, así como hacer más eficiente la recuperación del sistema en caso de fallo.

**Capítulo 4: Preparación para la administración del sistema:** Este capítulo se refiere a temas relacionados con la administración del sistema, menciona 4 elementos que combinados llevan a un buen nivel de desempeño del sistema, y que cuya carencia pueden ocasionar mal funcionamiento del mismo, éstos son: *Procesos, datos, herramientas y organización*. También este capítulo hace énfasis en la importancia de conocer el sistema que se está administrando, no solo técnicamente, sino también todo el ambiente que lo rodea. Por último este capítulo presenta las 5 fases que se deben seguir para la administración de un sistema; fijación de objetivos, planeación, ejecución, medición y control; éstas fases forman un proceso cíclico que llevará a una adecuada administración del sistema.

**Capítulo 5: Administración del nivel del servicio:** Una vez que se ha acordado con el usuario un nivel de servicio a proporcionar (es decir el compromiso por parte del proveedor a tener el sistema trabajando durante las horas acordadas) es necesario mantenerlo; en este capítulo se presentan varias recomendaciones para lograr el mantenimiento del nivel de servicio. Aquí se menciona que para lograr la adecuada administración es necesario la presencia de los cuatro elementos fundamentales mencionados en el capítulo 4.

**Capítulo 6: De ambientes de cómputo centralizado a ambientes distribuidos:** En este capítulo se habla de los 2 ambientes de cómputo que puede existir en un ambiente empresarial, el ambiente centralizado (que implica el uso de una computadora Mainframe y todos los usuarios se conectan con lo que se denomina terminales tontas) y el ambiente distribuido (que implica el uso de varios sistemas de cómputo con capacidades de procesamiento y almacenamiento); Presenta las ventajas y desventajas de cada uno de ellos.

**Capítulo 7: Técnicas destinadas a múltiples requerimientos de disponibilidad:**

Aquí se explican distintas técnicas para proteger a los sistemas; entre ellos se encuentran *redundancia de componentes, realización de respaldos de recursos críticos, clustering, sistemas tolerantes a fallos, etc.* Además brinda algunos pasos a seguir para asegurar al sistema tanto físicamente como lógicamente.

**Capítulo 8:** Técnicas especiales para la confiabilidad del sistema: Este capítulo da algunas recomendaciones para la adecuada elección de los componentes del sistema, hardware, software y personal.

**Capítulo 9:** Técnicas especiales para la recuperación del sistema: Se centra en las medidas que son necesarias para lograr una recuperación más rápida del sistema en caso de fallas; en este capítulo los autores recomiendan el uso de archivos con Journaling (registro de bitácoras), o sistemas de reinicio automático.

**Capítulo 10:** Técnicas especiales para mantener el servicio del sistema: Cuando se desea hacer una mejora el sistema se debe de cuidar que en la búsqueda de mejora no se afecte al sistema; este capítulo describe algunas formas y características que son deseables que cuenten los sistemas; como la habilidad de intercambiar componentes sin apagar el sistema. De igual manera para ofrecer un buen servicio es necesario cuidar que los mensajes de error que se muestran proporcionen la información completa y claramente explicada (razón del error, sus consecuencias, como solucionarlo. Por último se recomienda la instalación de las últimas versiones de los paquetes, así como sus respectivos parches.

**Capítulo 11:** Técnicas especiales para la administración del sistema: Se presentan 4 requerimientos con los que debe de cumplir un sistema para que sea fácil de manejar: cada elemento debe de proporcionar información acerca de sí mismo, el elemento debe de realizar tareas que ayuden a la administración (tal como el registro de bitácoras), el elemento debe de comunicarse con una aplicación que ayude a administrarlo, y por último el componente debe de poder ser administrado remotamente. También describe varias aplicaciones y protocolos que facilitan la administración del sistema.

**Capítulo 12:** Todos Junto Ahora: Este capítulo solo hace énfasis en el uso de la aplicación de las disciplinas relacionadas con la administración de una manera efectiva.

En el apéndice A de éste libro se presenta una descripción de las características de algunos productos de los principales proveedores de tecnología en cómputo, se analizan sus características que los hacen ideales para ser utilizados en los sistemas de alta disponibilidad.



## Lectura ligera

Libro: Sever Load Balancing  
Autor: Tony Bourke  
Editorial: O'Reilly  
Primera edición  
ISBN: 0-5596-00050-2

Este libro se enfoca en la instalación, configuración y administración de un balanceador de carga como medio para lograr la disponibilidad continua. Este libro está organizado en 3 partes principalmente:

---

### EXPLICACIÓN SINTÉTICA DE CADA CAPÍTULO

---

**PARTE I** Conceptos y Teoría del balanceo de carga en servidores (Capítulos: 1-4): Esta primera parte el autor explica los conceptos de balanceo de carga. En el capítulo 1, habla de la evolución de la tecnología y el por que fue necesario su surgimiento. El capítulo 2 está dedicado a explicar los conceptos básicos como son: modelo OSI, redundancia, etc. El capítulo 3 se refiere a la anatomía de un balanceador de carga (como esta constituido). El último capítulo de esta primera parte trata temas relacionadas al rendimiento de un balanceador de carga.

**PARTE II** Configurando un balanceador de carga (Capítulos: 5-7): Esta segunda parte como su nombre lo indica muestra las maneras de implementar un balanceador de carga. En los capítulos 6-7 se implementan un balanceador flat-based y un NAT based respectivamente.

**PART III** Configurando balanceador de servidores (Capítulos 8-11): Se abarcan las principales características y funcionamiento de varios proveedores de balanceadores de carga, algunos proveedores de los que se analiza su producto son: Alteon websites, Cisco F5, etc.

## Tesis

Tesis: Sistemas de alta disponibilidad en equipos de cómputo.

Autores:

Amaro Ballesteros Raúl

Hernández Espinosa Héctor

Luna Sánchez Martín Alberto

Soltero Hario Ricardo

México 1998

Ubicación: Biblioteca Central, UNAM

Colocación: 001-01126-A4-1998-1

Esta es una tesis para obtener el grado de ingeniero en electricidad y electrónica que se encuentra en los anaqueles de la biblioteca central de la UNAM. Debido a que fue realizada para obtener el grado de ingeniero, esta tesis se enfoca en las características del hardware que se requiere para establecer un sistema de alta disponibilidad.

---

### EXPLICACIÓN SINTÉTICA DE CADA CAPÍTULO

---

Capítulo 1 Introducción: Aquí se aborda, primeramente, algunos puntos básicos acerca del tema de la tesis. Conceptos como *alta disponibilidad*, *sistemas abiertos*, etc. son definidos. También señala las causas por las que se provoca la interrupción del servicio, identifican los puntos de falla y el ambiente de trabajo de misión crítica.

Capítulo 2 Descripción de los elementos utilizados: El capítulo 2 explica cada elemento que conforma a un sistema de alta disponibilidad: unidades de almacenamiento, conexiones red, arreglos de discos, etc. Se explican las características eléctricas de cada componente y por que es conveniente utilizarlos en esta clase de sistemas.

Capítulo 3 Sistemas de alta disponibilidad propuestos: Este capítulo trata acerca de la propuesta que hacen los alumnos a los diferentes tipos de sistemas que se pueden utilizar. En la última parte del capítulo proponen configuración para algunos sistemas que los autores consideran útiles.

Tesis: CLUSTERING: Solución para alta disponibilidad en servidores con tecnología Intel

Autora: Ulloa Cuevas Elizabeth

México, 2000

Clasificación: 001-41132-U2-2000

Esta tesis fue escrita por una alumna de la carrera de ingeniería en computación y se enfoca a mostrar las consideraciones que se deben de tener presentes para implantar un sistema de alta disponibilidad, a través de la tecnología del clustering.

---

EXPLICACIÓN SINTÉTICA DE CADA CAPÍTULO

---

**Introducción:** En esta parte la autora plantea sus objetivos y justificación del tema; además se establece la hipótesis de trabajo.

**Capítulo 1 Tecnologías para la administración de la información:** Este capítulo trata algunos de los elementos que componen a un sistema de cómputo, empieza hablando a nivel general acerca de la evolución de los sistemas de cómputo: Desde la primera computadora digital electrónica hasta la cuarta generación de los microprocesadores. Posteriormente habla de las redes de computadoras, su clasificación y sus beneficios. Como tercer punto habla, igualmente de manera general, de la evolución de los medios de almacenamiento: discos duros, discos ópticos, arreglos de discos, etc. Por último hace referencia al cluster como una solución de alta disponibilidad, define algunos conceptos como disponibilidad, alta disponibilidad y tolerancia a desastres con el objetivo de que sea más clara la comprensión del tema.

**Capítulo 2 Clustering:** Aquí se presenta el tema del clustering y su aplicación en los sistemas de alta disponibilidad. Habla acerca de las causas por las cuales ocurren los fallos en el sistema y como solucionarlos. La parte más destacable de este capítulo es acerca de la arquitectura de un cluster ya que explica como esta constituido y cuales son los tipos de cluster que existen y como trabaja cada uno.

**Capítulo 3 El mercado de la tecnología del cluster:** En este capítulo se presentan algunos de los productos dedicados al ramo de la alta disponibilidad. Las compañías de las cuales presenta productos son Hewlett Packard y su producto HP NetServer, series LH Pro, LX Pro y LXr Pro. IBM con su NetFinity. UniSys cuya propuesta se basa en Windows NT. Y por ultimo habla de la alianza Compaq-Tandem-Digital.

**Capítulo 4 Caso práctico de instrumentación y administración de cluster:** En este capítulo se hacen algunas recomendaciones que hay que tomar en cuenta a la hora de construir un sistema de alta disponibilidad. Además de que proporciona una guía de configuración del sistema utilizando herramientas de la Cia.

Microsoft corp.

Creo que esta tesis es una buena fuente de información cuando se requiera implantar un sistema de alta disponibilidad basado en tecnología Intel y utilizando el sistema Windows NT.

## Artículos de revistas

**Artículo:** Servidor de archivos de alta disponibilidad.

**Autores:** Steve Blackmon y John Nguyen

**Revista:** Sys admin Magazine

**Fecha:** Septiembre 2001

**Volumen:** 10 **Número:** 9

En este artículo se explica el procedimiento para la instalación de un servidor de archivos, altamente disponible; los autores explican la forma de hacerlo utilizando herramientas libres tales como Linux, Samba y Heartbeat. En su artículo proponen la creación de un pequeño cluster de 2 nodos (uno maestro y otro de respaldo) para lograr esto.

Este artículo es completamente de carácter técnico ya que se requiere de conocimiento en el manejo de las herramientas para poder comprenderlo. Los autores no se detienen a explicar los conceptos básicos de la alta disponibilidad, simplemente aplican las herramientas.

**Artículo:** Configurando un servidor de alta disponibilidad con Red Hat

**Autor:** Rafeeq Ur Rehman

**Revista:** Sys admin Magazine

**Fecha:** Abril 2001

**Volumen:** 10 **Número:** 4

Parecido al artículo anterior, éste explica un método para lograr la alta disponibilidad en los servidores, sin embargo el servicio del que se trata es el web, igualmente se propone la creación de un cluster de 2 nodos y el uso de Linux Red Hat y Heartbeat.

## Sitios de Internet

Linux High Availability HOWTO

<http://www.biblio.org/pub/Linux/ALPHA/Linux-ha/>

Este manual se enfoca en el sistema operativo Linux como opción para establecer un sistema de alta disponibilidad.

Sistemas de alta disponibilidad bajo Linux

<http://www.linuxfocus.org/Castellano/November2000/article179.shtml>

Este artículo esta enfocado a mostrar el procedimiento para la instalación de un sistema de alta disponibilidad utilizando el sistema operativo Linux y Heartbeat. Es completamente técnico por lo que esta dirigido a personas con conocimientos en la administración de sistemas con Linux.

Alta disponibilidad para Linux

Autor: Juan Pedro Paredes

<http://congreso.hispalinux.es/congreso2001/actividades/ponencias/>

Este es un documento que habla acerca de los principios básicos de la alta disponibilidad. También menciona las características que hacen de Linux un sistema robusto como para ser utilizado en los sistemas de misión crítica. Por último menciona algunas tecnologías aplicables a este tipo de sistemas.

Es un documento muy pequeño que sólo ofrece una vista muy general de lo comentado arriba, si se quiere profundizar en el tema es necesario que se utilice la bibliografía del documento ya que éste no ofrece mucha información.

High Availability Linux Project

<http://www.linux-ha.org>

La meta de este proyecto es la de proporcionar una solución para la alta disponibilidad (a través de la tecnología del clustering) para Linux la cual

promueva la confiabilidad, disponibilidad, durabilidad a través del esfuerzo de desarrollo de la comunidad. En esta página se pueden encontrar, además varias referencias a otros sitios y artículos que tratan acerca del tema. Creo que este es un sitio obligado que una persona interesada en la alta disponibilidad con Linux debe de visitar para obtener información.

Red Hat High Availability Server Project

<http://ha.redhat.com>

Esta página proporciona información acerca de la alta disponibilidad utilizando Linux Red Hat. En este sitio se pueden encontrar, además, las últimas versiones de las herramientas que se emplean para proporcionar alta disponibilidad a un servidor.

HispaCluster

<http://www.hispacluster.org>

Este es un sitio donde se pueden encontrar noticias relacionadas con el súper cómputo, tiene secciones dedicadas a la alta disponibilidad en donde se pueden encontrar: noticias, manuales y referencias a otros sitios y por su puesto software especializado.

## Conferencias

### **Ventajas y desventajas del software libre**

**Conferencista:** Cristian Othon Martínez

**Evento:** Segundo Congreso de GNU/Linux

**Fecha:** 4 diciembre 2002

**Lugar:** Veracruz, Veracruz

En esta conferencia se vieron y analizaron las ventajas desventajas de utilizar el software libre, tanto a nivel académico como a nivel empresarial. Esta conferencia fue muy importante para la tesis debido a que pude reafirmar conceptos sobre el software libre y conocer sobre su aplicación no sólo en la vida académica sino también en las empresas. Lo expuesto en esta conferencia se encuentra plasmado en la sección referente al tema del software libre del MC.

## Capítulo 3

# MARCO CONCEPTUAL

*“Nunca he encontrado una persona tan  
ignorante que no se pueda aprender nada de ella.”*

**Galileo Galilei**

Para tener una idea más clara acerca de lo que significa el término “sistemas de alta disponibilidad” es necesario hacer un análisis etimológico de las palabras que componen este término.

## Definiciones

### Derivación etimológica

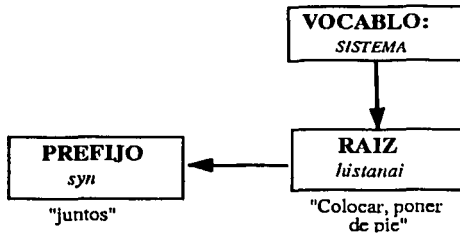
#### Sistema

La palabra sistema viene del latín “systema” y este a su vez del griego “systema”.

Derivaciones:

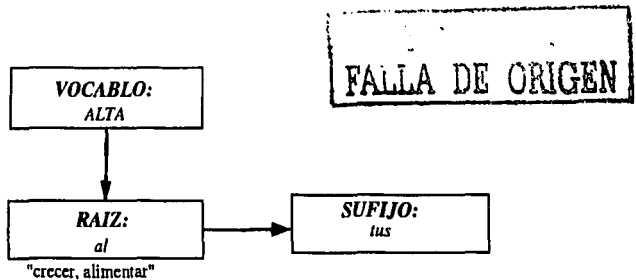
- sistemático
- sistemáticamente

- sistematización
- sistematizado



### Alta

La palabra alta (alto) del latín *altus* y éste del indoeuropeo *al-to* 'crecido'.



### Disponibilidad

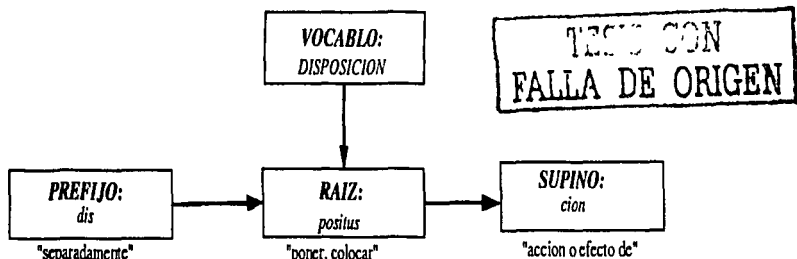
La palabra disponibilidad se puede considerar un derivado de la palabra disposición y ésta viene, del latín "*dispositonem*" acusativo de *disposito* (radical *disposition*). Disposición participio pasivo de *disponere* (disponer).



Derivaciones:

- disponible
- disponibilidad

Debido a la parte supina de la palabra disposición se puede considerar como la acción o efecto de colocar separadamente. La palabra **disponibilidad** significa en nuestra lengua: *situación o cualidad de disponible ó de que puede ser utilizado.*



## Definiciones del diccionario

*Real academia de la lengua española*

### Sistema

1. (Del latín *systema*). Conjunto de reglas o principios sobre una materia racionalmente entrelazados.
2. Conjunto de cosas que ordenadamente relacionadas entre sí, contribuyen a un determinado objetivo.

### Alta

1. Orden que da el médico al enfermo al declararlo oficialmente curado.
2. *Altamente*: Perfecta o excelentemente, en extremo en gran manera.

### Disponibilidad

1. Cualidad o condición de disponible, situación de disponible, referida a

funcionarios y militares.

### **Gran diccionario usual de la lengua española**

#### **Sistema:**

1. Conjunto ordenado de normas o procedimientos que contribuyen a un fin o conque funciona o se hace funcionar una cosa: *Sistema político; sistema educativo.*
2. Conjunto organizado de cualquier clase de cosas que se usan para un fin determinado: *sistema de signos; sistema monetario.*

#### **Altamente:**

1. En extremo, perfectamente: *un resultado altamente satisfactorio*

#### **Disponibilidad:**

1. Situación o cualidad de disponible o de que puede ser utilizado: *comprueba la disponibilidad de la suite antes de ofrecerla a los clientes.*
2. Situación laboral de quien no está en servicio activo, pero puede ser llamado a presentarlo, especialmente en el caso de los funcionarios.
3. Cantidad de dinero o bienes de los que se puede disponer: *mis disponibilidades no me permiten mucho gasto.*

## **Definiciones de autores**

**Peter S. Weygant** define a un sistema altamente disponible como un "sistema que está diseñado para evitar la pérdida del servicio mediante el uso de redundancia o el manejo de fallas, así como la minimización del tiempo planeado en que el sistema está fuera de servicio".

En la página del **IEEE Task Force on Cluster computing** se define la alta disponibilidad como sigue: "Se refiere a la disponibilidad de los recursos de un sistema de cómputo después de que ocurre alguna falla en el sistema. Esto puede ser logrado de varias maneras desde las soluciones que utilizan hardware redundante para asegurar la disponibilidad hasta otras que son soluciones basadas en software..."<sup>1</sup>

---

<sup>1</sup> <http://www.geocities.com/Paris/Musee/2712/tcfl.html>

**Ulloa Cuevas Elizabeth** da la definición de disponibilidad como: *“porcentaje de tiempo que un sistema de hardware está disponible en períodos operación (medidos en tiempo fuera o no planeado).”* Por lo que altamente disponible se referiría a tiempos de operación muy grandes.

Tomando de base las definiciones anteriores puedo definir a un sistema de alta disponibilidad como: *“Un sistema que, debido a su importancia, no admite tiempos de fuera de servicio y que para evitarlo hace uso de tecnologías y configuraciones especiales tanto de hardware como de software.”*

A continuación explico más a detalle esta definición:

- **Es un sistema:** Ya que esta conformado por otros elementos o subsistemas que en conjunto brindan mayor disponibilidad del servicio. Algunos subsistemas pueden ser: de almacenamiento, de red, comunicaciones, monitoreo, etc.
- **Debido a su importancia:** La importancia del servicio determinará el nivel de disponibilidad que el sistema puede permitir, ya que el uso de cierta arquitectura o tecnología depende de las necesidades que se deben de satisfacer para el sistema.
- **Uso de tecnologías y configuraciones especiales:** Un sistema de cómputo típico no es muy confiable cuando se trata de mantener funcionando un servicio de misión crítica; es por eso que se han creado tecnologías tanto de hardware como de software especialmente dirigidas a lograr la disponibilidad en los sistemas informáticos.

## Logrando la alta disponibilidad

La disponibilidad de un sistema se define como el tiempo en que este trabaja continuamente en condiciones favorables de desempeño. Para medir que tan disponible es un sistema se suele utilizar la siguiente formula:

$$A = \frac{MTBF}{MTBF + MTTR}$$

**A** = Availability - Disponibilidad

**MTBF** = Mean Time Before Failures - Tiempo de operación antes de fallas.

**MTTR** = Maximum Time To Repair - Tiempo máximo de reparación

Esta fórmula puede ser engañosa, ya que solamente considera fallos y no considera tiempos de respuesta altos, como ocurre cuando un sistema se satura por demanda excesiva.

En la tabla 3.1 se muestran diversos porcentajes de disponibilidad calculados en base a la fórmula anterior:

<b>Porcentaje de disponibilidad</b>	<b>Tiempo de fuera de servicio anual</b>	<b>Tiempo de fuera de servicio a la semana</b>
98 %	7.3 días	3 horas, 22 minutos
99 %	3.65 días	1 hora, 4 minutos
99.9 %	8 horas, 45 minutos	10 minutos, 5 segundos
99.99 %	52 minutos	1 minuto
99.999 %	5.25 minutos	6 segundos

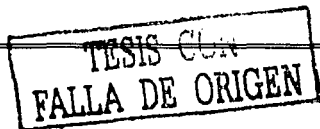
**Tabla 3.1:** Distintos porcentajes de disponibilidad

Peter S. Weegyant da algunas maneras para alcanzar la alta disponibilidad en los sistemas:

- Proporcionando la redundancia de los componentes
- Usando técnicas para el cambio de hardware y software
- Planeando cuidadosamente los tiempos de fuera de servicio (downtime)
- Eliminando la interacción de los humanos con el sistema<sup>2</sup>
- Definiendo respuestas automáticas a eventos de error
- Realizando pruebas completas a nuevos módulos que se le vayan a agregar al sistema
- Definiendo y practicando operaciones de respuesta a condiciones de error que no son manejadas automáticamente

Todas estas recomendaciones contribuyen en diferente medida a lograr la alta disponibilidad. En el presente trabajo se tratará más a profundidad el primer punto, debido a la naturaleza de la investigación; sin embargo en la medida de lo posible se

<sup>2</sup> Los humanos suelen cometer errores que afectan seriamente el servicio



mencionarán las demás.

## Tipos de sistemas de alta disponibilidad

No existe una clasificación oficial para los sistemas de alta disponibilidad; cada autor ofrece su propia clasificación un poco distinta respecto de los otros.

Peter S. Weygant	Floyd Piedad	Evan Marcus
<ul style="list-style-type: none"> <li>• Continuamente disponible</li> <li>• Tolerancia a Fallos</li> <li>• Tolerancia a desastres</li> </ul>	<ul style="list-style-type: none"> <li>• Alta disponibilidad</li> <li>• Operación continua</li> <li>• Disponibilidad Continua</li> </ul>	<ul style="list-style-type: none"> <li>• Disponibilidad regular</li> <li>• Disponibilidad incrementada: protección de datos del sistema</li> <li>• Recuperación de desastres</li> </ul>

**Tabla 3.2:** Distintas clasificaciones de los sistemas de alta disponibilidad

Creo que la mejor clasificación de estas tres es la que hacen Evan Marcus y Hal Stern debido a que hace una clara identificación de las características de cada nivel. Es por ello que a continuación se presenta:

### Nivel 1. Disponibilidad Regular

Es el nivel básico de "protección del sistema". No se toman medidas para la protección del sistema o fallas en los discos. Este nivel puede incluir respaldos, pero nada más, aquí se enfrentan los tiempos de interrupción del servicio como se vayan presentando, es decir sin ninguna planeación.



### **Nivel 2. Disponibilidad incrementada: Protección de datos**

Este nivel no es muy diferente del nivel 1, excepto por que incluye protección de datos. Esto significa emplear tecnología RAID; la mayoría de las veces esto significa el espejeo completo de discos. La falla de algún disco no conlleva a una pérdida de datos, debido a que los datos están almacenados en más de un disco. Los respaldos también protegen los datos contra la corrupción que pueda ser provocada por fallas.

### **Nivel 3. Alta disponibilidad: Protección del sistema**

Es comúnmente como alta disponibilidad, en esta configuración se toman dos servidores y se conectan, formando un *cluster*. De esta manera el servidor tiene duplicado cada componente del sistema, si algún componente falla, éste puede ser reemplazado automáticamente o manualmente por su duplicado. Aún ocurren las fallas en este nivel sin embargo la duración de éstas es muy limitada. Con este nivel se pueden conseguir una disponibilidad de un 99.98%, e incluso niveles mayores son posibles cuando protección adicional es implementada.

Este tipo de nivel es en el que se centra la presente tesis, debido a su "facilidad" de implementación y a que ofrece niveles de disponibilidad muy altos; además de que creó que pueda ser útil para sistemas de rango medio

### **Nivel 4. Recuperación de desastres: Protección de la organización**

La "recuperación de desastres" es el mayor nivel (y el más costoso), para la protección de un sistema. Cuando se implementa recuperación de desastres, se está protegiendo al sistema contra la pérdida completa del edificio, o el site, ya que se cuenta con un sitio alternativo ubicado a cierta distancia del site principal. Se necesita contar con todo el hardware necesario para operar los sistemas de reserva, junto con políticas y procedimientos necesarias para operar y efectuar el cambio del sitio que proporciona el servicio de manera rápida.

### **Sistemas Tolerantes a fallos**

Un sistema tolerante a fallos es un sistema de cómputo con hardware especializado que puede cubrir doble o triple redundancia de componentes en un solo sistema. Cuando un componente falla otro toma su lugar sin interrupción del sistema.

Es importante aclarar que la alta disponibilidad no necesariamente implica tolerancia a fallos, ya que si bien un sistema del Nivel 3 presenta un nivel de protección del sistema basado en la redundancia, los componentes de dichos sistemas pueden fallar y en ese momento otro nodo toma su lugar.

## **Causas de la caída de un sistema**

Existen muchas razones por las que un sistema puede fallar, a continuación se presentan las causas que pueden originar las fallas de los sistemas:

Cuando en un sistema existe algún componente cuya falla traería la indisponibilidad del servicio que se componente se le conoce como SPOF (por sus siglas en ingles) o punto único de fallo.

### **Hardware**

Los componentes que causarán la mayoría de las fallas son partes móviles, especialmente aquellas relacionadas con altas velocidades, tolerancias bajas y complejidad. Los discos son los primeros candidatos, ya que reúnen todas las características anteriores; además utilizan tarjetas controladoras y cables que pueden fallar. Los ventiladores son otros componentes con partes móviles, la falla de un ventilador puede no causar una falla de manera inmediata en el sistema, como la falla de un disco pero cuando esto ocurre, las consecuencias pueden ser impredecibles ya que cuando los procesadores o la memoria se sobrecalientan, estos pueden fallar de varias maneras.

Es por eso que los ventiladores y fuentes de poder presentan los peores MTBF's de todos los componentes del sistema. Las fuentes de poder pueden fallar resultando en la pérdida del servicio. La fallas en la fuente de poder pueden ser provocadas por varios factores incluyendo variaciones en la línea de voltaje y el desgaste que provoca las acciones de encender y apagar el equipo.

Para cubrir con estas deficiencias los equipos modernos cuentan con ventiladores, fuentes de poder extras y hardware especializado para diagnóstico.

### **Ambiente y Fallas físicas**

Existen elementos en el entorno que pueden ocasionar fallas, estos raramente son considerados; sin embargo afectan directamente al sistema. El elemento más obvio es la falla de la energía, esta puede venir desde la planta de energía o más

localmente una falla en alguna toma de corriente, o de algún fusible.

El aire acondicionado también puede fallar, provocando sobrecalentamiento en todos los equipos del centro de datos; de la misma forma sucedería sino se controlara el nivel de humedad del sitio.

Las plagas de roedores también representan un peligro para el funcionamiento del equipo. Existen otros elementos desde el fuego hasta el polvo que compromete la disponibilidad del servicio.

### **Fallas en la red**

Son varias las razones por las que la red puede dejar de funcionar: y estas van desde errores en el diseño físico de la red, hasta errores en la configuración. Entre las causas de la primera categoría se encuentran, fallas en los componentes físicos: *cables, switch, tarjetas de red*; respecto a los problemas de configuración se pueden encontrar: *errores en la configuración de los ruteadores, direcciones de IP duplicadas, etc.*

Aunado a esto si nuestra red esta conectada con alguna otra red externa de la cual no somos los responsables, esto incrementa el riesgo de fallo de la conexión entre el cliente y el servidor.

### **Fallos en los sistemas de bases de datos**

El corazón de la base de datos es el motor de la base de datos<sup>3</sup> Si este motor principal se detiene entonces todos los clientes que hagan uso de él dejarán de trabajar. Entre las fallas que puede presentar un sistema de bases de datos, es la insuficiencia de recursos, espacio en disco, memoria, etc. También existen problemas en la corrupción de los datos que puede ocurrir cuando el motor no ha terminado de realizar una transacción correctamente.

### **Fallas en los servidores web**

Existen varios lugares donde pueden surgir problemas, ejemplo de esto son los CGI's en Perl, Java, o código en Active/X que se encargan de servir paginas web. Si por alguna circunstancia un programa CGI se queda atorado en un ciclo infinito entonces la página nunca será entregada al cliente lo que muy posiblemente ocasionará que el cliente desista en su intento de acceder al sitio y se dirija a algún otro.

---

<sup>3</sup> La parte encargada de la lectura, escritura y ejecución de las consultas.

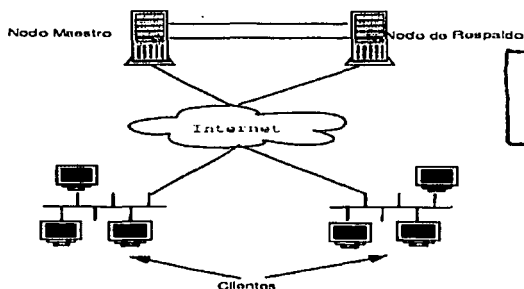


De igual modo cuando un servidor web se satura debido a una alta carga de peticiones hechas por parte de clientes, aquellos clientes cuyas peticiones no sean atendidas quedaran inconformes con el servicio y posiblemente se dirigirán a otro sitio.

Adicionalmente un servidor web esta expuesto a las fallas que pueden afectar a un sistema de computo tradicional; es decir, falla de componentes, corrupción en el sistema de archivos, etc.

## Clusters de alta disponibilidad

Un cluster se refiere a un grupo de computadoras (servidores) y otros recursos que están conectados a través de hardware, redes y software y se comportan como si fueran un sólo sistema (Ver figura 3.4).



TESIS CON  
FALLA DE ORIGEN

Figura 3.4: Ejemplo de un cluster muy simple

El principal objetivo de un cluster es hacer que un grupo de computadoras parezcan (ante el mundo exterior) como si fueran un sólo sistema totalmente integrando, siempre disponible y con gran rapidez. Hay varias ventajas que proporcionan el uso de clusters las cuales son:

- **Alta disponibilidad:** Esto es debido a que si falla un nodo otro nodo toma su lugar.
- **Balanceo de carga:** Balanceo de carga involucra la división del trabajo que un servidor tiene que hacer, entre 2 o más computadoras, así el trabajo es realizado más rápido. El balanceo de carga puede ser implementado con hardware, software o una combinación de ambas.
- **Procesamiento paralelo:** Para algunas aplicaciones, las instrucciones de los programas pueden ser divididas entre múltiples procesadores, así el tiempo total de ejecución de un programa es reducido drásticamente. Esta clase de cluster se utiliza en aplicaciones científicas o en algunas donde se requiera gran capacidad de procesamiento.
- **Administración del sistema:** La fácil administración de los servidores puede ser lograda a través del clustering. Ya que permite manejar un conjunto de sistemas como si fuera uno solo. Algunas características tales como: administración intuitivo, detección y recuperación de fallas y la habilidad para agregar y eliminar recursos del sistema sin la completa interrupción del servicio, son requeridas en los sistemas de administración de hoy en día.
- **Escalabilidad:** Se refiere a la habilidad de continuar con los niveles óptimos de desempeño, cuando se le agregan nuevos componentes (carga de procesamiento), de acuerdo como el crecimiento del negocio lo vaya exigiendo.

#### **Tipos de cluster de alta disponibilidad**

Existen 3 tipos básicos de configuraciones para clusters de alta disponibilidad que se pueden implementar:

**Activo / pasivo (Active/Standby):** Es aquel en donde se configura a un nodo para reemplazar al nodo que está corriendo la aplicación de misión crítica después de que ocurra una falla en el primero. En una configuración activo / pasivo dos o más nodos son conectados a los mismos discos que contienen los datos que nos interesan; cuando la primera aplicación falla entonces el segundo nodo empieza a proporcionar el servicio, además en esta clase de configuraciones el nodo de respaldo puede estar desocupado o posiblemente corriendo alguna aplicación menos importante.

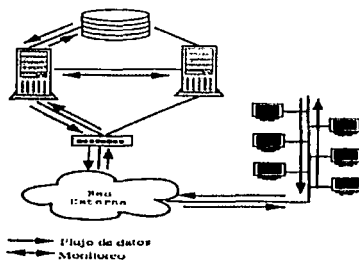


Figura 3.5: Configuración Activo / pasivo

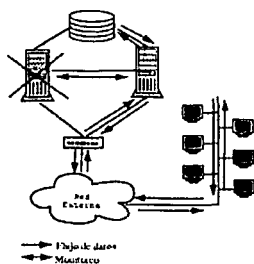
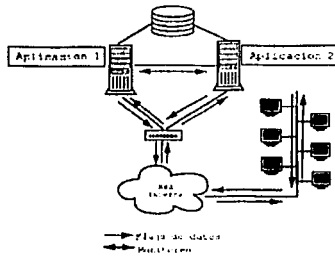


Figura 3.6: Configuración Activo / pasivo después de una falla

TESIS CON  
FALLA DE ORIGEN

**Activo / activo:** En este tipo de configuración dos o más nodos están físicamente conectados a los mismos discos de datos. Cuando la aplicación crítica falla en el sistema que está corriendo, entonces esta reinicia en otro sistema; en esta clase de configuración la aplicación puede correr en todos los nodos al mismo tiempo. En las figuras 3.7 y 3.8 muestra un ejemplo de este tipo de configuración donde el primer nodo se encuentra corriendo una aplicación crítica (aplicación 1) y el segundo está corriendo una aplicación de menor importancia (aplicación 2), al ocurrir el fallo la aplicación crítica es reiniciada en el segundo nodo.



TESIS CON  
FALLA DE ORIGEN

Figura 3.7: Configuración Activo / activo

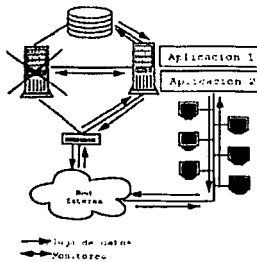
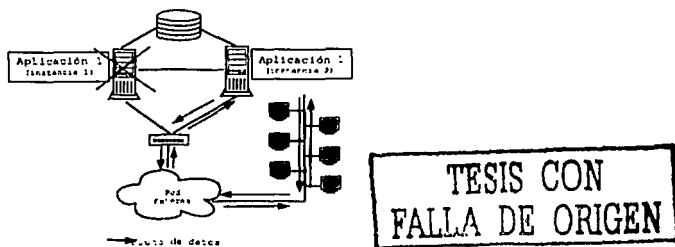
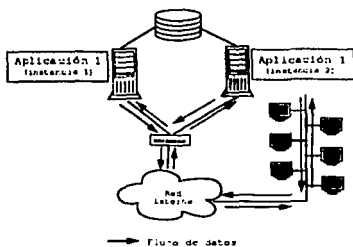


Figura 3.8: Configuración Activo / activo  
después de una falla

**Bases de datos paralelas:** Esta configuración consiste en un cluster en donde los diferentes nodos están corriendo la misma aplicación de bases de datos y todos acceden a la misma base de datos de manera concurrente (es decir al mismo tiempo). En esta configuración el fallo de un simple nodo no representa una gran pérdida, dado que los demás nodos se encargarán del trabajo. En las figuras 3.9 y 3.10 se muestra como es el funcionamiento de este tipo de configuración.



**Figura 3.9:** Configuración Base de datos paralela



**Figura 3.10:** Configuración Base de datos paralela después de una falla

## Componentes de un sistema de alta disponibilidad

### Sistemas de almacenamiento

Este componente es de suma importancia en el caso de los clusters de alta disponibilidad debido a que es el elemento que contiene la razón de ser de un sistema, la información que es recopilada, procesada e interpretada. Para asegurar a los sistemas de almacenamiento se pueden utilizar varias tecnologías:

- Arreglos de discos
- Sistema de archivos

- **Sistemas espejo**

### **Arreglos de discos**

Un arreglo de discos es un conjunto de discos que mediante algún tipo de redundancia aseguran la disponibilidad de los datos. Existen varias maneras de llevar a cabo los arreglos. Desde el básico que es aquel que utiliza dos discos conectados a un canal de datos en paralelo. Este arreglo con una realización ya sea en hardware o software asegura la disponibilidad de datos manteniendo copias de discos separados. En la actualidad existen arreglos mucho más sofisticados que utilizan múltiples redundancias y algoritmos realizados en hardware que permiten utilizar el uso del medio magnético.

Los arreglos de discos contrario a los independientes guardan la información en múltiples medios magnéticos. Los datos son distribuidos a través de los distintos discos. Utilizando varios discos los arreglos consiguen varias ventajas significativas sobre los discos sencillos e independientes:

- **Protección de datos:** Una parte de los discos del arreglo puede ser usada para guardar información, con la cual se recuperarán los datos en el momento de fallar uno de los discos. Estos mantienen el sistema funcionando y aseguran la integridad de la información aun cuando se presenta falla en uno de los discos.
- **Flexibilidad de configuración:** Los múltiples discos permiten una amplia variedad de formas de distribuir los datos a través de todo el arreglo. Esto permite maximizar el desempeño del sistema dependiendo de la carga de trabajo.
- **Incremento en la capacidad de almacenamiento:** Un arreglo de discos permite incrementar la cantidad de datos almacenados, al incrementar el número de discos conectados a una interfase.

### **RAID**

RAID son las siglas para Redundant Array of Independent Disk (Arreglo Redundante de Discos Independientes) y es el nombre genérico para una serie de algoritmos que se han establecido como un estándar industrial para la protección de datos en arreglos de discos. Existen varias categorías de RAID y son:

#### **RAID 0: Arreglo distribuido de disco sin tolerancia a fallo**

El RAID 0 permite el mejor desempeño y máxima capacidad sin protección de datos. En este nivel el controlador del arreglo escribe un bloque entero de datos a cada disco antes de continuar con el siguiente. El tamaño del bloque es definido por la profundidad de la distribución.

Las principales características de este nivel son:

- No-protección de datos
- Un muy alto índice de transferencia
- Una muy alta capacidad para concurrencia de entrada / salida.

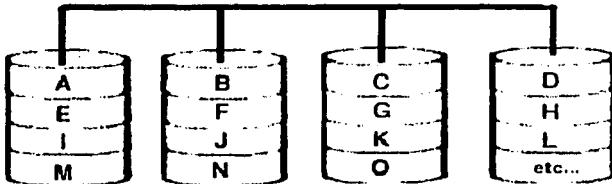


Figura 3.11: RAID 0

### RAID 1: Espejeo y Duplexing

El RAID 1 o par espejeado es un algoritmo en el cual se utiliza la copia en otro disco del arreglo para ofrecer la protección de los datos. En este tipo de implementación si un disco falla la información sigue disponible en el otro disco del arreglo. En este tipo de arreglo la protección de discos es máxima, pero la capacidad de almacenamiento se ve reducida en un 50%.

Las principales características de un arreglo en RAID 1 son las siguientes:

- Muy alta protección de datos
- Un índice muy alto de transferencia
- Alta capacidad para concurrencia de entrada / salida
- La capacidad del disco se ve reducida en un 50%

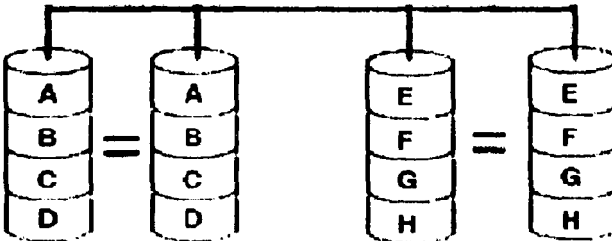


Figura 3.12: RAID 1

Existe un nivel llamado RAID 0+1 que une las características del RAID 0 y el RAID 1, es decir proporciona protección de fallos además de un gran desempeño en lectura y escritura.

**RAID 2:** Acceso paralelo con discos especializados. Redundancia a través del código Hamming

El RAID nivel 2 adapta la técnica comúnmente usada para detectar y corregir errores en memorias de estado sólido. En un RAID de nivel 2, el código ECC (Error Correction Code) se intercala a través de varios discos a nivel de bit. El método empleado es el Hamming. Puesto que el código Hamming se usa tanto para detección como para corrección de errores (Error Detection and Correction), RAID 2 no hace uso completo de las amplias capacidades de detección de errores contenidas en los discos. Las propiedades del código Hamming también restringen las configuraciones posibles de matrices para RAID 2, particularmente el cálculo de paridad de los discos. Por lo tanto, RAID 2 no ha sido apenas implementado en productos comerciales, lo que también es debido a que requiere características especiales en los discos y no usa discos estándares.

No existe una implementación comercial para este nivel de RAID

**RAID 3:** Transferencia paralela con paridad

El RAID 3 ofrece protección de datos con alta transferencia. Este nivel utiliza una técnica llamada *distribución de byte (Byte striping)* la cual distribuye la información en los discos byte por byte. Distribuyendo los datos de esta manera implica que todos los discos del arreglo se ven involucrados en cada transacción de datos. Este tipo de implementación ofrece un muy alto índice de transferencia ya que todos los discos trabajan en una transacción, pero ofrece un bajo nivel ante la concurrencia de entrada / salida, debido a que todos los elementos están ocupados sirviendo sólo a una transacción a la vez.

Las principales características de un arreglo RAID 3 son las siguientes:

- Alta protección de datos
- Alto índice de transferencia
- Baja capacidad para concurrencia de entrada / salida
- La capacidad del disco se ve reducida en mecanismo.



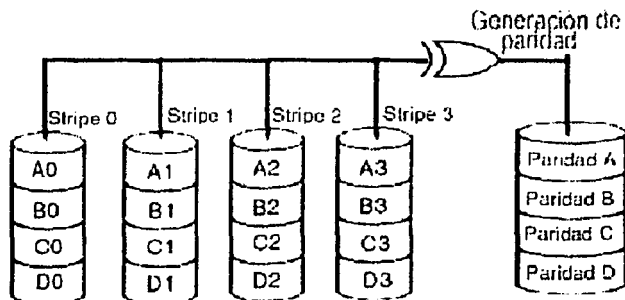


Figura 3.13: RAID 3

#### RAID 4: Acceso Independiente con un disco dedicado a paridad

Basa su tolerancia al fallo en la utilización de un disco dedicado a guardar la información de paridad calculada a partir de los datos guardados en los otros discos. En caso de avería de cualquiera de las unidades de disco, la información se puede reconstruir en tiempo real mediante la realización de una operación lógica de O exclusivo. Debido a su organización interna, este RAID es especialmente indicado para el almacenamiento de ficheros de gran tamaño, lo cual lo hace ideal para aplicaciones gráficas donde se requiera, además, fiabilidad de los datos. Se necesita un mínimo de tres unidades para implementar una solución RAID 4. La ventaja con el RAID 3 está en que se puede acceder a los discos de forma individual.

#### RAID 5: Discos de datos independientes con bloques de paridad distribuidos

El RAID 5 ofrece una alta capacidad de protección de datos con una alta capacidad de concurrencia de entrada / salida. Guarda información en forma de bloques a través de todos los discos del arreglo, así como la información de paridad esta distribuida en todos los elementos. El controlador del arreglo escribe un bloque de información en un disco antes de continuar con el siguiente. El RAID 5 utiliza distribución de bloques para distribuir datos a través de los discos del arreglo.

Las principales características de un arreglo RAID 5 son:

- Alta protección de datos

- Alto índice de transferencia durante lecturas y bajo durante escrituras
- Una alta capacidad para concurrencia de entrada / salida
- La capacidad en disco se ve reducida en un 20%

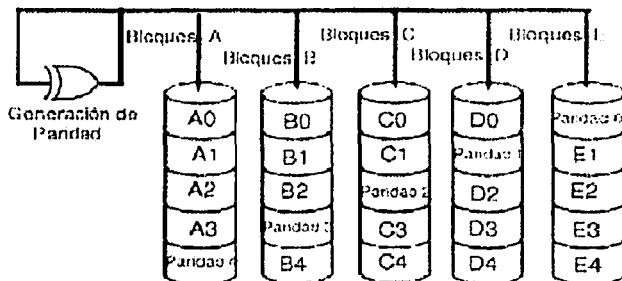


Figura 3.14: RAID 5

Estos no son los únicos niveles de RAID que existen sin embargo, son los que proporcionan mayor utilidad para la clase de sistemas que esta tesis trata.

TESIS CON  
FALLA DE ORIGEN

Nivel RAID	Característica principal	Aplicaciones en las que se puede utilizar	Discos necesarios para implantarlo
RAID 0	Distribución de los datos sin tolerancia a fallos.	<ul style="list-style-type: none"> <li>• Producción y edición de video e imágenes.</li> <li>• Cualquier aplicación que requiere alto ancho de banda.</li> </ul>	2
RAID 1	Espejo de discos	<ul style="list-style-type: none"> <li>• Contabilidad</li> <li>• Nomina</li> <li>• Cualquier aplicación que requiere alta disponibilidad</li> </ul>	2
RAID 2	Acceso paralelo con discos especializados. Redundancia a través del código Hamming	No existe implementación comercial para este nivel	No existe implementación comercial para este nivel
RAID 3	Transferencia en paralelo con paridad	<ul style="list-style-type: none"> <li>• Producción de video y transferencia de video en vivo</li> <li>• Edición de imagen y video</li> <li>• Aplicaciones en las que se requiera alta tasa de transferencia</li> </ul>	3
RAID 5	Discos independientes con bloques de paridad distribuidos	<ul style="list-style-type: none"> <li>• Servidores de archivos y aplicaciones</li> <li>• Servidores de bases de datos y servicio web</li> </ul>	3

Tabla 3.3: Cuadro resumen de la tecnología RAID

TESIS CON  
 FALLA DE ORIGEN

## Sistema de archivo

Un sistema de archivos es el software que se encarga del manejo (creación, lectura, escritura y eliminación) de las unidades de información conocidas como archivos; el sistema de archivos es la parte del sistema operativo que provee todos los mecanismos para lograr que la información almacenada en el disco duro pueda ser usada por varios procesos al mismo tiempo, también es el responsable de poder almacenar gran cantidad de datos, además de que se encarga de la gestión y mantenimiento del aspecto físico del almacenamiento de los datos. Esto incluye la asignación de espacio a nuevos archivos y el mantenimiento de la información sobre el espacio libre. Ejemplos de sistemas de archivos son: *jfs*, *ext3*, *reiserfs*, *xfs*

### JFS Journaling File System (Sistema de archivos con registro)

JFS proporciona consistencia estructural mejorada, recuperabilidad y tiempos mucho más rápidos de recuperación que sistemas sin journaling tales como HPFS, ext2 y los sistemas de archivos tradicionales de UNIX. JFS utiliza técnicas originalmente desarrolladas para las bases de datos para registrar información acerca de las operaciones realizadas en los meta datos del sistema de archivos como si fueran transacciones atómicas. En caso de falla del sistema, el sistema de archivos es restaurado a un estado consistente mediante la reejecución de las bitácoras y aplicando los registros de las bitácoras para las transacciones apropiadas. El tiempo de recuperación asociado con este enfoque basado en registros es mucho menor debido a que el programa de reparación sólo necesita examinar los registros de las bitácoras producidos por la actividad reciente del sistema de archivos, en lugar de examinar todos los meta datos del sistema de archivos.

Este es a nivel general el funcionamiento del JFS. Sin embargo existen otros sistemas de archivos que desempeñan tareas similares:

### EXT3

El sistema operativo Linux tiene soporte para varios tipos de archivos, sin embargo tiene como sistema de archivos nativo el llamado *ext2*. Fue diseñado por Remy Card, y tiene entre sus características:

- Soporta los siguientes tipos de archivos: *archivos regulares*, *directorios*, *archivos especiales*, *enlaces simbólicos*.
- Pueden asociarse nuevas propiedades a los archivos para modificar el comportamiento del núcleo.
- Al igual que los sistemas de archivos tradicionales en el UNIX, éste tiene la

organización de un árbol.

El sistema de archivos ext2 se compone de varias partes como son:

- Superbloque
- Grupos de bloques

El sector de arranque contiene el código de máquina necesario para cargar el núcleo en el arranque del sistema y cada uno de los grupos de bloques se descomponen a su vez en varios elementos que son:

- Superbloque: Contiene información para el control del sistema de archivos y se duplica en cada grupo de bloques para permitir arreglar fácilmente una corrupción en el sistema de archivos.
- Descriptores: Contienen las direcciones de los bloques que contienen información crucial como los bloques de bitmap y la tabla de i-nodos.
- Bitmap de bloques: Se utiliza para saber si un bloque está asignado. A cada bloque del grupo se le asocia un bit (contendrá 1 cuando el bloque este asignado y 0 cuándo no lo este)
- Tabla de i-nodos: Estos bloques contienen información de la localización de los archivos en el disco.
- Bloque de datos: Es el resto de los bloques se utiliza para almacenar los datos contenidos en los archivos.

Pues bien se podría decir que el sistema de archivos ext3 posee las mismas características que el sistema de archivos ext2 sólo que con journaling (es decir manejo de bitácoras). Este sistema de archivos has sido integrado al kernel de Linux desde la versión 2.4, es 100% compatible con ext2 de tal manera que puede hacer uso de aplicaciones que hayan sido desarrolladas para él.

### **ReiserFS**

Este es otro sistema de archivos originalmente desarrollado por Hans Reiser. ReiserFS trata a la partición del disco como si fuera una tabla de una base de datos. Directorios, archivos, meta datos son organizados en una estructura eficiente de datos llamada "*Balanced tree*" (árbol balanceado). Esto difiere un poco de la manera en que otros sistemas de archivos trabajan, sin embargo, ofrece mejoras en la velocidad de acceso para algunas aplicaciones, especialmente aquellas que utilizan una gran cantidad de archivos pequeños.

La lectura y escritura de archivos de gran tamaño (tales como imágenes de CDROM) es con frecuencia limitada por el hardware (velocidad del disco duro, o los canales I/O); pero cuando se trata de archivos pequeños, como scripts, es con frecuencia

limitada por el diseño del sistema de archivos. La razón de esto es que para abrir un archivo el sistema requiere: primero, localizar el archivo, y eso significa leer directorios en el disco, además, el sistema necesita verificar, por seguridad, los meta datos para ver si el usuario tiene permisos de acceder al archivo y eso significa lecturas adicionales en el disco.

ReiserFS utiliza sus "árboles balanceados" para modernizar el proceso de búsqueda de los archivos y verificación de seguridad en los meta datos. Para archivos extremadamente pequeños, los datos pueden ser almacenados físicamente cerca de sus meta datos; de esta manera ambos pueden ser leídos juntos con solo un poco o ningún movimiento del mecanismo de búsqueda del disco duro. Si alguna aplicación requiere abrir muchos archivos rápidamente, entonces este sistema mejora significativamente el desempeño.

Otra característica de este sistema es que no maneja el enfoque tradicional en el manejo del espacio (como ext2 lo hace). En ext2 el espacio es asignado en bloques que van desde 512 hasta 4096, sin embargo esto ocasiona que el espacio en disco sea desperdiciado, por ejemplo supongamos que tenemos un sistema que está organizado en bloques de 1024 bytes y se necesita almacenar un archivo de 8195 bytes, entonces se necesitarían 8 bloques para almacenar 8192 bytes por lo que se necesitaría un noveno bloque para almacenar los 3 bytes restantes, esto implicaría que ese está desperdiciando prácticamente un bloque completo. ReiserFS utiliza la estructura del árbol para mantener el conteo exacto de los bytes que se necesitan para almacenar algún archivo.

Algunas aplicaciones se benefician más que otras de este tipo de optimización. Por ejemplo un directorio con cientos de archivos en formato PNG ó GIF que son utilizados como iconos en páginas web de un sitio muy visitado, esta situación está echa a la medida para el sistema ReiserFS. Del mismo modo un sitio web con miles de archivos html, cada uno, con unos cuantos kilobytes de tamaño es un excelente candidato.

Además de todo esto ReiserFS es un sistema de archivos con journaling como xfs, ext3 y jfs, por lo que también proporciona rapidez en la recuperación de los sistema en caso de que haya ocurrido una falla.

## **XFS**

XFS fue desarrollado por Silicon Graphics, es un sistema de archivo de alto desempeño con soporte para arquitecturas de 64 bits. Entre sus características se encuentran el manejo de grupos; cuando se crea un sistema de archivos tipo XFS se crean 8 o más regiones en el dispositivo, a estas regiones se les conoce como grupos cada grupo maneja su tabla de i-nodos y espacio libre, son algo así como subsistemas de archivos dentro de un sistema tipo XFS. De ésta manera se hace

más eficiente el manejo de I/O ya que es en forma paralela, esto es debido a que cada grupo es efectivamente su propia entidad, de esta manera el kernel puede interactuar con varios grupos al mismo tiempo, así pueden haber varios procesos de acceso al disco corriendo al mismo tiempo. Si XFS no manejara los grupos entonces el sistema utilizaría un esquema lineal para hacer modificaciones en la tabla de inodos.

Este sistema de archivos también tiene journaling permitiendo la rápida recuperación del sistema en caso de fallas. XFS permite que la bitácora sea almacenada en otro dispositivo, ya sea otra partición u otro disco. Está característica funciona bien para mejorar el desempeño del sistema de archivos aun más.

### Duplicación de los datos

El espejeo o *Data Mirroring* es una técnica que consiste en hacer que dos (o más) computadoras contengan exactamente los mismos datos, de manera que no importa a que servidor accese el cliente, siempre parecerá que está trabajando en la misma máquina. También se puede utilizar para realizar respaldos completos de equipos de tal manera que el equipo de respaldo pueda entrar a funcionar inmediatamente después de la falla del equipo primario.

El data mirroring (espejeo de datos) es una forma de eliminar el SPOF que representa tener unidades de almacenamiento compartido (discos standalone), ya que al estar los datos disponibles en más equipos entonces se logra una distribución de carga, y redundancia de los datos.

Entre las herramientas que se utilizan para hacer replicación de datos en "tiempo real" se encuentran:

**FAM:** File Alteration Monitor o Monitor de alteración de archivos, provee un API que las aplicaciones pueden usar para ser notificadas cuando archivos o directorios específicos han cambiado. FAM viene en dos partes: FAM, el "demonio" que escucha las peticiones y envía las notificaciones, y libfam, una librería que la aplicación cliente puede usar para comunicarse con FAM.

**IMON:** Inode Monitor o Monitor de Inodos, es la parte del kernel que le dice a FAM cuando un archivo ha cambiado. Cuando las aplicaciones le dicen a FAM que están interesadas en archivos o directorios, FAM pasa esos intereses a IMON.

Estas dos herramientas están liberadas bajo la licencia GPL, lo que las hace adaptables a las necesidades de quien desee utilizarlas.

## Software Libre

Uno de los objetivos de ésta tesis es la de proponer el uso del software libre para brindar un nivel mayor de protección a los sistemas contra la falla de alguno de sus componentes. Sin embargo antes de eso es necesario explicar de manera clara que es el software libre y por que ha ganado tanta popularidad recientemente.

### ¿Qué es el software libre?

En el sitio del proyecto GNU (GNU is Not UNIX) definen al software libre de esta manera: *“software libre se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, cambiar, y mejorar el software...”*

La palabra libertad en el contexto del software libre se refiere a 4 tipos de libertades

- 1.La libertad de usar el programa con cualquier propósito. (libertad 0)
- 2.La libertad de estudiar cómo funciona el programa y adaptarlo a tus necesidades (libertad 1). El acceso al código fuente es una condición previa para esto.
- 3.La libertad de distribuir copias, con lo que puedes ayudar a tu vecino (Libertad 2)
- 4.La libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie (libertad 3). El acceso al código fuente es un requisito para esto.

Existe cierta confusión con el término software libre o “free software” debido a que la palabra inglesa *free* tiene un doble significado “libre” y “gratis” es debido a esto que se cree que el software libre es al mismo tiempo por obligación gratuito, lo cual es falso ya que nada impide que se pueda vender el software.

### ¿Cómo surgió?

En 1983 un miembro del laboratorio de inteligencia artificial del MIT<sup>4</sup> llamado Richard Stallman envió un correo a un grupo de noticias net.unix-wizards en donde anunciaba el inicio de su proyecto llamado GNU, el objetivo era el de realizar un sistema operativo basado en UNIX pero con algunas mejoras; Y ponerlo a disposición de quien le pudiera ser útil.

En 1984 Richard Stallman renunció al MIT y comenzó a trabajar en el proyecto GNU,

---

<sup>4</sup> Massachusetts Institute of Technology - Instituto Tecnológico de Massachusetts



al descubrir que lo primero en lo que tenía que empezar a trabajar era en un compilador, se dedico a desarrollar lo que con el tiempo sería el compilador GNU CC (GCC), anteriormente ya había trabajado en un editor de texto llamado EMACS (Editor de Macros), que estaba basado en LISP.

En 1985 se fundó la FSF, Free Software Foundation (Fundación del software libre) que está dedicada a promover entre los usuarios de computadoras, software libre, es decir programas con derecho de uso, copiado, modificación y redistribución.

Con la misma filosofía en mente ha surgido otro movimiento parecido al del software libre denominado *opensource* (Fuente abierta). En ocasiones se llega a nombrar al software libre como *opensource* y viceversa.

## Ventajas y desventajas del uso del Software Libre

El uso de el software libre implica ciertas ventajas y desventajas que a continuación se mencionan:

### Ventajas

#### □ Ventajas Académicas:

- **Aprendizaje:** Uno de los principales objetivos que persigue la filosofía del software libre, es el permitir el aprendizaje mediante el análisis del código fuente del programa. Es posible mediante el acceso al código fuente, como funciona internamente un sistema operativo, un lenguaje de programación o un servidor web.
- **Económico:** Al ser de libre distribución, el software libre puede ser utilizado en varios aspectos de la vida académica: docencia, investigación, etc., sin tener que pagar grandes cantidades de dinero por concepto de licencias.
- **Modificación de programas:** Las licencias libres permiten la modificación del código fuente con el fin de mejorarlo o adaptarlo a las necesidades específicas del usuario. De esta manera se puede experimentar, con fines de investigación, con algoritmos que alguien más haya programado.
- **Desarrollo en equipo. (Comunidad):** Las herramientas de software libre son el resultado del trabajo conjunto de programadores de todo el mundo; esto permite tener contacto con diferentes metodologías de programación y métodos de análisis.
- **Herramientas de desarrollo poderosas y flexibles:** Las herramientas de desarrollo del software libre, constantemente se están modificando con el

fin de agregarle mayor funcionalidad y corregir errores.

- Empresariales
  - Oportunidad de negocio en empresas de tecnología: Las empresas pueden ofrecer nuevos productos basados en el SL; lo que permite el aprovechar el desarrollo que otras personas han realizado.
  - Servicios basados en software libre: Las empresas dedicadas a la consultoría pueden ofrecer soluciones basadas en SL, lo que le permitiría obtener beneficios por los servicios de instalación y (en caso de que se requiera) modificación del software ya disponible, sin tener que cargar al cliente costos de licencias, lo que les permitiría crear soluciones más atractivas para los clientes.
  - Adecuaciones para necesidades individuales: Se puede cobrar a empresas por adecuarles software libre a sus necesidades (ya que ambos, el derecho de cobro y el derecho de modificación) están permitidos. De igual manera las compañías se pueden ahorrar dinero si modifican a sus necesidades software ya desarrollado, en lugar de invertir en el desarrollo completo de su software.
  - Reducción de costos: Las empresas pueden ahorrarse grandes cantidades de dinero por concepto de pago de licencias, el cual pueden utilizar para invertirlo en actualizaciones de equipo o capacitación de personal.
  - Software robusto, serio y confiable: El permitir que varias personas participen en su desarrollo, ocasiona que el producto final sea funcional, casi libre de errores y con suficiente documentación.
  - Ahorro de recursos de cómputo: Debido a que el SL fue diseñado para explotar al máximo los recursos de hardware; no se necesita adquirir nuevo equipo para poder instalarlo, ya que se puede utilizar aquel con el que cuenta la empresa.
  - Soporte técnico por parte de la comunidad: Es posible obtener consejos y resolver problemas que enfrentamos, mediante el contacto (generalmente, mediante listas de correo) con otros usuarios que hayan experimentado los mismos problemas.
  - Soporte técnico por parte de los desarrolladores: Es posible tener contacto (generalmente mediante listas de correo) con los desarrolladores del software que utilizamos, para obtener respuestas a los problemas que enfrentamos.
- Gubernamentales
  - Seguridad (En los datos).
  - Confidencialidad.
  - Acceso a la información.
- Personales
  - Formato de archivo de datos abiertos.
  - Sin restricción en la copia y distribución de información y contenidos.
  - Independencia de proveedor.

## Desventajas

- **Curva de aprendizaje:** Se requiere más tiempo para aprender a utilizar herramientas libres, ya que en muchos casos las interfaces de usuario no resultan tan intuitivas como las de los productos comerciales.
- **La transición no es transparente:** Debido a que pueden existir ciertas incompatibilidades que habrá que resolver antes de hacer la transición; además de que es preferible de que esta se haga por etapas y no completamente de un día para otro. Otra cosa que hay que considerar es la renuencia al cambio por parte del usuario.
- **Soporte técnico en ocasiones irregular:** Si no se contrata a alguna compañía que brinde soporte técnico dedicado, se está a expensas de que los demás miembros de la comunidad (usuarios, desarrolladores) brinden el soporte; sin embargo, ninguno de ellos está obligado a hacerlo, lo que en ocasiones lleva que no se obtenga una solución (o al menos no en un corto tiempo) al problema que se nos presenta.
- **Muchas opciones. (A veces es difícil escoger):** Existen en ocasiones muchas aplicaciones que realizan la misma función y es necesario probar alguno de ellos para poder decidir cual hace mejor su trabajo.
- **Faltan algunas aplicaciones clave:** Algunas aplicaciones que por ser muy especializadas aun no tienen un grado de desarrollo (o inclusive no han sido de desarrolladas) suficiente para ser utilizadas; ejemplo: aplicaciones relacionadas con impuestos y puntos de venta.
- **Faltan desarrolladores:** Mientras el número de usuarios del SL esta incrementándose rápidamente, aun son pocas personas las que se dedican a su desarrollo.

## Licencias

Existen muchos tipos de licencias bajo las cuales puede ser distribuido el software libre, cada una de ellas, define sus propios términos de copia, modificación y distribución. A continuación se describen los tipos de licencias más importantes:

### **GPL General Public License (Licencia Pública General)**

Es una de las licencias más conocidas (sino es que la más) bajo la cual puede licenciarse software libre. Fue creada para garantizar la libertad, de los usuarios, para compartir y cambiar software, y que éstas libertades continúen para todos los usuarios. Es necesario aclarar que esta licencia es muy extensa en su contenido (a diferencia de otras), ya que abarca todas las situaciones bajo las cuales puede ser

utilizado el software libre. Entre los términos de esta licencia están:

1. Se puede copiar y distribuir copias del código fuente del programa, tal y como se recibió, por cualquier medio, manteniendo en cada copia una transcripción de la licencia GPL y una declaración de garantía. Se permite hacer un cargo por la transferencia física del programa; y a opción se puede ofrecer garantía (por una comisión) por el intercambio.
2. Se permite modificar la copia recibida o la copia de 3ros o una parte del programa guardando las siguientes condiciones:
  1. Agregar una nota en los archivos modificados donde se mencione que ese archivo fue modificado del original.
  2. Cualquier trabajo que contenga partes de este programa deberá de ser licenciado bajo los términos de la GPL.

### **Licencia MIT**

Por medio de esta licencia se otorga a cualquier persona, que haya obtenido una copia de un programa bajo esta licencia, permiso para su libre distribución, uso, copia, modificación, combinación, sub-licenciamiento o venta bajo las siguientes condiciones:

1. Se debe de conservar el aviso de copyright, en todas las copias, (o porciones de código) que sean redistribuidas.
2. Igualmente el software se otorga sin ninguna garantía de ningún tipo
3. Ni los poseedores de la licencia, ni los autores del programa son responsables por los daños puedan ser ocasionados por el uso del software.

### **Licencia BSD**

Esta licencia es equivalente a la anterior, con la particularidad de que a esta se le agrega una cláusula:

1. Ni el nombre de la <ORGANIZACIÓN> ni el nombre de los colaboradores pueden ser utilizados para vender o promover, productos derivados de este software.

### **Licencia de Apache**

Con esta licencia se libera software que fue desarrollado por la Apache Software Foundation, y por medio de ésta se permite el libre uso y redistribución ya sea con o sin modificaciones siempre y cuando se cumplan los siguientes términos.

1. Se debe de conservar el aviso de copyright, la lista de términos de uso y el

aviso de no garantía, ya sea en el código fuente o en la documentación.

2. La documentación para el usuario deberá de contener la siguiente nota: "This product includes software developed by Apache Software Foundation (<http://www.apache.org/>)".
3. Los productos derivados del software no se deberán llamar "Apache", ni contener la palabra "Apache" si no es con permiso escrito. Así como tampoco deberán de ser usados para promover productos derivados del software.
4. Al igual que en las licencias anteriores aquí tampoco se ofrece ninguna garantía de ningún tipo.

El apéndice B contiene la lista completa de licencias consideradas opensource.

## Software para proteger el sistema

Como se mencionó anteriormente existe software que es distribuido bajo una licencia opensource, el cual ha sido desarrollado para brindar protección a los sistemas de cómputo. A continuación se presenta la descripción de algunos de estos programa.

### Heartbeat

Esta herramienta sirve para crear un pequeño cluster de dos nodos basado en el intercambio de paquetes o heartbeats (latidos de corazón). La manera en que funciona es la siguiente:

Con Heartbeat instalado en las dos maquinas, una debe de funcionar como el nodo que proporciona el servicio (nodo principal) mientras que la otra está monitoreandolo mediante el envío de los heartbeats, cuando el segundo nodo nota que el nodo principal ha fallado entonces éste nodo de respaldo realiza un takeover (que consiste en obligar al otro nodo a ceder el servicio y se hace un intercambio de IP), de esta manera el servicio no sufre interrupción.

Existen varias maneras mediante las cuales Heartbeat puede enviar los heartbeats:

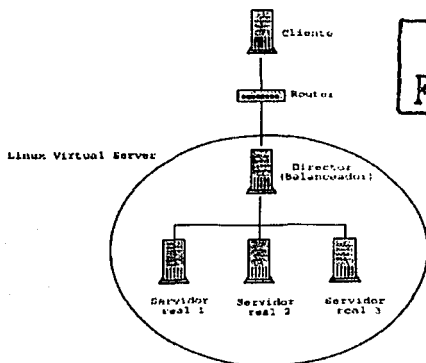
- **heartbeats en anillo:** En esta clase de envío los nodos se encuentran organizados en forma de anillo, cada nodo envía un heartbeat a su vecino, para monitorearlo. En esta configuración se utiliza el puerto serial para enviar los paquetes y los nodos son interconectados mediante un cable conocido como "null modem", dependiendo del número de nodos que conformen el cluster se requerirán 1 o 2

- puertos seriales, (1 cuando se trata solo de 2 nodos y 2 cuando son más).
- heartbeats en broadcast Aquí se utilizan otros medios para enviar los paquetes, tal como tarjetas de red, o envío de señales en infra rojo. En los casos donde el número de nodos del cluster es grande, esta configuración es mucho más sencilla de cablear que la configuración en anillo.

Lo mejor de esta herramienta es que está bajo la licenciada GPL por lo cual el acceso al código está permitido, así como su libre uso y distribución. Además de que al estar bajo un esquema de desarrollo comunitario tiene mayores probabilidades de mejoras.

### LVS Linux Virtual Server

Un servidor virtual de Linux, es un cluster de computadoras (servidores reales) que ante el mundo exterior parecen ser un solo servidor, ésta apariencia de ser un solo servidor es llamado *servidor virtual*, estos servidores reales están bajo el control de otro servidor cuya función es la de realizar el balanceo de la carga. Este último servidor (el que balancea la carga) debe de correr una versión modificada del kernel Linux, esto es para darle la capacidad al servidor de correr el código *ipvs*.



TESIS CON  
FALLA DE ORIGEN

Figura 3.15: Configuración del LVS

LVS es una buena opción debido a lo siguiente:

- El costo de incrementar el tráfico mediante la agregación de otro servidor real en

LVS es lineal, mientras que este costo se incrementa más rápidamente cuando se trata de un sólo servidor de gran capacidad.

- Por redundancia. Cada maquina puede ser removida del cluster que forma al servidor virtual, para actualizaciones, mantenimiento o incluso para cambiar el sitio donde se encuentran, y regresada al cluster nuevamente sin que haya una interrupción del servicio.
- Por su adaptabilidad. Si la demanda de servicio crece o decrece (de acuerdo como el crecimiento de la organización lo vaya requiriendo), de una manera muy fácil, pueden ser agregados (o eliminados) del cluster, más maquinas que proporcionen el servicio.

Cuando los clientes se conectan a un servicio que utiliza LVS, éste ve solo una dirección IP y cree que está conectado a una sola maquina, debido a que las otras maquinas utilizan direcciones IP privadas y el balanceador se encarga de redirigir las conexiones a otras, por lo que el cliente puede estar conectado a una maquina en una ocasión y en una conexión posterior podrá utilizar otra maquina. Además los servidores creen que ellos están conectados directamente con el cliente; es decir el hecho de que exista un servidor intermedio que este repartiendo la carga no afecta ni al cliente ni al servidor.

Este software esta liberado bajo licencia GPL.

## **MONIT**

Es una herramienta para monitorear programas llamados demonios (en sistemas UNIX), es capaz de iniciar programas que no estén corriendo o reiniciar aquellos que no estén respondiendo. Esto es de gran ayuda para automatizar la tareas de repuesta a errores ya que cuando una de nuestras aplicaciones falle entonces MONIT será capaz de reiniciarla y reportar el error al administrador.

Las herramientas presentadas en esta apartado no son todas las existentes ni tampoco necesariamente las mejores; sin embargo son de gran utilidad para ser utilizadas en sistemas de alta disponibilidad. Si se desea obtener una lista más amplia por favor consulte la dirección: <http://www.linux-ha.org>

## **Monografía del tema**

Esta es una monografía que define lo más importante de este capítulo.

## SISTEMAS DE ALTA DIPONIBILIDAD

<p style="text-align: center;"><b>Alta Disponibilidad (HA)</b></p> <p>Se refiere a poder disponer de recursos en un sistema de cómputo, aun en el caso de que de algún componente del sistema falle.</p>	<p>La alta disponibilidad esta basada en la replicación de elementos, mucho más baratos que in solo elemento tolerante a fallos. Si hablamos de replicar servidores, hablaremos de un cluster de alta disponibilidad.</p> <p>Subsistemas involucrados en el esquema de alta disponibilidad:</p>	<p style="text-align: center;"><b>Disponibilidad Continua</b></p> <p>Esto implica un servicio que se brinda ininterrumpidamente. Representa un estado ideal, y es generalmente utilizado para indicar altos niveles de disponibilidad en los cuales solo se permiten muy pocos minutos de caída del sistema.</p>
<p style="text-align: center;"><b>Tolerancia a Fallos</b></p> <p>Este es un medio para lograr muy altos niveles de disponibilidad. Un sistema tolerante a fallos tiene la habilidad de continuar con la prestación del servicio a pesar de una falla en el hardware o el software, y se caracteriza por la redundancia de hardware, incluyendo CPU, Memoria y subsistemas de I/O.</p>	<ul style="list-style-type: none"> <li>• Sistema de computación</li> <li>• Sistemas de Red</li> <li>• Sistemas de Almacenamiento</li> </ul>	<p style="text-align: center;"><b>Single Point Of Failure (SPOF)</b></p> <p>Punto Único de Fallo es un componente, ya sea de hardware o software, cuyo falló resulta en la pérdida del servicio; tales componentes no están respaldados por componentes redundantes.</p>
<p style="text-align: center;"><b>Proyectos y Productos Relacionados de HA</b></p> <p>Existen algunos productos y proyectos; tanto de hardware y software, destinados exclusivamente a conseguir la Alta disponibilidad en los sistemas.</p>	<p style="text-align: center;"><b>Alta Disponibilidad Con Linux</b></p> <p>Actualmente Linux es conocido, como un sistema operativo estable y robusto. Linux debe ofrecer una serie de facilidades, para que los sistemas de alta disponibilidad puedan integrarse correctamente tal y como se hace en otros entornos. Distribuciones como VA, SuSE, RedHat o Conectiva y Debian están desarrollando soluciones de HA basadas en Linux.</p>	





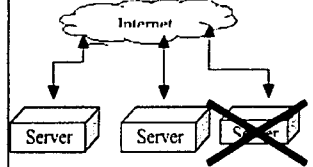
# SISTEMAS DE ALTA DIPONIBILIDAD



**HighAvailability**

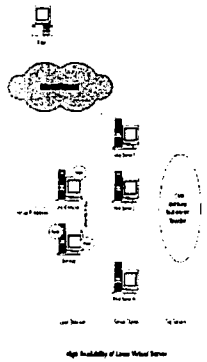
Alta Disponibilidad (HA)

Tolerancia a Fallos



**Microsoft**

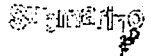
Proyectos y Productos Relacionados de HA



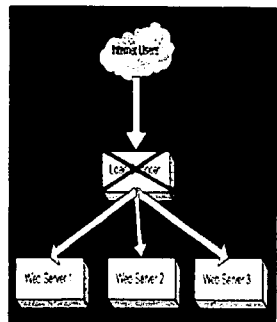
High Availability of Linux Web Server



Alta Disponibilidad con Linux



Disponibilidad Continua



Single Point Of Failure (SPOF)

TESIS CON FALLA DE ORIGEN

## Alta disponibilidad con Linux

En la siguiente sección presentaré unas razones por las que considero que Linux es un sistema operativo lo suficientemente confiable como para establecer sobre él algún sistema de alta disponibilidad.

### Sistemas de archivos

Como ya expliqué en las secciones pasadas de este capítulo, los nuevos sistemas de archivos implementan lo que se conoce como registro de bitácora (journaling), con lo cual se puede recuperar el sistema de una manera más rápida. Linux (el kernel) implementa varios tipos de sistemas de archivos que permiten de manera más rápida la recuperación del sistema en caso de que hubiera existido algún fallo. Entre los sistemas de archivos que Linux implementa están: ext3 y ReiserFS.

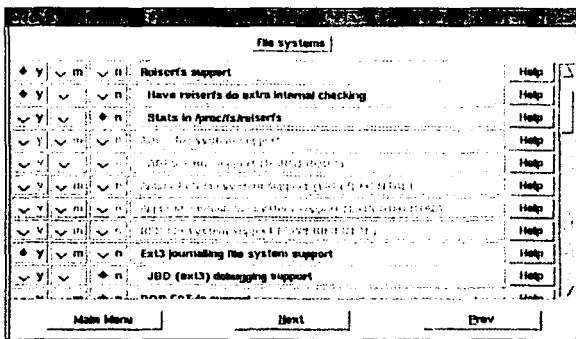


Figura 3.16: Sistemas de archivos de Linux

Otra característica que posee el Linux es que a partir de la versión 2.4.15 se ha incorporado el sistema de archivos en red llamado intermezzo (<http://www.intermezzo.org>), el cual está enfocado a la alta disponibilidad mediante la replicación de servidores. Aun está en una etapa experimental sin embargo con el tiempo se podrá contar con un versión completamente estable, la cual será muy útil para los servicios de alta disponibilidad.

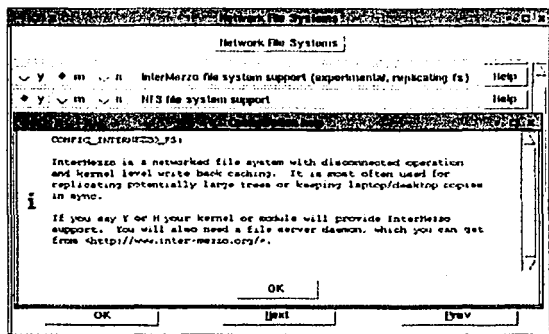


Figura 3.17: Sistemas de archivos Intermezzo

## Sistemas de almacenamiento

Linux posee la característica de poder combinar varias unidades físicas en unidades lógicas, lo que permite obtener configuraciones de RAID y LVM. Los niveles de RAID que Linux soporta son los niveles 0 (datos distribuidos), 1 (espejo de datos), 4 (discos independientes con paridad compartida) y 5 (Discos de datos independientes con bloques de paridad distribuidos). Lo que permite obtener mayores niveles de protección.

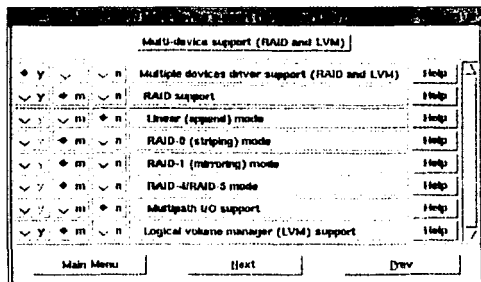


Figura 3.18: Niveles de RAID en Linux

Todas estas características están incluidas en el kernel de Linux, pero combinadas con otros programas de software libre, pueden lograr mayores tiempos de disponibilidad en servicios de cómputo como paginas web, bases de datos y servidores de archivos.

## Proyecto Ultramonkey

<http://www.ultramonkey.org>

Ultramonkey es un proyecto que busca la creación de servicios de altamente disponibles, además proporciona las herramientas necesarias para lograr el balanceo de carga entre los servidores.

Las herramientas proporcionadas por este proyectos están enfocadas en servidores con el sistema operativo Linux, sin embargo debido a que son distribuidas en código fuente pueden ser portadas a otros sistemas operativos tipo Unix.

### **Topologías de sistemas de alta disponibilidad que se pueden utilizar**

Existen varios diseños de sistemas de alta disponibilidad que se pueden implantar con el uso de las herramientas del proyecto Ultramonkey. Básicamente se basan en dos cosas: balanceo de carga y respaldo del sistema que proporciona el servicio mediante un pequeño cluster.

A continuación se muestran las diferentes topologías que pueden ser aplicadas para la creación de sistemas de alta disponibilidad.

#### ***Alta disponibilidad***

Esta topología es la más fácil de implementar, ya que el hardware que se requiere es mínimo. Consisten en tener dos servidores (reales) monitoreándose uno al otro y donde uno de ellos sea el que proporcione el servicio. Cuando uno el servidor maestro falle, en otro tomará su lugar. Esto se realiza utilizando el paquete conocido como Heartbeat.

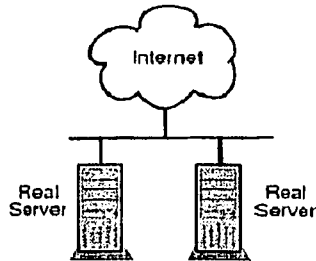


Figura 3.19: Alta disponibilidad

### Balanceo de carga

Este diseño se enfoca en proporcionar servicios de balanceo de carga con un mínimo de requerimientos de hardware.

Para esta topología se necesita una maquina que hará el papel de director y varias maquinas que serán las que se encargarán de atender las peticiones. La maquina Director actúa como si fuera un gateway para la red de servidores reales. La maquina Director se encarga de distribuir las peticiones ya sea de los protocolos TCP/IP o UDP hacia los servidores reales. Este mecanismo de control de conexiones es conocida como *Switcheo de capa 4*

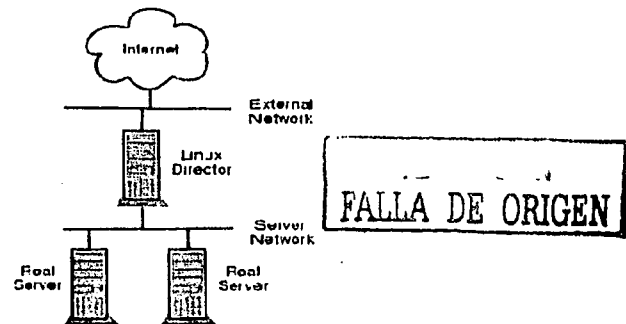


Figura 3.20: Balanceo de Carga

### Alta disponibilidad y Balanceo de carga

Esta topología permite una alta disponibilidad combinado con el servicio de balanceo de carga. Se requiere un mínimo de 4 nodos para la implementación de esta topología.

Esta topología existe un servidor llamado Director que esta activo, mientras que otro permanece como respaldo. El Director se encarga de aceptar el tráfico para una dirección IP. Las conexiones que lleguen son distribuidas a través de él hacia los servidores reales, mediante el uso de LVS. Los dos directores (el activo y el de respaldo) se monitorean mutuamente y cuando el servidor activo falla, el servidor de respaldo asume la dirección IP y se encarga de proporcionar el servicio.

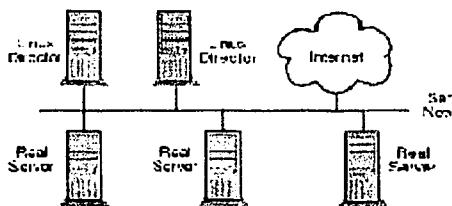


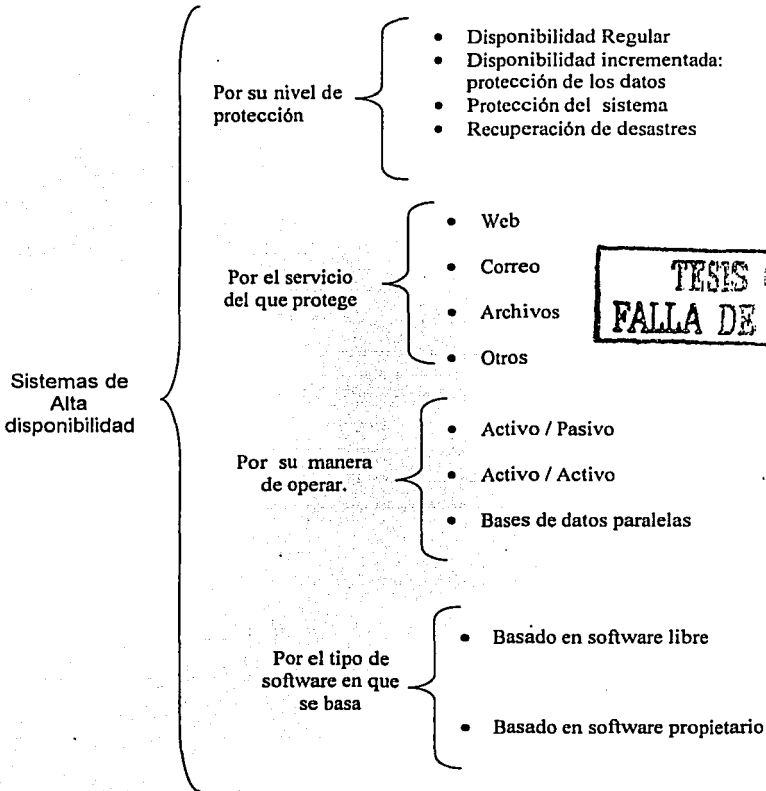
Figura 3.21: Alta disponibilidad

### Cuadro de clasificación del tema

El siguiente es un cuadro donde se puede ver la clasificación de los sistemas de alta disponibilidad de acuerdo a varios criterios.

TESIS CON  
FALLA DE ORIGEN

Clasificación de los sistemas de  
alta disponibilidad



TESIS CON  
FALLA DE ORIGEN

## Capítulo 4

# Marco Metodológico

*"Donde todos piensan igual, nadie piensa mucho"*

**Auguste Rodin**, escultor francés

### Variables

Como ya mencione en el capítulo 1, las variables tanto dependientes como independientes que están involucradas en el estudio de este problema son muy variadas, y algunas producen más de un efecto. Es por eso que decidí hacer una clasificación de las variables de acuerdo a características técnicas del software libre, los grupos son:

- Las relativas al grado de desarrollo técnico del software libre.
- Las ajenas al grado de desarrollo técnico del software libre.

Esta nueva agrupación permite identificar con mayor claridad las causas por las cuales el SL no es utilizado y permite de manera más fácil, al lector de esta tesis, identificar las causas por las que el SL no es utilizado.

De tal manera que todas las variables presentadas en el capítulo 1 pueden ser clasificados de esta manera:



Causas relativas al grado de desarrollo técnico del SL de Causas ajenas al grado de desarrollo técnico del SL

- |  |  |
|--|--|
| <ul style="list-style-type: none"> <li>□ Hardware incompatible. (No existen drivers)</li> <li>□ Porque no esta lo suficientemente desarrollado (software propietario es mejor)</li> <li>□ Aplicaciones difíciles de utilizar. (interfaz de usuario)</li> </ul> | <ul style="list-style-type: none"> <li>□ Por que no existen compañías que brinden soporte para el software libre en caso de que ocurran problemas.</li> <li>□ Personal no capacitado</li> <li>□ Costumbre.</li> <li>□ Políticas Internas.</li> </ul> |
|--|--|

## Variables de control

No se utilizaron variables de control para esta investigación.

## Hipótesis definitiva

Llegado a este punto fue necesario modificar la variable preliminar presentada en el capítulo I, con el fin de adecuarla a las modificaciones realizadas en el punto 4.1 relativo a las variables del problema.

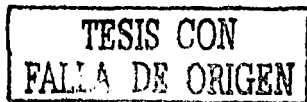
Ahora la **variable independiente** viene a ser el conjunto formado por las causas que no son relativas al grado de desarrollo técnico del SL. Mientras que la **variable dependiente** continua siendo la utilización (o para el caso de esta tesis la **no-utilización**) del software libre en los sistemas de alta disponibilidad.

Por lo que la hipótesis de trabajo definitiva queda como sigue:

***"Las características ajenas al desarrollo técnico del software libre, tales como falta de compañías que respalden los productos de software, falta de conocimiento por parte de los administradores de la existencia del software libre, y en ocasiones la costumbre, y las políticas internas de las compañías, son las que evitan que el software libre sea implementado en los sistemas de alta disponibilidad"***

## Definición del Universo

El universo del problema lo constituyen todas aquellas personas que administran sistemas de alta disponibilidad que fueron implantados con software propietario.



Esto es debido a que intento saber cuales fueron las causas que hicieron que dichos sistemas se implantaran con ese tipo de software en lugar de utilizar el software libre.

El universo resulta ser muy heterogéneo ya que existe gran diversidad en cuanto a tipos de sistemas que se tratan, la experiencia de quien administra el sistema, y los requerimientos de disponibilidad.

Como este trabajo no va enfocado a hacer realizar pruebas plenas, simplemente se darán la opinión de personas que señalen que la hipótesis es válida.

## Muestra

Para este trabajo maneje lo que se conoce como una muestra no probabilística de juicio o intencionada, ya que sólo emitiré una opinión sobre la base de los conocimientos adquiridos en este trabajo y ayudado por un cuestionario que fue aplicado, a personas que se encargan de la administración de sistemas de alta disponibilidad.

## Definición del método de la investigación

Para esta investigación utilice el método de la **encuesta**. Mediante un cuestionario me presente ante las personas que componen la muestra y pedí sus opiniones referente a la implementación del software libre en los sistemas de alta disponibilidad.

## Costos de la investigación

Costos de la investigación			
Sueldos			\$12,000
Renta de oficina			\$10,000
Mobiliario y equipo			
	Computadoras	\$11,000	
	Impresora	\$1,500	
	Escritorio	\$2,000	
	Archivero	\$700	\$15,200
Papelería			\$6,000
Gastos Diversos			
	Teléfono	\$2,000	
	Transportación	\$3,000	
	Libros	\$4,000	
	Internet	\$300	\$9,300
<b>TOTAL</b>			<b>\$52,500</b>

## Colaboradores y apoyos

Para la realización de esta parte de la investigación, conté con el apoyo de varios compañeros de la carrera que me ayudaron a contactar a administradores de sistemas de alta disponibilidad. Por lo que aprovecho este momento para reconocer su colaboración.

- Cruz Rodríguez José Luis
- Revimar Gallardo Javier
- Vicores Hernández Edgar Ricardo

## Cuestionario preliminar

1. El sistema que usted administra debe de estar disponible durante:

- Sólo uno o dos días a la semana   
  Toda la semana   
  Siempre (24x7)
- Otro:

**Razón de la pregunta:** Conocer el nivel de disponibilidad de los sistemas que los encuestados administran.

**Respuesta esperada:** Cualquiera que el usuario conteste.

2. ¿Cuáles son las principales fallas que presentan los sistemas que usted administra?

- Fallo de hardware    Fallo de software    Pérdida de energía  
 Otro:    Errores humanos    Red

**Razón de la pregunta:** Conocer las principales fallas que se presentan en los sistemas informáticos.

**Respuesta esperada:** "Fallo de hardware".

3. Cuando ocurre una falla de hardware en el equipo, ¿el servicio deja de ser proporcionado?

- Sí    NO

**Razón de la pregunta:** Saber si el sistema de que administra el encuestado, está protegido contra fallas de hardware

**Respuesta esperada:** NO

4. Cuándo ocurre una falla de hardware ¿cuánto tiempo (en promedio) tarda en que el servicio este funcionando nuevamente?

- Menos de 1 minuto    Menos de 1 hora    Menos de 1 día  
 Más de un día    Otro:

**Razón de la pregunta:** Conocer cuánto es el tiempo promedio que se invierte en recuperar el servicio cuando se presenta esta clase de fallas.

**Respuesta esperada:** Menos de 1 minuto

5. De la siguiente lista de fallas por favor enumere cuales son las que se presentan con mayor frecuencia, asignándole un 1 a la falla que se

presenta con mayor frecuencia, 2 a la inmediata y así sucesivamente.

- |                                   |  |                                  |
|-----------------------------------|--|----------------------------------|
| <input type="checkbox"/> Hardware | <input type="checkbox"/> Sistema Operativo | <input type="checkbox"/> Energía |
| <input type="checkbox"/> Software | <input type="checkbox"/> Aplicaciones      | <input type="checkbox"/> Otro:   |
| <input type="checkbox"/> Humanos  | <input type="checkbox"/> Red               | <input type="checkbox"/> Otro:   |

**Razón de la pregunta:** Conocer el orden de importancia en que se presentan las fallas.

**Respuesta esperada:**

- |              |                       |             |
|--------------|-----------------------|-------------|
| (1) Hardware | (2) Sistema Operativo | (7) Energía |
| (5) Software | (3) Aplicaciones      | ( ) Otro:   |
| (4) Humanos  | (5) Red               | ( ) Otro:   |

6. Cuando ocurre una falla en el sistema operativo del equipo, ¿Cuánto tiempo tarda en que el servicio este funcionando nuevamente?

- |  |   |   |
|--|---|---|
| <input type="checkbox"/> Menos de 1 minuto | <input type="checkbox"/> Más de 1 minuto -<br>menos de 1 hora | <input type="checkbox"/> Más de 1 hora -<br>menos de un día |
| <input type="checkbox"/> Más de 1 día      | <input type="checkbox"/> Otro:                                |   |

**Razón de la pregunta:** Conocer el tiempo en que tarda en recuperarse el funcionamiento del servicio cuando ocurre una falla de software.

**Respuesta esperada:** Menos de 1 minuto.

7. ¿Usted cree que el o los servicios que brindan alojan los equipos que usted administran necesitan alguna clase de protección contra fallas, dada su importancia?

- Sí       NO

**Razón de la pregunta:** Saber si los equipos que el entrevistado administra requieren de protección adicional.

**Respuesta esperada:** Sí

8. Utiliza alguna herramienta / configuración / programa para la protección del sistema:

SÍ

NO

**Razón de la pregunta:** Saber si el entrevistado protege el sistema de alguna forma.

**Respuesta esperada:** Sí

9. Utiliza herramientas de software libre para brindar mayor disponibilidad de los servicios que usted tiene a su cargo:

SÍ

NO

¿Por qué? :

**Razón de la pregunta:** Saber si el encuestado utiliza software libre para la protección de los servicios; Esta pregunta será clave para la comprobación o negación de la hipótesis.

**Respuesta esperada:** Sí

10. ¿Cree usted que las herramientas desarrolladas bajo alguna licencia libre y que están dedicadas a buscar la protección de los sistemas informáticos, cuentan con las características ideales para ser utilizadas en sistemas en producción?

SÍ     NO

**Razón de la pregunta:** Saber la opinión que tiene el entrevistado acerca de las herramientas de software libre.

**Respuesta esperada:** Sí

**Cuestionario Definitivo**

NOMBRE:

PUESTO:

TELÉFONO DE OFICINA:

CORREO:

1. ¿Cuál es el requerimiento de disponibilidad del servicio más crítico que usted administra? Entiéndase por servicio como la finalidad o utilidad del equipo de cómputo, Ej. un sitio web, el servicio de correo, etc.

- Sólo uno o dos días a la semana  
 Toda la semana  
 Siempre (24x7)  
 Otro:

2. ¿Cuál es la principal falla que presentan los servidores que administra?

- Fallo de hardware  
 Fallo de software  
 Pérdida de energía  
 Otro:  
 Red  
 Errores Humanos

3. De la siguiente lista por favor marque aquellas que serían las causas por las cuales los usuarios no podrían hacer uso del sistema:

- Fallo en la tarjeta de Red  
 Fallo en la fuente de poder  
 Fallo en el disco de sistema operativo  
 Fallo en el disco de datos  
 Fallo en el procesador  
 Realización de respaldos  
 Fallo en tarjetas controladoras  
 Otro:  
 Ninguno

4. Cuando ocurre una falla de hardware ¿cuánto tiempo (en promedio) tarda en que el servicio este funcionando nuevamente?

- Menos de 1 minuto  
 Menos de 1 hora  
 Menos de 1 día  
 Más de 1 día  
 Otro:

5. Cuando ocurre una falla en el sistema operativo de equipo, ¿el servicio deja de ser proporcionado?

 Sí NO

En caso de respuesta afirmativa de la pregunta anterior, contestar la pregunta 6.

6. Cuando ocurre una falla en el sistema operativo del equipo, ¿cuánto tiempo tarda en que el servicio este funcionando nuevamente?

- ( ) Menos de 1 minuto ( ) Más de 1 minuto - ( ) Más de 1 hora -  
Menos de 1 hora Menos de 1 día  
( ) Más de 1 día ( ) Otro:

7. Usted cree que el o los servicios que brinda el equipo que usted administra necesitan alguna clase de protección contra fallas, dada su importancia

- ( ) Sí ( ) NO

8. Utiliza herramientas de software libre para brindar mayor disponibilidad de los servicios que usted tiene a su cargo: Entiéndase por software libre aquel programa de computadora al cual se tiene acceso a su código fuente y se permite su modificación y redistribución.

- ( ) Sí ( ) NO  
¿Por qué? :

En caso de respuesta afirmativa en la pregunta 8, conteste la pregunta 9.

9. ¿Qué herramientas utiliza?

10. ¿Utiliza alguna herramienta que no sea software libre para brindar mayor disponibilidad a los servicios que usted administra?

- ( ) Sí ( ) NO



Desea recibir la información acerca de los resultados que arroje esta investigación: SÍ ( ) NO ( )

Gracias.

## **Realización de la encuesta**

Las personas que contribuyeron a la investigación contestando el cuestionario presentado en la sección anterior, fueron:

- García Olivares Enrique
- López Pineda Daniel Antonio  
Pronósticos Deportivos
- Ochoa Sánchez Aloy  
Cía. de Luz y Fuerza del centro
- Ramírez Álvarez Marco Antonio  
Secretaría de Salud
- Vega Alvarado Armando  
Dirección General de Cómputo Académico - UNAM

## **Captura de la información**

La captura de las respuestas de los cuestionarios esta organizada en forma de matriz, lo cual permite más fácilmente analizar las respuestas que cada encuestado proporciona. Las columnas de esta matriz están organizadas como sigue: 1) La primera contiene la pregunta completa realizada, 2) contiene las opciones que esa pregunta contenía, esa columna se encuentra dividida en renglones dependiendo del número de opciones de la pregunta, 3 - 7) Son las columnas reservadas para registrar las respuestas de los entrevistados proporcionaron.

Pregunta	Opciones	Armando Vega Alvarado	Ramírez Álvarez Marco Antonio	Enrique García Olivares	López Pineda Daniel Antonio	Ochoa Sánchez Aloy
1. ¿Cuál es el requerimiento de disponibilidad del servicio más crítico que usted administra?	Sólo uno o dos días a la semana					
	Toda la semana			*	*	
	Siempre (24x7)	*	*			*
	Otro					
2. ¿Cuál es el principal tipo de falla que presentan los servidores que administra?	Fallo de hardware					
	Fallo de software					
	Pérdida de energía		*			
	Red	*		*		*
	Errores humanos				*	
	Otro					

TESIS CON  
FALLA DE ORIGEN

Pregunta	Opciones	Armando Vega Alvarado	Ramírez Álvarez Marco Antonio	Enrique García Olivares	López Pineda Daniel Antonio	Ochoa Sánchez Aloy
3. De la siguiente lista por favor marque aquellas que serían las causas por las cuales los usuarios no podrían hacer uso del sistema:	Fallo de tarjeta de red	*			*	
	Fallo de disco de datos				*	
	Fallo de tarjetas controladoras				*	
	Fallo de fuente de poder				*	
	Fallo en el procesador				*	
	Fallo en el disco del sistema operativo				*	
	Realización de respaldos y mantenimiento				*	
	Ninguno		*	*		*

TESIS CON  
FALLA DE ORIGEN

Pregunta	Opciones	Armando Vega Alvarado	Ramírez Álvarez Marco Antonio	Enrique García Olivares	López Pineda Daniel Antonio	Ochoa Sánchez Aloy
4. Cuando ocurre una falla de hardware ¿cuánto tiempo (en promedio) tarda en que el servicio este funcionando nuevamente?	Menos de 1 minuto			*		*
	Menos de una hora	*			*	
	Menos de un día					
	Mas de 1 día		*			
	Otro					
5. Cuando ocurre una falla en del sistema operativo del equipo, ¿el servicio deja de ser proporcionado?.	SI		*			
	NO	*		*	*	*
En caso de respuesta afirmativa de la pregunta anterior, contestar la pregunta 6						

TESIS CON  
FALLA DE ORIGEN

Pregunta	Opciones	Armando Vega Alvarado	Ramírez Álvarez Marco Antonio	Enrique García Olivares	López Pineda Daniel Antonio	Ochoa Sánchez Aloy
6. Cuando ocurre una falla en el sistema operativo del equipo, ¿Cuánto tiempo tarda en que el servicio este funcionando nuevamente?	Menos de un minuto					
	Más de 1 minuto - menos de 1 hora					
	Más de 1 hora - Menos de un día					
	Más de 1 día		*			
	Otro					
7. Usted cree que el ó los servicios que brindan los equipos que usted administra necesitan alguna clase de protección contra fallas, dada su importancia.	SI	*	*	*	*	*
	NO					

TESIS CON  
FALLA DE ORIGEN

Pregunta		Opciones	Armando Vega Alvarado	Ramírez Álvarez Marco Antonio	Enrique García Olivares	López Pineda Daniel Antonio	Ochoa Sánchez Aloy
8. Utiliza herramientas de software libre para brindar mayor disponibilidad de los servicios que usted tiene a su cargo:		SI					
		NO	*	*	*	*	
¿ Por que?	Causas relativas a las características técnicas del software libre	Porque no está lo suficientemente desarrollado					
		Porque es muy difícil de manejar					
	Causas externas a las características técnicas del software libre	Por que no existen compañías que brinden soporte para el software libre en caso de que ocurran problemas			*	*	
		Por que no lo conozco	*				
		Por políticas de la organización					
		Costumbre		*			*

TESIS CON  
FALLA DE ORIGEN

Pregunta	Opciones	Armando Vega Alvarado	Ramírez Álvarez Marco Antonio	Enrique García Olivares	López Pineda Daniel Antonio	Ochoa Sánchez Aloy
9. ¿Qué herramientas utiliza?						
10. ¿utiliza alguna herramienta que no sea software libre para brindar mayor disponibilidad a los servicios que usted administra?	SI	*	*	*	*	*
	NO					

TESIS CON  
FALLA DE ORIGEN

## Análisis de los resultados

1. ¿Cuál es el requerimiento de disponibilidad del servicio más crítico que usted administra?

Pues los sistemas que estas personas administran tienen dos requerimientos de disponibilidad distintos, lo que nos comprueba que un sistema de alta disponibilidad no necesariamente tiene que estar disponible las 24 horas al día de los 365 días del año, sino que debe de estar disponible en el tiempo que los usuarios requieran hacer uso de él.

2. ¿Cuál es el principal tipo de falla que presentan los servidores que administra?

Ya que estos sistemas cuentan con redundancia en su hardware, las fallas de esta clase de componentes se ven disminuidas, siendo las más comunes las de red, pérdida de energía y errores humanos.

3. De la siguiente lista por favor marque aquellas que serían las causas por las cuales los usuarios no podrían hacer uso del sistema:

Esta pregunta 3 menciona las causas que los sistemas comunes (sin ninguna clase de protección) presentan con mayor frecuencia, esto nos dice que las fallas de 3 de ellos mencionaron que ninguna de esas fallas evitaría que los usuarios utilizaran el sistema.

4. Cuando ocurre una falla de hardware ¿Cuánto tiempo (en promedio) tarda en que el servicio este funcionando nuevamente?

Vemos que el tiempo de indisponibilidad de los servicios se ve considerablemente reducida en la mayoría de los casos, donde la respuesta fue "menos de 1 minuto", no hay una pérdida completa del servicio sino simplemente una degradación del rendimiento. En el caso donde la respuesta fue "más de un día", es debido a que era necesario llamar al proveedor del equipo para que enviara a su personal de soporte técnico.

5. Cuando ocurre una falla en del sistema operativo del equipo, ¿El servicio deja de ser proporcionado?

Pues aquí casi fue unánime donde se respondió que las fallas en el sistema operativo no afectan el servicio, debido a que otro equipo toma el lugar del que fallo.

Cuando ocurre una falla en el sistema operativo del equipo, ¿Cuánto



tiempo tarda en que el servicio este funcionando nuevamente?

El hecho de que el servicio estuviera fuera por más de un día para los casos en los que hubiera una falla en el sistema operativo, es debido a que es necesario llamar al equipo de soporte técnico del proveedor del que se adquirió el equipo.

6. ¿Usted cree que el o los servicios que brindan los equipos que usted administra necesitan alguna clase de protección contra fallas, dada su importancia?

Las respuestas fueron unánimes todos los administradores del sistema están de acuerdo que sus servicios requieren protección contra fallas, lo que nos indica que existe una preocupación por buscar e implementar tecnologías que ayuden a proteger al sistema.

7. ¿Utiliza herramientas de software libre para brindar mayor disponibilidad de los servicios que usted tiene a su cargo. ? ¿Por qué?

Esta fue la pregunta clave de la investigación ya que aquí los encuestados dieron sus razones por las que no utilizan el software libre para los sistemas de alta disponibilidad. Cada cual dio sus razones las cuales fueron agrupadas de acuerdo a lo establecido en la sección 4.1 del trabajo. Ninguno de los entrevistados señaló como razón para no utilizar el software libre alguna de las encontradas en el grupo de las relativas a las características técnicas del software libre, por el contrario la mayoría de las respuestas se enfocan a que no existen compañías que den soporte para el software libre en caso de que ocurran problemas.

8. ¿Utiliza alguna herramienta que no sea software libre para brindar mayor disponibilidad a los servicios que usted administra?

Es confirmación, esta pregunta nos dice que los sistemas de estas personas si están implementados con alguna clase de protección basada en software no libre.

## **Conclusiones de la encuesta**

1. Sobre la base de los resultados presentados en la sección anterior se puede ver que el uso de sistemas de alta disponibilidad reduce en gran medida los tiempos de fuera de servicio que se experimentan cuando ocurren fallas ya sea de hardware o de software.
2. Es muy notorio que un gran número de soluciones para los sistemas de alta disponibilidad están basados en software no libre lo que en ocasiones lleva a que se este atado a los programas de soporte técnico que el

proveedor de la solución tenga contemplado.

3. Que las razones por las que no se utiliza el software libre son por lo general razones ajenas a los aspectos técnicos (funcionalidad, seguridad, robustez), y entre la principal causa es que no existen compañías que respalden la tecnología, lo cual provoca desconfianza en los administradores para el uso del software libre en los sistemas de alta disponibilidad.
4. Dado que este trabajo de tesis no está enfocado a realizar pruebas plenas; y que la elección de una muestra no probabilística de juicio o intencionada, fue con el objetivo de obtener las opiniones de expertos que validaran la hipótesis y con base a las respuestas de los cuestionarios aplicados puedo concluir que bajo la relación hipotética planteada en la sección 4.3 es válida.

## Capítulo 5

# Marco Instrumental

*"Siempre que enseñes, enseña a la vez a dudar de lo que enseñas"* José Ortega y Gasset

### Propuestas de acción

Estas son las propuestas de acción que emprenderé:

1. Crear una página en Internet donde se cuente con la información relativa a los temas de "alta disponibilidad" y "software libre".
2. Participar con la exposición de este trabajo en Segundo Congreso Nacional de Software Libre (CONSOL), a llevarse a cabo a principios del año 2003
3. Escribir un artículo acerca del software libre y sugerir su inclusión en el suplemento llamado enter@te, que se publica cada mes con la gaceta UNAM
4. Sugerir a la directora de la carrera M. en I. Graciela Bribiesca Correa, la realización de una conferencia donde se trate el tema de software libre.
5. Sugerir la inclusión en la materia de sistemas operativos multiusuarios la inclusión de varios temas referentes al software libre.

## Plan de acción para la realización de la página en Internet

He creado un sitio en la dirección <http://www.opensource.org.mx/~victor/> donde se puede encontrar información relacionada con los sistemas de alta disponibilidad y el software libre.

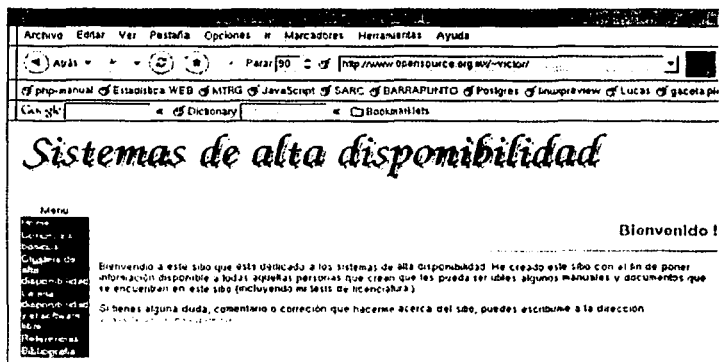


Figura 5.1: Pantalla de bienvenida

TESIS CON  
FALLA DE ORIGEN

Archivo Editar Ver Favoritos Opciones de Menú Herramientas Ayuda

Inicio

# Sistemas de alta disponibilidad

## Conceptos Básicos

Existen sistemas que por su importancia requieren del uso de redundancia y software especializado y en la medida de la posibilidad y el presupuesto más sistemas a través de la protección de un tercer servidor ubicado en las localidades distantes.

El objetivo de un sistema de alta disponibilidad es garantizar la continuidad de los servicios de los usuarios. El primer requisito es la alta disponibilidad de los servidores que se debe garantizar a través de un tercer servidor ubicado en una localidad distante al tercero cuando la carga de trabajo es excesiva y el tercero se encuentra disponible.

Existe un tercer servidor de respaldo en el momento de emergencia que se debe garantizar la máxima disponibilidad. Cada uno de estos servidores puede ser operado por un administrador de los sistemas. El primer requisito es la alta disponibilidad de los servidores que se debe garantizar a través de un tercer servidor ubicado en una localidad distante al tercero cuando la carga de trabajo es excesiva y el tercero se encuentra disponible.

Los SPC's se pueden instalar en:

• Servidores de Alta Disponibilidad. Estos son los servidores de respaldo que se encuentran en las localidades distantes.

Figura 5.2: Conceptos básicos

Archivo Editar Ver Favoritos Opciones de Menú Herramientas Ayuda

Inicio

# Sistemas de alta disponibilidad

## Bibliografía

Aplicaciones críticas y sistemas de alta disponibilidad. Los sistemas de alta disponibilidad.

John Deegan y John Deegan.

High Availability Systems.

Este libro presenta un enfoque práctico de alta disponibilidad. A través de los ejemplos de los sistemas de alta disponibilidad se puede crear un sistema de alta disponibilidad que sea más seguro que se prepare para poder operar en caso de un fallo de los servidores.

High Availability Systems.

Este libro muestra un enfoque práctico de alta disponibilidad. A través de los ejemplos de los sistemas de alta disponibilidad se puede crear un sistema de alta disponibilidad que sea más seguro que se prepare para poder operar en caso de un fallo de los servidores.

High Availability Systems.

Este libro muestra un enfoque práctico de alta disponibilidad. A través de los ejemplos de los sistemas de alta disponibilidad se puede crear un sistema de alta disponibilidad que sea más seguro que se prepare para poder operar en caso de un fallo de los servidores.

High Availability Systems.

Este libro muestra un enfoque práctico de alta disponibilidad. A través de los ejemplos de los sistemas de alta disponibilidad se puede crear un sistema de alta disponibilidad que sea más seguro que se prepare para poder operar en caso de un fallo de los servidores.

CELESTERS

Figura 5.3: Referencias

Este sitio puede ser accesado desde el 2 de septiembre de 2002 y periódicamente se estará actualizando con información nueva.

TESIS CON  
FALLA DE ORIGEN

## **Plan de acción para la participación en el CONSOL 2003**

El día 5 de febrero de 2003 a las 19:00 hice la presentación de este trabajo en el segundo congreso nacional de software libre que se celebró en el auditorio B de la Universidad Pedagógica Nacional. Fue una sesión técnica que duró aproximadamente 1 hora en donde hablé acerca de temas básicos de alta disponibilidad y presenté algunas herramientas de software libre.

A esta plática asistieron alrededor de 20 personas que mostraron mucho interés en las herramientas que pueden ser utilizadas en el sistema operativo Linux.

## **Plan de acción para artículos en el suplemento enter@te**

ENTER@TE es un suplemento mensual que se publica en el último jueves de cada mes en la gaceta UNAM. Está dirigido a personas que tienen interés en temas informáticos pero que no poseen conocimientos muy profundos en el área.

He enviado un artículo a los encargados del suplemento para proponerles que lo publiquen; por lo que si lo consideran de interés lo publicarán en septiembre o en octubre del 2002.

## **Plan de acción para proponer a la Jefa de la Carrera de Informática la celebración de una conferencia**

El día 26 de enero de 2003 hice llegar a la jefa de la División de Informática la Ing. Graciela Bribiesca una propuesta, para que en el momento que ella considerara adecuado se me permitiera dar una charla (ya sea dentro de algún foro especial o evento relacionado con la carrera de Informática) acerca del software libre.

La respuesta por parte de la división de Informática fue favorable ya que la propuesta ha sido tomada en cuenta para una posible presentación dentro del marco de la "semana de la informática" a llevarse a cabo del 24 al 26 de marzo de 2003 en las instalaciones de la Facultad.

**Plan de acción para proponer a la Jefa de la carrera de Informática, la inclusión en materias de la carrera, temas relacionados con el software libre.**

Hice la propuesta de que dentro del temario de la materia de sistemas operativos, se incluyeran los temas que se encuentran en el anexo C. La respuesta fue favorable ya que se mostró gran interés en dichos temas, por lo cual la propuesta será puesta a consideración por el comité encargado de hacer la revisión de los planes de estudio de la carrera de informática.

## Capítulo 6

# Conclusiones

*"Una conclusión tiene el poder de terminar*

*Una investigación así como de abrir un centenar de ellas"*

**VASR**

Uno de los principales objetivos de este trabajo es el de proponer al software libre como una opción para la implementación de los sistemas de alta disponibilidad a pequeño y mediano nivel. A través de los distintos capítulos se ha presentado la manera como puede ser implementado, las ventajas y desventajas de su uso.

De todo el estudio presentado en esta tesis se pueden obtener las siguientes conclusiones:

### Conclusiones

- Un sistema de alta disponibilidad es aquel que mantiene un nivel de servicio óptimo aun cuando ha ocurrido una falla en alguno de sus componentes.
- Que para alcanzar un nivel óptimo de servicio es necesario eliminar los puntos únicos de fallo (SPOF por sus siglas en ingles), mediante la redundancia de los elementos del sistema cuya falla puede hacer que el sistema completo esté inaccesible.
- Para la eliminación de los puntos únicos de fallo se pueden utilizar varios métodos dependiendo del elemento que se desee proteger, siendo los más comunes la creación de lo que se conoce como un cluster de alta disponibilidad, que es un grupo de computadoras que trabajan en forma coordinada y dan la impresión (al usuario final) de que se trata de una



sola computadora de manera que cuando una de ellas falla el usuario no lo nota; también entre las técnicas más utilizadas está la creación de arreglos de discos que permiten proteger los datos en caso de que ocurra una falla en alguno de ellos.

- Existen varias herramientas que son distribuidas bajo alguna licencia libre (Ver anexo B), que permiten elevar el nivel de protección de los datos. Ejemplos de estas herramientas son: Heartbeat, LVS, sistemas de archivos con registro de transacciones y módulos del kernel del sistema operativo Linux, entre otras.
- El tema principal que ocupó este trabajo fue el determinar las causas por las que estas herramientas no son muy utilizadas. En la hipótesis de trabajo se planteó que el hecho de que no fueran muy utilizadas se debía a causas ajenas a las características técnicas del SL, es decir, que su estabilidad, funcionalidad o diseño interno. Para esto solicité su opinión a administradores de sistemas, dicha opinión en ocasiones difería de otros encuestados. El conjunto de las opiniones señalaron que se debía a que no existían compañías que respaldaran los productos en caso de que surgieran problemas, o a la falta de conocimiento en las herramientas y a la costumbre.
- El hecho de que las causas por las cuales no se utiliza el SL fueran de esta clase da pie a pensar que puede ser utilizado siempre y cuando se tengan los conocimientos adecuados en el manejo de las herramientas, a tal grado de que se pueda sentir una seguridad equiparable a la que una compañía puede ofrecer.
- Es importante señalar que el SL (y en general cualquier software) no debe ser utilizado sin antes haber hecho un estudio de viabilidad donde se demuestre que cumple con los requerimientos buscados de seguridad, funcionalidad y robustez. Ya que como mencioné en el capítulo I, para cuestiones realmente importantes (en donde estén en riesgo vidas humanas y no sólo pérdidas económicas) se puede optar por otras opciones más probadas.

## Consideraciones finales

El presente trabajo sólo cubrió una porción de un campo de estudio que creo ofrece muchas áreas de investigación; algunos ejemplos de temas relacionados con esta área que pueden ser investigados más a detalle son: replicación de datos a nivel global, hardware especializado para sistemas de alta disponibilidad.

También es un buen momento para mencionar que existen varias tecnologías que aun se encuentran en una fase de desarrollo tan temprana que no es posible en este momento utilizarla para sistemas en producción. Tal es el caso del proyecto Supersparrow (<http://www.supersparrow.org/>), el cual permite hacer balanceo de carga entre puntos geográficamente muy separados, la versión que en este momento esta disponible es la 0.0.0 lo que indica que aún carece de muchas cosas y puede presentar muchos problemas.

Otra tecnología que me parece es interesante es el sistema de archivos intermezzo el cual es la implementación libre del sistema de

archivos conocido como CODA. Intermezzo permite sincronizar todo un sistema de archivos lo que permite que se tenga varias copias de la misma información, y resulta muy útil para entonos de trabajo con recursos compartidos.

Espero que algún día estas tecnologías sean investigadas para trabajos de tesis de alguien más ya que a mi parecer son tecnologías muy interesantes y donde hay mucho información que distinguir.

Otro punto que estuve pensando mientras llevaba acabo este trabajo es de que se podría poner un comparativo entre el software libre y el software propietario. De manera que se muestre con plenitud las ventajas, desventajas, similitudes y diferencias entre estos dos tipos de software así las personas podrán decir con mayor facilidad por alguno de ellos.

# Glosario

## **API**

Application Program Interface. Interfaz de programación de aplicaciones.

## **Cluster**

Se le denomina cluster a un grupo de dos o más computadoras que están interconectadas proporcionando un servicio y funcionan de tal manera que el usuario final no lo nota.

## **Fallover**

Se le denomina al proceso por medio del cual un servicio que se encontraba corriendo en un servidor que ha fallado, es movido a otro servidor para seguir proporcionando el servicio.

## **Downtime**

Es el tiempo en el cual el sistema no esta disponible, ya sea por fallas o por mantenimiento.

## **heartbeats**

Son los mensajes que los servidores intercambian ( con a yuda del software Heartbeat) para verificar que están funcionando.

## **I-nodo o inodo**

El i-nodo o nodo índice es una estructura donde el sistema operativo guarda información acerca del archivo como la dirección de los bloques que lo conforman, sus atributos, etc.

## **Internet**

Sistema de redes de computadoras enlazadas, con alcance mundial y de continuo crecimiento, que facilita servicios de transmisión de datos como el inicio de sesión remota, transferencia de archivos, correo electrónico, World Wide Web y grupos de noticias. Internet, la cual descansa sobre TCP/IP,

asigna a cada computadora conectada una *dirección de Internet*, única conocida también como dirección IP, con el fin de que dos computadoras puedan localizarse entre sí en la red para intercambiar datos. Aunque existen conexiones de Internet prácticamente en todo los países, el idioma predominante en la red es el inglés y la mayoría de los usuarios viven países anglosajones. Dentro de estas naciones como un nuevo medio de comunicación publica a la par con el sistema telefónico o la televisión por su ubicuidad e impacto.

### **Intranet**

Red de computadoras, basadas en Internet (TCP/IP), diseñada para satisfacer las necesidades internas de una sola organización o compañía. Una intranet, que no necesariamente es abierta a Internet y que casi nunca es accesible desde el exterior, permite a las organizaciones poner recursos a la disponibilidad interna, mediante clientes familiares a Internet como navegadores Web, lectores de noticias, y correo electrónico. Tan sólo con la publicación de la información, como manuales de empleado o directorios telefónicos en Web más que en medios impresos, las compañías pueden lograr significativas recuperaciones sobre la inversión (ROI), mediante la creación de intranets.

### **LVM**

Logical Volume Manager. Es un software que permite la creación de un disco a partir de partes de discos físicos, es decir que a pesar de estar compuesto por varias particiones da la impresión de ser un único disco.

### **LVS**

Linux Virtual Server. Servidor Virtual de Linux, es un software que permite la creación de servidores virtuales a partir de servidores reales, con el fin de realizar el balanceo de carga de las peticiones que lleguen a un servidor.

### **Metadatos**

Son los datos que están contenidos en la tabla de i-nodos y que proporcionan información acerca de un archivo; ejemplo de metadatos son: propietario, fecha de actualización, etc.

### **PCMCIA**

Personal Computer Memory Card Interface Adapter (Adaptador de interfaz de tarjeta de memoria para computadoras). Es una asociación comercial internacional que ha desarrollado estándares para dispositivos como módems y unidades externas de discos duros que pueden conectar con facilidad a las computadoras notebook.

## **RAID**

Redundant Array of Independent Disk. Es un conjunto de estándares para la creación de arreglos de discos que brinde mayor protección y mejor desempeño en discos.

## **SAN**

Storage Area Network es un nuevo modelo para la interconexión de dispositivos de almacenamiento (discos o cintas) con sistemas que hagan uso de ellos.

## **SCSI**

Small Computer System Interfase. Es un estándar para la conexión de dispositivos de almacenamiento.

## **SPOF**

Single Point Of Failure (punto único de falla) se refiere a un elemento crítico dentro de un cluster (o sistema único), y que cuya falla puede llevar a la pérdida del servicio. Los sistemas de alta disponibilidad eliminan o reducen estos puntos mediante la redundancia de componentes.

## **Superbloque**

El súper bloque es el bloque uno del dispositivo. Este bloque contiene información sobre el sistema de archivos, tales como su tamaño en bloques, el nombre del sistema de archivos, número de bloques reservados para i-nodos, la lista de i-nodos libres y el comienzo de la cadena de bloques libres. También contiene el nombre del volumen, momento de la última actualización y tiempo del último backup. Siempre reside en un bloque de 512 bytes.

## **Takeover**

Ocurre cuando un nodo nota que otro ha fallado, entonces el primero obliga al segundo a ceder el servicio de manera que el nodo que ha fallado es eliminado del cluster.

## **UPS**

Uninterrupted Power Supply. Fuente de energía interrumpida.

## **Web**

Aunque comúnmente al servicio web se le conoce simplemente como Internet, este no es más que uno más de los servicios disponibles en Internet, y esta basado en el protocolo HTTP.

# ANEXO B

## Licencias Opensource

La siguiente es una lista de las licencias que son consideradas Opensource aprobadas por la OSI. Cada una tiene sus propios terminos de distribución.

- The GNU General Public License (GPL)
- The GNU Library or "Lesser" Public License (LGPL)
- The BSD license
- The MIT license
- The Artistic license
- The Mozilla Public License v. 1.0 (MPL)
- The Qt Public License (QPL)
- The IBM Public License
- The MITRE Collaborative Virtual Workspace License (CVW License)
- The Ricoh Source Code Public License
- The Python license (CNRI Python License)
- The Python Software Foundation License
- The zlib/libpng license
- The Apache Software License
- The Vovida Software License v. 1.0
- The Mozilla Public License 1.1 (MPL 1.1)
- The Jabber Open Source License
- The Nokia Open Source License
- The Sleepycat License
- The Nethack General Public License
- The Common Public License
- The Apple Public Source License
- The X.Net License
- The Sun Public License
- The Eiffel Forum License
- The W3C License
- The Motosoto License
- The Open Group Test Suite License
- The Zope Public License
- The University of Illinois/NCSA Open Source License

Tomado del sitio <http://www.opensource.org>, para mayor información acerca de cada una de estas licencias, puede dirigirse al sitio.



UNIVERSIDAD NACIONAL  
AVENIDA DE  
MEXICO

México D.F, miércoles 29 de enero de 2003

**Asunto:** Propuesta de conferencia  
sobre software libre.

M. en I, Graciela Bribiesca Correa  
Jefa de la división de Informática  
de la Facultad de Contaduría y Administración

Estimada Jefa de la división de Informática:


En los últimos años, el movimiento de software libre ha ganado mayor aceptación y respaldo por parte de grandes empresas. Como alumno recién egresado de la carrera de Informática, he adquirido los conocimientos necesarios para saber que este movimiento puede traer beneficios a la pequeña y mediana empresa, así como al estudiante de la carrera de Informática.

Es por eso, que mediante este conducto me dirijo a usted con el fin de proponerle la realización de una conferencia, en el tiempo y forma que usted considere adecuado, que presente a la comunidad de la Facultad de Contaduría y Administración las ventajas y desventajas del software libre. Para la realización de esta conferencia me propongo como ponente, ya que mi objetivo es, además de dar cumplimiento con un propósito plasmado en mi tesis profesional, dar difusión de este tipo de tecnología entre la comunidad de la Facultad de Contaduría y Administración.

Además, quisiera proponerle la inclusión al temario de la materia de "Sistemas Operativos Multiusuarios", los temas que presento en el anexo a esta carta.

Sin más por el momento me despido, ofreciéndole mi apoyo para la realización de las propuestas presentadas aquí, y quedando al pendiente de su respuesta.

Atentamente

  
Víctor Augusto Sánchez Ramírez  
Ex-alumno de la carrera de Informática  
Tel: 56 22 60 72  
victor@mail.estadistica.unam.mx

JEFATURA  
ENE. 29 2003

169

Propuesta de temario para inclusión en  
la materia de  
Sistemas Operativos Multiusuarios.

	Tiempo Propuesto	
<b>Tema I: Software Libre</b>		
Antecedentes	15 min.	
Conceptos	15 min.	
Licencias libres	30 min.	
Ventajas y desventajas	30 min.	
<b>TOTAL TEMA I</b>		<b>90 min.</b>
<b>Tema II: Linux</b>		
Conceptos básicos.	20 min.	
Diferencia con otros UNIX	30 min.	
Instalación y configuración	60 min.	
Sistema de archivos	30 min.	
Comandos básicos	30 min.	
<b>TOTAL TEMA II</b>		<b>170 min.</b>



# ANEXO D

## Epígrafe

*Epígrafe:* (Del gr. epigraphe, inscripción, título). Cita que a veces se coloca al comienzo de una obra literaria o científica o de alguna de sus partes.

Cada capítulo de esta tesis comienza con una pequeña cita que a mi parecer expresa su objetivo. Estas citas fueron recogidas de diferentes fuentes tanto impresas como electrónicas, a continuación me propongo a explicar cual fue la razón de escoger la cita que da comienzo con a capítulo.

**Capítulo I: Marco Problemático:** *"no sería paradójico afirmar que el hombre que plantea un problema no es enteramente el mismo que lo resuelve"* **Ramón Y Cajal**

Esta frase la encontré en el libro llamado *"La aventura del trabajo intelectual"*, en la sección referente al planteamiento del problema. Creo que esta frase le queda muy bien a este capítulo, ya que es donde se hace el planteamiento del problema. A esta frase la encuentro completamente cierta, ya que por un lado un problema puede ser resuelto por una persona diferente a la que lo plantea, y en otro caso, aunque dicha persona sea la misma, se podría decir que esa persona paso por un proceso de transformación y al final de ese camino conocido como investigación, se encuentra que ahora ya tiene los conocimientos suficientes para dar solución al problema que el mimos planteo; es decir, evoluciono.

**Capítulo II: Marco Teórico:** *"Los libros no dan sabiduría donde antes no la habla, pero donde algo hay, la lectura la incrementa"* **SIR John Harington**

En el Marco Conceptual se mencionan fuentes escritas que existen sobre el tema de investigación. Pues esta frase refleja bien el objetivo del capítulo, que es de la adquisición de conocimiento a través de la lectura no solamente de libros, sino también de artículos de revistas, sitios en Internet y conferencias.

**Capítulo III: Marco Conceptual:** *"Nunca he encontrado una persona tan ignorante que no se pueda aprender nada de ella."* Galileo Galilei

En el capítulo III se hace el desarrollo de los temas que se relacionan con la investigación. Este capítulo puede servir a los lectores para aprender los conceptos teóricos acerca de los sistemas de alta disponibilidad y el software libre, ya que aquí se presenta la información sintetizada que fue obtenida de las distintas fuentes de información consultadas. Es por eso que creo que de este capítulo se puede aprender varias cosas.

**Capítulo IV: Marco Metodológico:** *"Donde todos piensan igual, nadie piensa mucho"* Auguste Rodin, escultor francés

En este capítulo es donde se presentan las opiniones de los expertos que fueron entrevistados para comprobar la validez de la hipótesis, la opinión de cada uno de ellos es diferente de los otros, (incluso también de la mía), sin embargo eso resulta más enriquecedor que si todos hubieran opinado lo mismo.

**Capítulo V: Marco Instrumental:** *"Siempre que enseñes, enseña a la vez a dudar de lo que enseñas"* José Ortega y Gasset

Con esta frase hago la invitación para que lo aquí plasmado no se tome como una verdad absoluta ya que pude haber caído en errores y omisiones propias de un humano.

**Capítulo VI: Conclusión:** *"Una conclusión tiene el poder de cerrar una investigación o abrir un centenar de ellas", V.A.S.R.*

Con esta frase quiero decir que, aun cuando en una conclusión da fin a un trabajo de investigación, puede existir la posibilidad de que otras personas al conocer dicho trabajo (y obviamente dicha conclusión), se dediquen a comprobar dicha conclusión o ¿por qué no? a refutarla, lo que daría inicio a otra investigación y así sucesivamente pueden encadenarse los trabajos de investigación.

## **Bibliografía**

**WEYGANT S., PETER**  
Clusters for high availability  
Prentice Hall - Hewlett Packard  
1999  
ISBN: 0-13-89355-2

**PIEDAD, FLOYD AND HAWKINS, MICHAEL**  
High Availability: desing, techniques and processes  
Prentice Hall  
2001  
ISBN: 0-13-096288-0

**Marcus, Evan AND Stern, Hal**  
Blueprints for High Availability  
WILEY  
2001

**REAL ACADEMIA ESPAÑOLA**  
Diccionario de la lengua española  
REAL ACADEMIA ESPAÑOLA  
1992  
Madrid

**LAROUSSE**  
Gran diccionario usual de la lengua española  
LAROUSSE  
1998

**Tony Bourke**  
Server Load Balancing  
O'reilly  
2001  
ISBN: 0-596-00050-2

Ulloa Cuevas, Elizabeth  
CLUSTERING: Solución para alta disponibilidad con tecnología intel  
Facultad de ingeniería, UNAM  
Septiembre,2000

Steve Blackmon AND John Nguyen  
High-Availability File Server with heartbeat  
Sys Admin Magazine  
2001

Rafeeq Ur Rehman  
Configuring a High Availability Server with Red Hat  
Sys Admin Magazine  
2001, Vol. 9

## REFERENCIAS ELECTRONICAS

Proyecto GNU  
<http://www.gnu.org>

Sitio de Opensource  
<http://www.opensource.org>

HispaCluster  
<http://www.hispacluster.org>

Proyecto: Linux-ha  
<http://www.linux-ha.org>

Proyecto Ultramonkey  
<http://www.ultramonkey.org/>