

# Anexos

<b>ANEXOS .....</b>	<b>49</b>
<b>A. Web x.0: Estudio de oportunidad .....</b>	<b>49</b>
A.1.Historia .....	49
A.2.Contexto y ámbito del proyecto.....	50
A.3.Misión .....	52
A.4.Alcance .....	52
A.5.Visión.....	53
A.5.1. Atributos .....	59
A.5.2. Herramientas y fundamentos .....	60
A.6.Diagrama de oportunidad .....	94
<b>B. Aplicación de Cloud Computing a CyC.....</b>	<b>96</b>
B.1. Precedentes .....	96
B.2. Definición .....	97
B.3. Tipos de nube según el ámbito .....	98
B.4. Cloud pública vs. Cloud privada .....	99
B.5. Estructura bajo el Cloud .....	100
B.6. Cloud y SOA.....	101
<b>C. Virtualización e hipervisor.....</b>	<b>102</b>
C.1. Introducción .....	102
C.2. Fundamentos de la virtualización .....	102
C.2.1. Estados de máquina.....	102
C.2.2. Requerimientos para la virtualización.....	103
C.2.3. Definiciones .....	104
C.3. Historia de la virtualización.....	104
C.3.1. Virtualización de mainframes .....	105
C.3.2. La necesidad de la virtualización x86 .....	105
C.4. Arquitectura x86 .....	105
C.4.1. IA-32 (x86) .....	107
C.4.2. Niveles de privilegio .....	107
C.4.3. Virtualización de servidor .....	109
C.5. Evolución de la virtualización .....	110
C.5.1. Virtualización total.....	111
C.5.2. Paravirtualización.....	113

C.5.3. Virtualización asistida por hardware.....	114
C.6. Conclusión .....	115
<b>D. Comparativa de hipervisores.....</b>	<b>117</b>
D.1. Análisis inicial .....	117
D.2. Microsoft vs. VMware.....	118
D.2.1. Informe Técnico.....	119
D.2.2. Informe Económico .....	121
D.2.3. Informe Estratégico.....	122
D.3. Conclusiones.....	123
<b>E. Tipos de almacenamiento .....</b>	<b>124</b>
E.1. Introducción .....	124
E.2. Almacenamiento de conexión directa (DAS) .....	125
E.3. Almacenamiento conectado a la red (NAS) .....	126
E.4. Red de área de almacenamiento (SAN).....	127
E.5. Puerta de acceso NAS con una SAN .....	128



## A. Web x.0: Estudio de oportunidad

### A.1. Historia

Hablar de la historia de la WWW supone remontarse al nacimiento del concepto de *hipertexto*. En 1945, el Dr. Vannevar Bush propone MEMEX (*Figura A.1*), una hipotética interfaz capaz de almacenar texto e imágenes y de establecer vínculos a la información contenida en otras máquinas de similares características. Veinte años después, en 1965, el término *hipertexto* quedará acuñado por Ted Nelson en un artículo: “*Texto no secuencial que salta, y que permite al lector realizar elecciones a través de una pantalla interactiva*”. La primera implementación de algo que responde a esas características se lleva a cabo en 1967 en IBM.



*Figura A.1. Bush y MEMEX*

Establecido el motor que impulsará la futura red, habrá que esperar hasta 1980 para que Tim Berners-Lee (*Figura A.2*) se atreva a proponer un sistema que permita vínculos entre nodos arbitrarios. Lo hace ante el CERN, y bautiza a su idea “*Enquire-Within-Upon-Everything*” (Preguntando todo sobre todo), un entorno multiusuario que posibilita a varias personas acceder a los mismos datos. En 1989 redacta una propuesta en firme con una descripción completa, y un año después, a través de la máquina *NeXTcube*, Lee, junto a Robert Caillau, implementa el primer navegador, que será conocido con el mismo nombre que el sistema cuyo nacimiento supone: *World Wide*

*Web.* El hermanamiento con otra tecnología emergente, ARPANET, en 1991 hace posible que el servicio esté disponible públicamente.



*Figura A.2. Sir Tim Berners-Lee*

Surge en este punto la necesidad de un lenguaje sencillo, y sobre todo versátil, que permita exportar la idea a cualquier máquina independientemente de su tecnología o SO. El mismo Lee publica las directrices del HTML en 1991, y previendo el alcance y repercusión del sistema que ha creado, funda el W3C en 1994, organismo que velará por el correcto avance de la WWW, y entre cuyos primeros hitos figura el desarrollo del protocolo HTTP de transferencia de *hipertexto*.

## **A.2. Contexto y ámbito del proyecto**

Durante aproximadamente 2 décadas, la WWW ha ido creciendo y evolucionando, acercándose al concepto que su creador, Tim Berners-Lee, había imaginado: una red de interconexión entre máquinas inteligentes capaces de comunicarse y extraer de manera automática el contenido que un usuario solicite desde un punto cualquiera de la misma.

En el viaje hasta este ideal que todavía continúa, somos capaces de distinguir varias generaciones, cuyas fronteras exactas no están definidas pero donde sí pueden acotarse ciertas tendencias. Nova Spivack, experto de la web, elaboró una gráfica (*Figura A.3*) que pretende resumir el estado del arte.

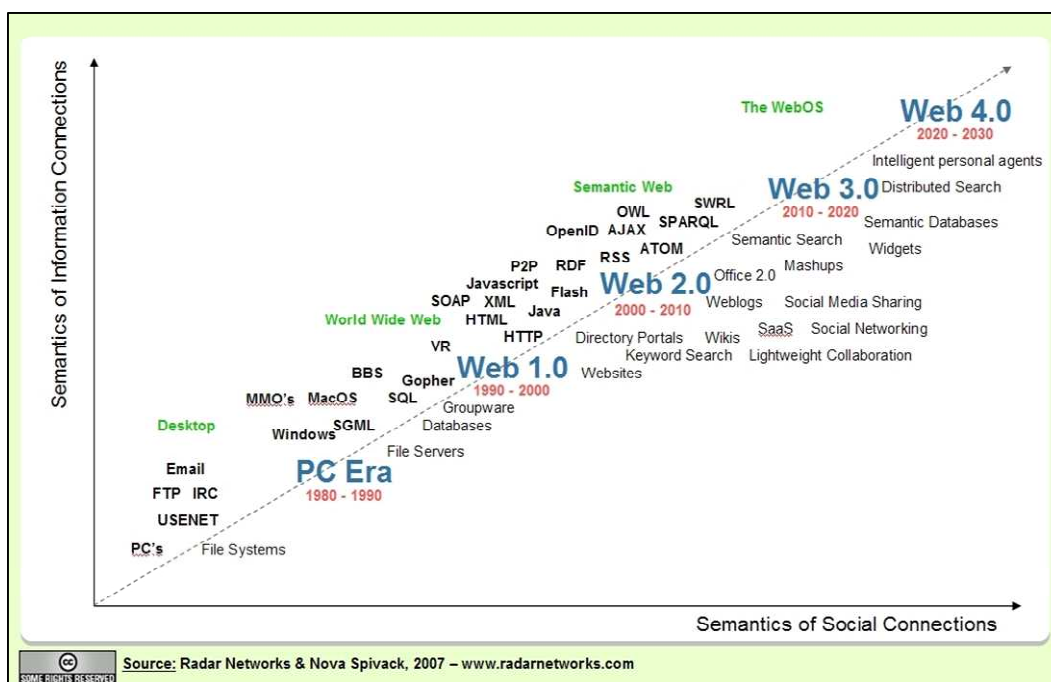


Figura A.3. Evolución de la Web

De acuerdo con esto, y con la realidad en la que nos desenvolvemos, el momento actual se sitúa en la transición entre la *web 2.0* y la *3.0*. Cuánto tardará esta última en ser una realidad es algo que depende de los esfuerzos de todos aquellos que trabajamos desarrollándola.

Sintetizando los conceptos, podríamos resumir las sucesivas etapas de la siguiente manera:

- **Web 1.0:** Web presencial. Los usuarios son meros observadores.
- **Web 2.0:** Web social. Los usuarios interactúan con la red. Observan y aportan.
- **Web 3.0:** Web semántica. La red *entiende* el contenido. Existe comunicación usuario-usuario, usuario-máquina y máquina-máquina.
- **Web 4.0:** Web inteligente. La red es capaz de establecer relaciones de manera automática y explorar en su contenido para ofrecer la respuesta que más se acerque a los deseos del usuario.

De manera gráfica, la evolución de las corrientes de la información queda reflejada en la figura A.4:



*Figura A.4. Las edades de la Web*

### A.3. Misión

La meta es obtener una visión lo más amplia posible de todos los recursos que la evolución de la WWW pone a nuestro alcance. Para llegar a ella, se perseguirán diversos objetivos:

- RECABAR información sobre las herramientas existentes, tanto aquellas derivadas del uso social (2.0) como las caracterizadas por su orientación a los contenidos semánticos (3.0).
- ANALIZAR cada una de las tecnologías y arquitecturas vinculadas a estas herramientas.
- SINTETIZAR todos los conocimientos que el uso de las mismas nos proporciona.
- DOCUMENTAR los conceptos, usos y aplicaciones que resulten patentes a lo largo del estudio.

### A.4. Alcance

Dada la profunda extensión de los conceptos relacionados con el estudio propuesto, se considerará alcanzada la meta en el momento que se cumplan las siguientes premisas:

- Las diferentes herramientas web han sido identificadas y definidas conceptualmente.
- Se han recopilado un número razonable de ejemplos de uso y/o casos de éxito.
- Se han establecido relaciones entre las herramientas disponibles y sus especializaciones, tecnologías asociadas y estándares.
- Se ha analizado en cada caso los posibles usos enfocados a la empresa y la e-administración.



- Se ha discriminado la opción más interesante para servir de fuente de un PFC.

Se establece un periodo de **cuatro meses** para alcanzar la consecución de estos objetivos. Este periodo está contemplado como fase de definición dentro del proyecto más amplio que contempla la realización del PFC.

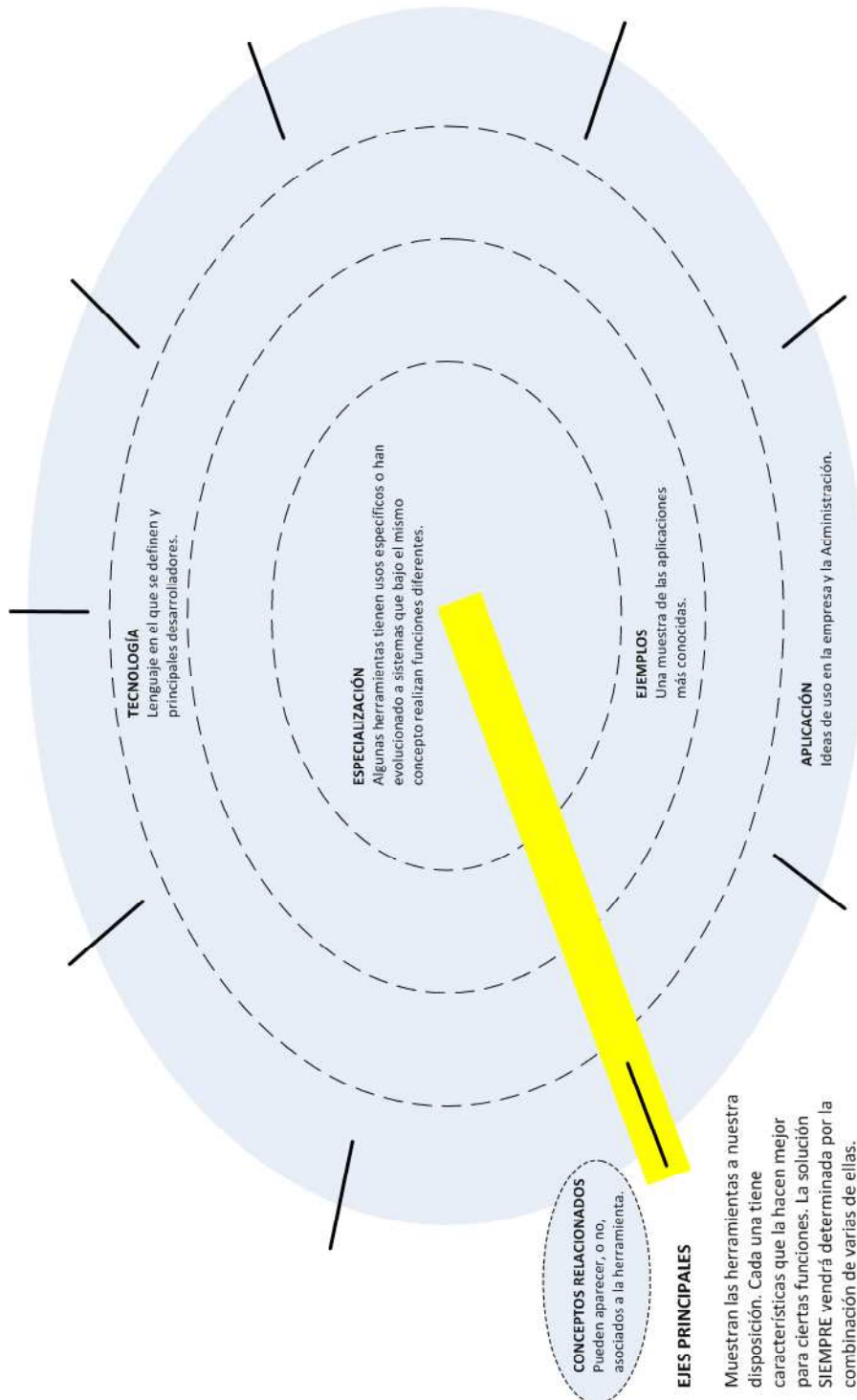
#### **A.5. Visión**

Las siguientes páginas muestran, la primera, una explicación de la distribución del esquema que se pretende presentar, y la segunda, el esquema de lo que debería contener una aplicación desarrollada con “espíritu” Web x.0. Se diferencia entre atributos, herramientas y fundamentos; el verdadero potencial se alcanza combinando varios de ellos, en función de las necesidades y del perfil del cliente final.



## ATRIBUTOS

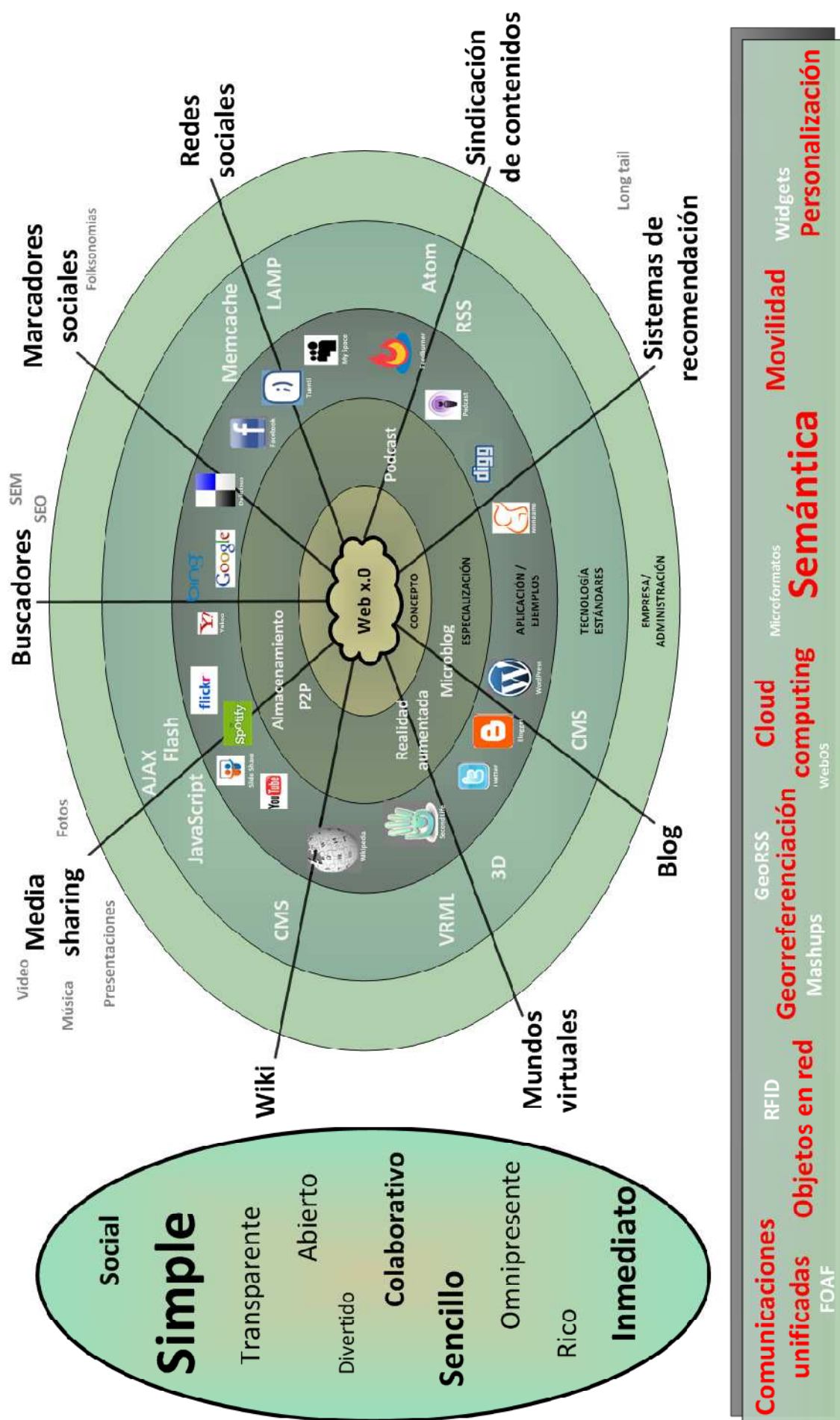
Estas son las características que debe percibir el usuario final del sistema que hemos diseñado. Idealmente, cuantas más de ellas estén presentes mejor, si bien no todas aparecen con el mismo nivel de importancia. El tamaño de cada palabra va asociado a la atención que debemos poner en aplicarla a nuestro diseño.



## FUNDAMENTOS ARQUITECTÓNICOS

Son la base sobre la que corre nuestro diseño. Constituyen el entorno de usuario y definen la accesibilidad a las herramientas que incorporemos. Su elección vendrá determinada por la manera que el cliente final quiera acceder a la aplicación. Algunas de ellas tienden a aparecer en cualquier diseño futuro (semántica) y otras sólo tienen sentido en algunos contextos (georreferenciación, comunicaciones unificadas).







### A.5.1. Atributos

El usuario que conoce y hace uso de los contenidos de la web da por hecho de manera implícita una serie de atributos que desea encontrar cuando se enfrenta a una nueva aplicación, y que decidirán si vuelve a hacer uso de ella o bien buscará alternativas que le resulten más cómodas y familiares. Estas definiciones se refieren al ideal extremo y, por tanto, no siempre alcanzable (por ejemplo, toda nueva herramienta requiere de un periodo de aprendizaje), pero nuestro diseño será tanto más bueno en cuanto más se aproxime a la idea expuesta aquí.

- **Simple:** Se debe abarcar todo su contenido de un vistazo, el entorno no debe ser recargado ni con información innecesaria. Un ejemplo sería el buscador de Google: cuadro de búsqueda sobre fondo blanco y nada más.
- **Sencillo:** Acceder y utilizar, que el proceso de aprendizaje sea mínimo y no requiera de manuales o cursos.
- **Inmediato:** Que no requiera de tiempos de espera o tediosos procesos de registro, identificación... La solución a solo un click.
- **Social:** Que se identifique con los usos, relaciones o capacidades que se utilizan en sociedad, en el día a día. Que no sea un entorno “profesionalizado”, sino que esté al alcance de las masas.
- **Omnipresente:** Desde cualquier ordenador, móvil, consola de videojuegos... El acceso no debe depender del dispositivo, tiene que estar disponible siempre.
- **Colaborativo:** Que permita opinar, crear a partir de esfuerzos colectivos, con contraste de ideas y opiniones. La individualidad limita.
- **Rico:** Que ofrezca un abanico de soluciones, sea multifunción y traiga todos los complementos para ser utilizado. Cuantos menos “extras” tengamos que ir a buscar fuera, mejor.
- **Transparente:** No debe dar la sensación de que hay información exclusiva, reservada a grupos privilegiados o explícitamente oculta al usuario.
- **Abierto:** En constante evolución, sin versión definitiva. Susceptible de que cualquier idea de mejora pueda ser incorporada sin dificultad.
- **Divertido:** Que genere interés por ser utilizado y resulte amable.

### **A.5.2. Herramientas y fundamentos**

Se presenta primero una explicación de la estructura que seguirá el documento para presentar cada uno de los elementos (denominados herramientas y fundamentos) objeto de estudio.

#### **ESQUEMA**

Ilustra de manera sencilla el concepto que se pretende transmitir de cada una de las herramientas/fundamentos.

#### **CONCEPTO**

Definición teórica de la herramienta.

#### **USOS/HERRAMIENTAS ASOCIADAS**

Una rápida enumeración de los usos característicos de cada herramienta. También se ofrece un listado de la relación más directa con otras herramientas Web.

#### **EJEMPLOS**

Recopilación de las principales plataformas existentes, además de algunos ejemplos concretos que destacan por su originalidad.



**ESPECIALIZACIÓN/TÉRMINOS RELACIONADOS**

Para aquellas herramientas que han desarrollado usos específicos, se detallan aquí sus principales características y algunos ejemplos que los ilustran.

Si existe algún concepto fuertemente relacionado con la herramienta y que merezca ser destacado, también aparece en este espacio.

**TECNOLOGÍA: ESTÁNDARES Y HERRAMIENTAS**

Se destacan aquellas tecnologías o estándares definidos para las herramientas y se enumeran algunas aplicaciones existentes para desarrollarlas.

En algunos casos destacados, se recogen aquí los esfuerzos que los grandes promotores (Microsoft, Google y otros) están realizando en la tecnología.

**EMPRESA Y ADMINISTRACIÓN**

Una descripción de los potenciales usos aplicados al mundo empresarial.

# Blog



## USOS Y FUNCIONALIDADES

- Como diario (cuaderno de bitácora)
- Sucesión de artículos
- Comentarios a las entradas editables por el administrador
- Uno o varios autores
- Clasificación de los posts por categorías
- Actualizable desde el móvil
- A menudo integrados como un servicio en las redes sociales
- Rankings de blogs
- Redes de blogs relacionados (o no)

## HERRAMIENTAS ASOCIADAS


- TAGs
- Media share
- Sindicación (o suscripción via mail)
- Mashups
- Widgets

## EJEMPLOS

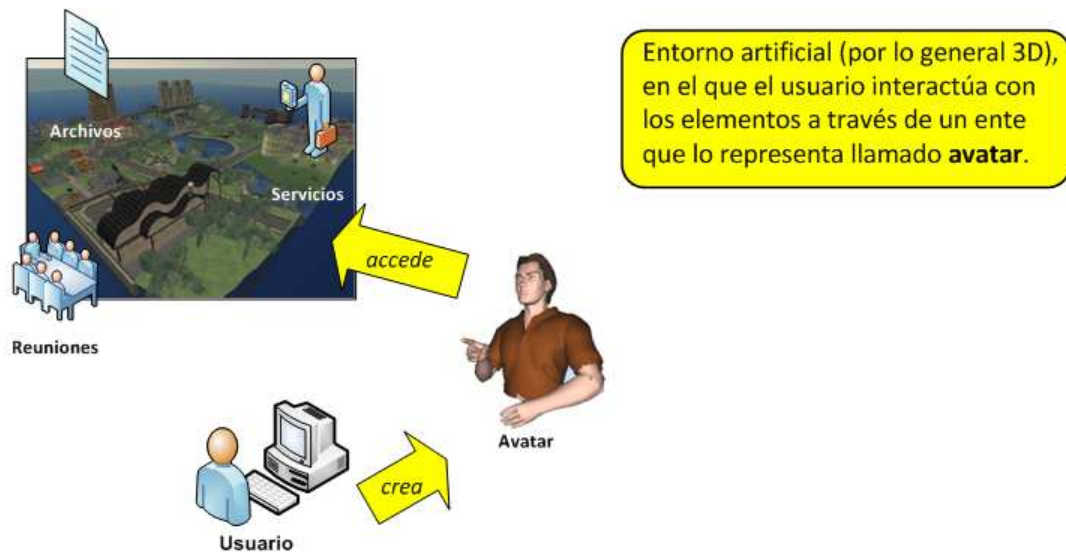
<http://technorati.com/>: El más potente buscador de la blogsfera.

<http://www.dreig.eu/caparazon/>: Un blog tecnológico gestionado por una sola persona, buen ejemplo de que seriedad y subjetividad no están reñidas, y del uso de varias de las aplicaciones Web comentadas.

[http://blogs.msdn.com/microsoft\\_office\\_word/](http://blogs.msdn.com/microsoft_office_word/): Blog corporativo de Microsoft Word.

ESPECIALIZACIÓN	<p><b>Microblog</b>          Blog de mensajes breves (alrededor de 140 caracteres) que dinamiza su uso (debido a lo rápido que se generan nuevas entradas) y facilita la actualización del mismo por SMS.          Dada su similitud con los mensajes cortos, se utiliza para convocar manifestaciones, difundir noticias rápidamente, rumores... Está relacionado con la "real-time-web" y con el Sentimet Analysis.          Algunos ejemplos:</p> <p><a href="http://twitter.com/">http://twitter.com/</a> : El mayor representante del fenómeno del microblogging, tiene como máxima el "What are you doing?"</p> <p><a href="http://www.tumblr.com/">http://www.tumblr.com/</a> : Más que microblogging, se trata de blog express. Se pueden añadir imágenes videos, presentaciones, audio, y añadir entradas de texto o de voz con un solo click. Se utiliza en la creación de lifestreams. Le da al usuario lo que pide: útil, sencillo y atractivo.</p> <p><b>Video blog (Vlog)</b>          Blog cuyas entradas están en formato de video. Se está posicionando cómo una suerte de <i>TV on demand</i> de contenidos no profesionales.</p>
TECNOLOGÍA	<p>La tecnología utilizada en la creación de blogs es <b>CMS (Content Management System)</b>, que permite separar el diseño de los contenidos.          Se diferencian 2 categorías: los servicios para blogs y las aplicaciones para blogs.</p> <p><b>- Servicios para blogs: Sistemas de creación y alojamiento</b></p>  <p>Portales web de carácter genérico que permiten a los usuarios crear blogs y que actúan de host. Facilitan la administración de los mismos haciendo transparente para el administrador la mayoría de las aplicaciones que sostienen el servicio.</p> <p><a href="https://www.blogger.com">https://www.blogger.com</a> : Servicio de blogs de Google. Permite crearlos rápidamente y de manera sencilla.</p> <p><a href="http://wordpress.com/">http://wordpress.com/</a> : Una de las mayores comunidades blog del mundo.</p> <p><b>- Aplicaciones para blogs</b>          Esta alternativa requiere estar en posesión de un dominio propio en el que alojar los blogs creados, no obstante permite mucho más control sobre las funcionalidades.</p> <p><a href="http://wordpress.org">http://wordpress.org</a> : La versión Open Source de Word Press, pone al alcance sus herramientas y amplía las opciones disponibles.</p>
EMPRESA Y ADMINISTRACIÓN	<ul style="list-style-type: none"> <li>- Recoger "experiencia cronológica" de una manera informal.</li> <li>- Seguimiento de proyectos de manera colaborativa</li> <li>- Brainstorming, consulta de ideas, mejora de propuestas</li> <li>- Información/toma de contacto para nuevas incorporaciones</li> <li>- Memoria escrita de la empresa. Repositorio clasificado de proyectos o líneas pasadas</li> </ul> <p>Una versión más extensa de los beneficios del blog en la empresa está disponible en el archivo <a href="#">blog_intranets.docx</a></p>

# Mundos virtuales



## USOS Y FUNCIONALIDADES

- Eliminar las distancias entre lugares remotos
- Aportar la variable profundidad (3D)
- Simular entornos y roles similares a la realidad
- Interactuar en tiempo real
- Realizar tareas 'pseudo-presenciales' (ej. conferencias)
- Entorno de e-learning

## HERRAMIENTAS ASOCIADAS

- Media share
- Redes sociales
- Buscador
- Mashups


## EJEMPLOS

<http://secondlife.com/>: Uno de los VW más extendidos, con moneda propia conmutable en dinero real. Permite explorar y utilizar aplicaciones integradas en él, y visitar organismos oficiales que han creado allí una réplica (y realizar gestiones asociadas a los mismos).

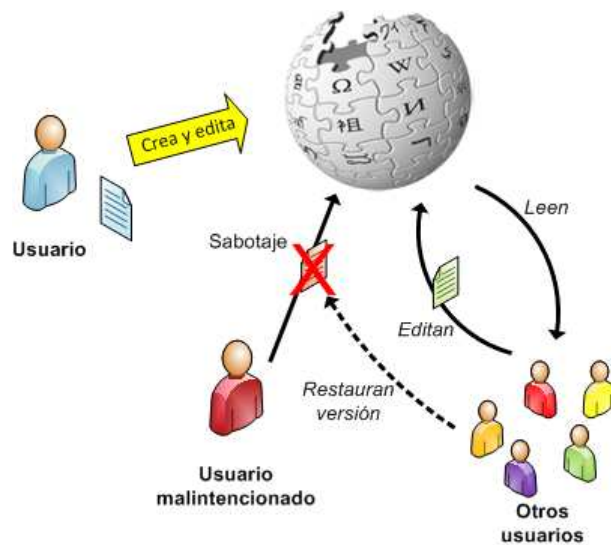
<http://www.forterrainc.com/>: Empresa que desarrolla software de mundos virtuales para profesionales. OLIVE es su herramienta más conocida.

<http://www.clubcooe.com/>: Un servicio de mensajería instantánea que muestra los avatares 3D en nuestro escritorio.



ESPECIALIZACIÓN	<p><b>Realidad aumentada</b> Aplicaciones proyectadas, mediante un interfaz, sobre el mundo físico. (GPS sobre el parabrisas, teclado sobre el escritorio, reconocimiento facial mediante móvil...)</p>  <p>Los principales esfuerzos de desarrollo de esta tecnología se enfocan actualmente en caracterizar el espacio físico en el que se desenvolverá la aplicación y tratar de independizarlo de una toma previa de datos (que el sistema sea capaz de aprender el entorno).</p> <p>Como herramienta para los desarrolladores, existe un sistema de programación llamado D.A.R.T, que permite experimentar la interacción real-virtual.</p> <p>Un ejemplo para experimentar esta tecnología (requiere webcam): <a href="http://ge.ecomagination.com/smartgrid/?c_id=Huff#/augmented_reality">http://ge.ecomagination.com/smartgrid/?c_id=Huff#/augmented_reality</a></p>
TECNOLOGÍA	<p>Las aplicaciones de mundos virtuales se crean sobre <b>motores de gráficos vectoriales (3D)</b>, generalmente heredados del mundo de los videojuegos. Second Life, por ejemplo, basa su motor gráfico en Cry Engine, de Critek. La sandbox de Second Life está disponible en Open Source.</p> <p><a href="http://www.crytek.com/downloads/technology/">http://www.crytek.com/downloads/technology/</a>: Página de Crytek.</p> <p><a href="http://opensimulator.org/wiki/Portada">http://opensimulator.org/wiki/Portada</a>: Un Open Source que te permite crear un entorno 3D de acceso online.</p> <p>Existe no obstante un estándar denominado <b>VRML</b> para el desarrollo de gráficos 3D orientados a web. El Web3D Consortium es el organismo que vela por el mismo, y se puede consultar la especificación en su página web:</p> <p><a href="http://www.web3d.org/x3d/specifications/vrml/">http://www.web3d.org/x3d/specifications/vrml/</a></p>
EMPRESA Y ADMINISTRACIÓN	<p>Crear, en un espacio virtual 3D, soluciones muy próximas al modelo mental del usuario basado en su experiencia natural.</p> <ul style="list-style-type: none"> <li>- Puntos de encuentro y espacios de reunión.</li> <li>- Espacios de aprendizaje y conferencias on-line.</li> <li>- Planos que reflejan distribuciones de planta, inventario, archivo.</li> <li>- Entornos no necesariamente reales, oficina virtual...</li> <li>- Tours virtuales</li> <li>- Agentes/guías virtuales</li> </ul> <p>Hay una serie de preguntas que podemos hacernos para ver si nuestra empresa aprovecharía un entorno 3D, en el archivo <a href="#">3D Internet for Business.pdf</a>, Pág 11</p>

# Wiki



Sitio web cuyas páginas pueden ser editadas por cualquier usuario mediante un navegador. Los títulos de los diferentes wikis quedan automáticamente referenciados entre sí a través de hipervínculos.

## USOS Y FUNCIONALIDADES

- Creación de artículos
- Edición de artículos existentes
- Creación de contenido colaborativa
- Base de datos de documentos
- Diseño de páginas web para usuarios sin nociones

## HERRAMIENTAS ASOCIADAS

- Media share
- TAGs

## EJEMPLOS

<http://www.wikipedia.org/>: Esfuerzo colaborativo mundial por crear una enciclopedia universal lo más rigurosa y fiable posible. En el momento de redactar este documento, la versión inglesa contiene más de 3.000.000 de entradas, y la española 519.000.

[http://wikitravel.org/en/Main\\_Page](http://wikitravel.org/en/Main_Page) : Proyecto para crear una guía de viajes a nivel mundial. Recopila datos geográficos, rutas, medios de transporte, hoteles, restaurantes...

<http://www.madripedia.es/wiki/Portada>: Wiki sobre Madrid. Tiene incrustados varios mashups en su página de inicio: videos de YouTube, Google Maps de Madrid...

## ESPECIALIZ.

La especialización de los wikis se refleja en los contenidos más que en diferentes herramientas. Su concepto es tan versátil que se utiliza en diversos ámbitos especializados, dónde cada comunidad lo desarrolla con las características que le resultan más útiles.

## TECNOLOGÍA

Al igual que los blogs, se basa en tecnologías **CMS**. La aplicación de la Fundación Wikimedia trabaja sobre **PHP y My SQL**.

<http://www.mediawiki.org>: Motor de Wikipedia, pretende ser la herramienta base de todos los wikis. Un aspecto importante es su extensión de la wiki semántica: <http://www.semantic-mediawiki.org>

<http://sourceforge.net/apps/trac/exe/wiki>: Wiki que ofrece aplicaciones en código abierto para aprovechar el potencial de crear páginas web.

También cómo en los blogs, podemos optar por buscar servicios de creación con alojamiento incorporado, cuyas funcionalidades serán más limitadas.

## EMPRESA Y ADMINISTRACIÓN

Edición colaborativa entre múltiples usuarios y organización de la información mediante folksonomías.

- Elaboración de manuales, tutoriales, formularios... que pueden ser corregidos y ampliados de manera sencilla y con alta disponibilidad para su consulta (directamente en la web, sin descargas ni necesidad de otros programas, incompatibilidad de formatos, etc.
- Creación colaborativa de documentos en los cuales cada editor tiene sólo una parte de la información.
- Repositorio de cualquier tipo de contenido de la empresa/administración con un sistema de índice eficiente (etiquetas y categorías) y un motor de búsqueda preciso

# Media sharing



Poner archivos de diferentes formatos a la disposición de todos o parte de los usuarios de la red para su visualización, con posibilidad de descarga o no, utilizando servidores como alojamiento remoto.

## USOS Y FUNCIONALIDADES

- Biblioteca de archivos multimedia
- Publicación de contenidos
- e-learning
- Refuerzo y enriquecimiento del mensaje
- Enriquecimiento de datos georreferenciados
- Repositorios para sitios web

## HERRAMIENTAS ASOCIADAS

- Blog
- Wiki
- Portales web
- Redes sociales
- TAGs

## EJEMPLOS

### Video

[www.youtube.com](http://www.youtube.com): El gestor de videos online más extendido, pretende ser un clasificador de todos los videos online de la red.

<http://qik.com/>: Una aplicación que permite hacer video stream desde el móvil.

<http://corp.kaltura.com/>: Plataforma de código abierto que permite editar video e insertar todas las funcionalidades del video online en un portal web.

### Fotos

<http://www.flickr.com/>: Album online que permite almacenar, clasificar y editar fotos. Entre sus características más destacadas está la selección de zonas sensibles en las fotos y la posibilidad de referenciar en un mapa el lugar en que fueron tomadas.

<http://www.panoramio.com/>: El álbum de fotos de Google. Está georreferenciado mediante Google Maps y Google Earth.

### Presentaciones

<http://www.slideshare.net/>: Portal para compartir presentaciones en PowerPoint u OpenOffice. Transforma las presentaciones a Flash y las aloja online. Permite incrustarlas en una página.

### Audio

<http://blip.fm/>: Asociado a una cuenta de las redes sociales más importantes (Twitter, Facebook o FriendFeed son algunas de las posibles), contiene una extensa base de datos de canciones que puedes clasificar y compartir.



ESPECIALIZACIÓN	<p>Los servidores o sitios web pueden venir acompañados de aplicaciones de etiquetado, búsqueda, clasificación, georreferenciación, etc.</p> <p><b>Audio</b> Un 99% del audio compartido es música. Existen cientos de emisoras y listas de reproducción online. Si lo que pretendemos es encontrar voz, el formato más utilizado son los podcast (explicado en el apartado de sindicación, pues son adjuntos a fuentes RSS).</p> <p><b>Almacenamiento web</b> Son servidores de gran capacidad que ofrecen espacio para almacenar contenido disponible para descarga. Dependiendo de si las cuentas son o no de pago, ofrecen diferentes funcionalidades.</p> <p><a href="http://www.megaupload.com/">http://www.megaupload.com/</a>: Permite almacenar hasta 1024MB y descargar hasta 1GB si no estás registrado. Los usuarios registrados obtienen privilegios, como no tener demora en sus descargas. Posee un reproductor de video streaming, Megavideo.</p> <p><a href="http://www.rapidshare.com/">http://www.rapidshare.com/</a>: Ofrece funcionalidades similares, aunque crea una cola de espera proporcional al tamaño de la última descarga.</p> <p><b>P2P</b> El intercambio peer-to-peer es otra manera de compartir archivos. Este sistema cliente-servidor se basa en un software que rastrea la red en busca del archivo que queremos descargarnos. Cualquier otro PC del mundo que tenga instalado dicho software puede actuar de fuente si posee el archivo en su disco duro. Esto permite que el ancho de banda no dependa de un único servidor al que se conectan todos los clientes, aumentando el rendimiento y la velocidad de transferencia.</p> <p><a href="http://www.emule-project.net">http://www.emule-project.net</a>: El programa de intercambio más extendido.</p> <p><a href="http://www.bittorrent.com/">http://www.bittorrent.com/</a>: Software P2P que basa parte de su arquitectura en servidor para acelerar las descargas y reducir tiempos de espera.</p>
TECNOLOGÍA	<p>La principal tecnología que soporta los archivos multimedia es <b>Flash</b>, pues su ligereza reduce significativamente los tiempos de carga.</p> <p>El lenguaje más extendido para la incursión de estos archivos, y para aumentar la cantidad de información asociada a los mismos es <b>AJAX</b>. Esto se debe a sus características de transmisión asíncrona que permite refrescar secciones de páginas web sin necesidad de realizar la descarga de la página entera desde el servidor.</p> <p>Una introducción a lo que es AJAX puede encontrarse aquí: <a href="http://adaptivepath.com/ideas/essays/archives/000385.php">http://adaptivepath.com/ideas/essays/archives/000385.php</a></p>
EMPRESA Y ADMINISTRACIÓN	<p>Incorporación de formatos más eficientes en el proceso de asimilación de conocimientos. Humanización de la comunicación.</p> <ul style="list-style-type: none"> <li>- Tutoriales/manuales</li> <li>- Repositorio para autoaprendizaje</li> <li>- Promoción, información, contacto, transmitir un mensaje desde la empresa</li> <li>- Campañas de publicidad (también en blogs, en mundos virtuales...)</li> <li>- Almacén del histórico de multimedia de la empresa, que puede ser categorizado e indexado</li> </ul>

# Buscadores



## USOS Y FUNCIONALIDADES

- Diferentes métodos de entrada
- Búsqueda temática, específica, con parámetros configurables
- La semántica incrementa sus posibilidades
- Distinta presentación de los resultados

## HERRAMIENTAS ASOCIADAS

- Blog
- Wiki
- TAGs

## EJEMPLOS

<http://www.google.es/>: El buscador más extendido, en España supone el 98% de las búsquedas.

<http://es.yahoo.com/>: Es otro de los grandes buscadores.

<http://www.bing.com/>: El buscador de Microsoft, que antes fue MSN Search y Live Search.

<http://www.oskope.com/>: En este buscador se navega mediante desplegables, y el resultado aparece en modo de iconos.

<http://www.ujiko.com/>: Clasifica los resultados por categorías que muestra agrupadas por colores y que permiten ir afinando la búsqueda a su contenido "semántico".

<http://mobile.yahoo.com/onestarch/voice>: Orientado a teléfonos móviles, esta versión del buscador de Yahoo! admite entradas mediante voz.

**SEO/SEM**

Una de las disciplinas más desarrolladas desde el punto de vista de marketing es el posicionamiento en buscadores (SEO). La filosofía es sencilla: según las estadísticas, si no apareces en la primera página de Google (u otro buscador, pero en España Google supone el 98% de las búsquedas y en EEUU, el 80%), no existes.

*Esta captura de "puntos calientes" refleja donde se centran los clicks (y por tanto, la atención) de un usuario de Google:*

**TÉRMINOS RELACIONADOS**

Siendo coherentes con la terminología, hablaríamos de SEO cuando nos referimos al posicionamiento natural en el buscador (dejando que sea el algoritmo de Google quien nos posicione) mientras que SEM sería el término para referirse a todo tipo de posicionamiento (natural y de pago). Por deformación del término, se suele utilizar SEM para referirse únicamente al método de pago (AdWords y AdSense en Google).

La mayor parte de los esfuerzos en este campo se han orientado a aplicar ingeniería inversa para tratar de averiguar el funcionamiento del PageRank de Google.

Existe numerosa bibliografía sobre el tema, aunque es necesaria la continua formación para lograr un método con razonables posibilidades de éxito.

<http://www.manualdeposicionamiento.com/guia-de-referencia-seo/>

Un nombre propio es Matt Cutts, responsable del servidor de Google, quien se encarga de desmentir o aclarar cada cierto tiempo las directrices adoptadas por los que realizan SEO.

**TECNOLOGÍA**

Programar un buscador implica navegar por bases de datos. Como ejemplo, un manual para programar un buscador en **PHP y MySQL**:

<http://www.desarrolloweb.com/articulos/2087.php>

Una herramienta importante a la hora de generar una base de datos para un buscador web son los crawlers. Un **crawler** es un programa que rastrea la WWW almacenando URLs. Se pueden encontrar numerosos ejemplos. Uno de ellos:

<http://www.cs.cmu.edu/~rcm/websphinx/>

**EMPRESA Y ADMIN.**

Eficiencia (productividad) y eficacia (calidad de resultados) en el acceso a los activos de conocimiento.

- Marketing, hoy por hoy el SEO es la mejor estrategia para ser 'visible' en la red
- Buscadores semánticos suponen búsquedas más precisas y más variadas en el formato de los resultados (documentos, video, audio). Esto facilita la Gestión del Conocimiento

# Marcadores sociales



## USOS Y FUNCIONALIDADES

- Clasificación del contenido por temas
- Filtrado subjetivo
- Indexado para futuras búsquedas
- Nubes de palabras

## HERRAMIENTAS ASOCIADAS

- Blogs
- Buscadores
- Media share
- Wiki

## EJEMPLOS

<http://delicious.com/>: El motor de etiquetado web más extendido. Permite guardar vínculos y categorizarlos mediante tags. El valor añadido que aporta es poder compartir públicamente los etiquetados, de modo que podemos aprovechar el esfuerzo clasificador realizado por otros.

<http://www.diigo.com/>: Además de bookmarking permite subrayar y añadir notas (Post-It) a páginas web visitadas.

## TÉRMINOS RELACIONADOS

**Folksonomías**

Podría traducirse como 'etiquetado social'. Se denomina así al proceso de categorización realizado por cualquier usuario o comunidad de cualquier contenido de la red y , sobre todo, su compartición pública. Las características principales son:

- No es un método riguroso, basado en normas o realizado por expertos.
- Es un método potente, sus agentes son todos los usuarios de la web.
- Puede ser un método acotado a un entorno (página, portal...) o a toda la red, mediante comunidades como Delcious.

## TECNOLOGÍA

El referente en etiquetado (delicious) utiliza sencillo código **HTML y sindicación RSS** para cumplir su cometido de ser un bookmark social y compartido.

Más interesante resulta analizar las herramientas de etiquetado semántico, llamadas así por analizar el significado del texto del que les proveemos para sugerirnos etiquetas relacionadas.

<http://www.opencalais.com/>: Analiza el significado de un texto y aporta tags, imágenes y videos relacionados.

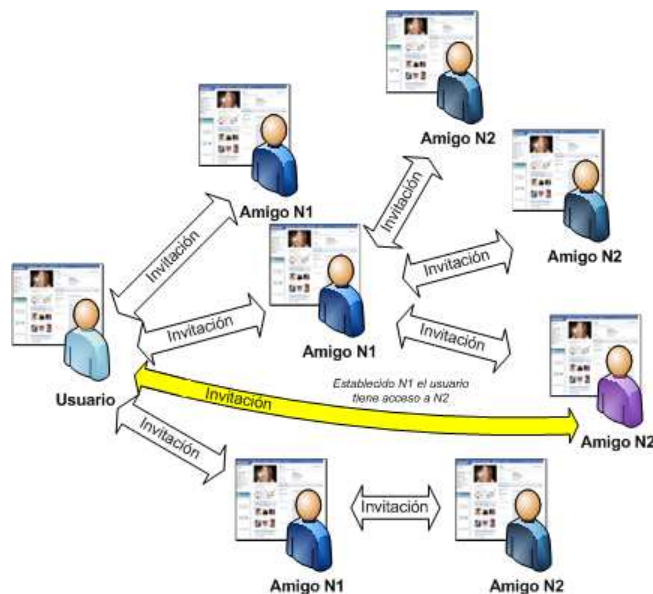
EMPRESA  
Y ADMIN.

En un sistema basado en folksonomía, mediante un etiquetado inteligente y humano, mejora la eficacia de la metainformación.

- Un sistema de etiquetado global independiza el tipo de contenido de los resultados obtenidos en las búsquedas (es decir, para una búsqueda completa, te devuelve todo lo asociado a esa(s) palabra(s)
- Crear un repositorio de links común (en línea)
- Ayuda a la navegación y sugerencias a través de nubes de palabras



# Redes sociales



Portales web que permiten a un usuario crear un perfil personal que puede actualizar con texto y adjuntos, mostrar a otros usuarios y recibir de estos sus propios perfiles.

## USOS Y FUNCIONALIDADES

- Compartición de contenidos
- Creación de comunidades
- Cooperación
- Agenda de contactos
- Panel de anuncios (eventos)
- Transmisión viral de mensajes

## HERRAMIENTAS ASOCIADAS

- Blogs
- Media sharing
- Sindicación de contenidos
- TAGs
- Wiki

## EJEMPLOS

<http://www.xing.com/>: Red social entre profesionales. Incluye la posibilidad de publicar tu perfil (CV), la gestión de agenda, eventos y diferentes motores de búsqueda.

<http://www.facebook.com/>: Una de las redes sociales más extendidas. Diversas corporaciones la utilizan para acercarse a los clientes, escuchar, proponer. Es una alternativa a los complejos estudios de mercado, acudiendo directamente a la fuente.

<http://www.twine.com/>: Una red social semántica. Prolonga la potencialidad de las redes conocidas estableciendo relaciones en función del perfil, los contenidos aportados...

## ESPECIALIZACIÓN

Aunque a nivel estructural no haya diferencias, existen dos subgrupos conceptuales muy diferenciados entre las redes sociales: las redes **generalistas** o **personales**, y las **especializadas** o **profesionales**. Las primeras se caracterizan por ser informales, en ellas el usuario decide los contenidos y usos, y es él quien pone los límites. En las segundas, existen ciertas normas, restricciones a la hora de establecer relaciones o generar contenidos y suelen tener una serie de recomendaciones de buen uso allí donde no se puede limitar el software.

Mientras que las redes personales están asociadas al ocio, las profesionales se están mostrando cómo una fuerte herramienta para gestionar las relaciones laborales o como comunidades de expertos con intereses comunes.

## TECNOLOGÍA

La herramienta más conocida, Facebook, fue originalmente programada en **PHP y MySQL**. El secreto de su velocidad es **Memcache+LAMP**.

<http://www.danga.com/memcached/>

Sus desarrolladores van ingeniando día a día maneras de optimizarlo (apoyados en C++, Java, Erlang...). Para esto, dados sus fallidos intentos de migrar a lenguajes como Phyton, han adaptado un sistema de compatibilidad de lenguajes llamado **Thrift**.

<http://incubator.apache.org/thrift/>

Como en muchas otras aplicaciones de la web social, Facebook tiene un foro de desarrolladores (en este caso, una wiki) donde tienes acceso a una parte de su API, generalmente para desarrollar aplicaciones.

[http://wiki.developers.facebook.com/index.php/Main\\_Page](http://wiki.developers.facebook.com/index.php/Main_Page)

## EMPRESA Y ADMINISTRACIÓN

Proyección digital y promoción de la faceta profesional / empresarial en una red administrada y vinculada. Soporte técnico al Social Networking. Desde el punto de vista de gestión del conocimiento, crea una red de nodos de conocimiento tácito.

- Gestionar la red interna de personal basándola en perfiles, ricos en contenido (qué está haciendo en cada momento, hilos sobre sus proyectos, lista de contactos/recursos con listas de correo internas, links a mail, teléfono, etc., integración con servicios de mensajería instantánea).

Es interesante la visión que, junto blogs y wikis, se nos presenta en el archivo [Web 2.0 Talent Management and employee engagement.pdf](#)

# Sindicación de contenidos



## USOS Y FUNCIONALIDADES

- Generar contenido actualizado en tiempo real
- Calendarios, eventos y agendas
- Filtrado de contenidos
- Radar de opinión, noticias...
- Agrupar varias fuentes de información en un solo lector

## HERRAMIENTAS ASOCIADAS

- Blog
- Wiki
- Media share
- Redes sociales
- En general, cualquier herramienta a la que tenga sentido agregar una alerta

## EJEMPLOS

<https://www.google.com/accounts/ServiceLogin?service=feedburner> : Generador de fuentes RSS. Feedburner era una aplicación independiente actualmente absorbida por Google.

<http://www.rsscalendar.com/> : Permite crear calendarios compartidos sindicados.

<http://www.simpletracking.com/> : Te permite el seguimiento de un paquete DHL, FedEx o del US Postal Service introduciendo el nº de referencia del envío.

<http://www.dapper.net/> : Permite recortar contenidos de la web y convertirlos en feeds, insertarlos en otra web... Básicamente implementa el *copy-paste* en los navegadores.



**Podcast**  
 Archivo multimedia agregado a una fuente web. No se envía el contenido, sino que se genera un hipervínculo. Los formatos más utilizados son **mp3**, **aac** y **ogg**.

El uso más extendido es el de archivos de audio, generando una radio online, en la que cada usuario difunde aquello que le interesa. Si contiene video, se llama **vodcast**.

Como ejemplo, el podcast de Cadena SER:  
<http://www.cadenaser.com/podcast/>

**Diagrama:** Muestra el flujo de un podcast. Los 'Contenidos disponibles en la web' permiten una 'Suscripción previa' a un 'PC'. El PC realiza una 'Descarga de contenido' y, si el dispositivo lo permite, una 'descarga directa' a un 'iPOD MP4'. También se muestra una 'Copia offline' desde el PC al iPOD.

Existen 2 tecnologías que nos permiten syndicar contenidos: RSS y Atom

**RSS**  
 Con base en XML, dependiendo de la versión sus siglas tienen significados diferentes, debido a que el protocolo se orienta a distintas funciones:

- *Rich Site Summary* (Versiones 0.91, 0.92, 0.93 y 0.94), y posteriormente *Really Simple Syndication* (Versión 2.0): Se refieren a la especificación sobre XML **no-RDF**.
- *RDF site Summary* (Versiones 0.9 y 1.0): Suponen la versión **RDF** de la especificación.

La especificación del estandar puede encontrarse en el siguiente enlace:  
<http://www.rssboard.org/rss-specification>

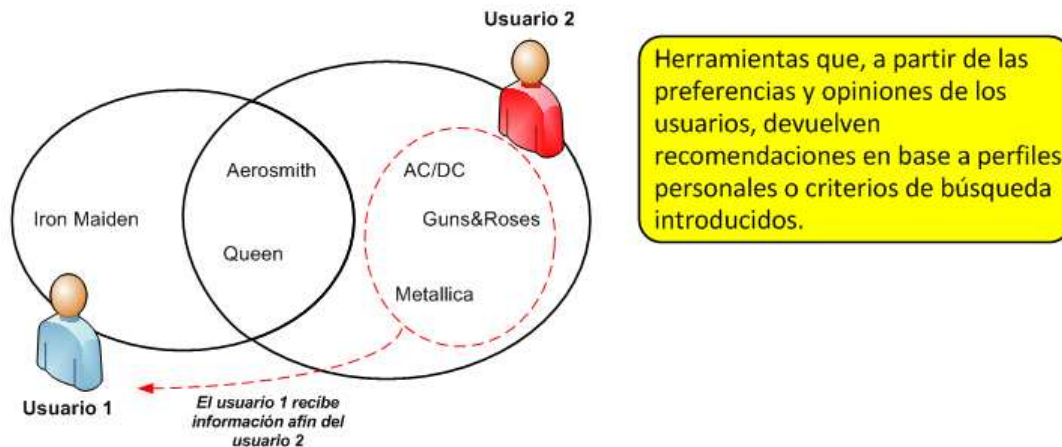
**ATOM**  
 También basado en XML, nació tratando de mejorar las limitaciones de interoperabilidad de RSS. La especificación de los estandarés es la siguiente:

- Sindicación: <http://tools.ietf.org/html/rfc4287>
- Protocolo de publicación (Actualización de sitios web): <http://tools.ietf.org/html/rfc5023>

Proyección de contenidos a través de canales temáticos. Permite separar la generación y la explotación de contenidos. En destino, permite realizar agregaciones multicanal.

- Mediante podcast difusión de manuales y tutoriales, o bloques de noticias o instrucciones que el usuario puede escuchar cuando mejor le convenga
- Junto con algunas de las otras tecnologías puede ser un sistema de alertas ante actualizaciones de información relevante (estado de proyectos...)
- Se puede seleccionar un compendio de fuentes de referencia y visualizarlas juntas. Cotejar distintos puntos de vista, establecer tendencias, comparar posturas...

# Sistemas de recomendación



## USOS Y FUNCIONALIDADES

- Filtrado de contenido
- Navegación eficiente
- Estudio de tendencias
- Especialización mediante el uso
- Asistencia, navegación dirigida

## HERRAMIENTAS ASOCIADAS

- Redes sociales
- TAGs
- Buscadores

## EJEMPLOS

<http://meneame.net/>: Sistema de recomendación de noticias, en el que los usuarios proponen y votan noticias de cualquier tipo

<http://www.hunch.com/>: Mediante el sistema Q20 (20 preguntas) es capaz de recomendarte sobre cualquier tema. Si te registras vas aprendiendo.


<http://www.strands.com/>: Sistema de recomendación para deportistas. Genera rutas de entrenamiento, tablas, series y gustos sobre webs especializadas o recursos en Internet.

<http://aggregateknowledge.com/>: Sistema que orienta hacia qué público debes dirigir tus esfuerzos dependiendo el tipo de producto que generes.

<http://www.amazon.com/>: Esta conocida librería de Internet amplía día a día su sistema de recomendación apoyándose en los comentarios y votaciones de los propios compradores.

**TÉRMINOS RELACIONADOS**

**The long tail**



Este fenómeno estadístico (la larga cola) se refiere a un elevado número de elementos que aparecen con muy poca frecuencia. En el mercado, suponen artículos especializados o raros, orientados a un reducido número de usuarios. Gracias a los sistemas de recomendación, la atención sobre este tipo de productos ha aumentado enormemente hasta el punto de ser tan significativos como los 'best sellers'.

\*Picture by Hay Kranen / PD\*

(\*) El término también es utilizado en SEO para referirse a cadenas cada vez más largas de palabras clave que ofrecen resultados más filtrados (específicos).

**TECNOLOGÍA**

Un apartado determinante de un sistema de recomendación es el algoritmo de decisión utilizado. Un manual teórico sobre los algoritmos de recomendación:  
<http://www.hipertext.net/web/pag227.htm>

Y una guía que describe la interacción con el usuario en sistemas de recomendación web:  
<http://www.hipertext.net/web/pag227.htm>

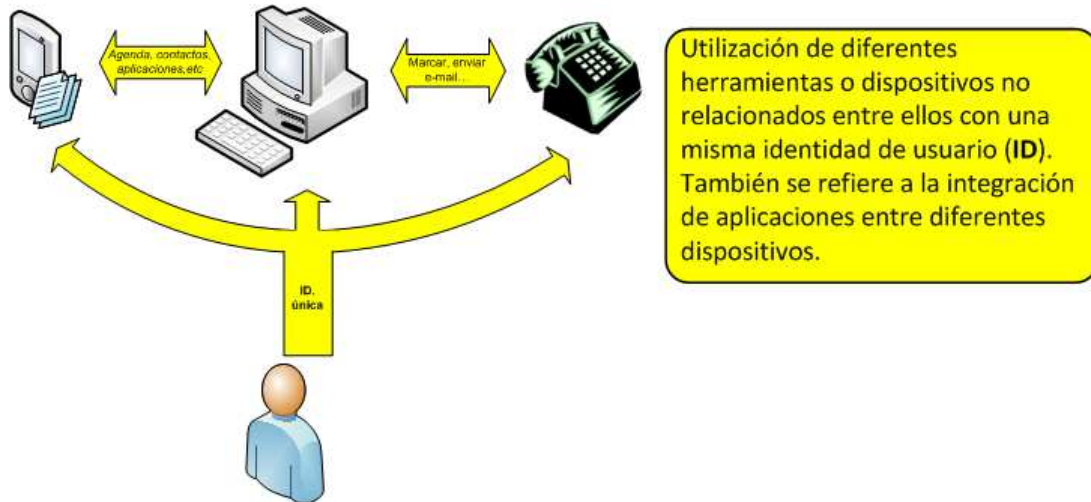
El proceso completo de diseño e implementación de un sistema de recomendación puede ser visto en este documento, tesis de graduación de unas estudiantes de la Escuela Politécnica del Litoral en Guayaquil, Ecuador:  
<http://www.dspace.espol.edu.ec/bitstream/123456789/6359/1/An%C3%A1lisis,%20dise%C3%B1o%20e%20implementaci%C3%B3n%20de%20un%20sistema%20adaptivo.pdf>

**EMPRESA Y ADMINISTRACIÓN**

Introduce la democracia electrónica. El usuario opina y recomienda. Distribuye el sistema de autoridad y se muestra como alternativa a los canales oficiales de opinión.

- Las aplicaciones se centran sobre todo en el contacto con el cliente. Se puede utilizar para conocer sus gustos y focalizar la publicidad. También para ofrecerle servicios más a medida que mejore su fidelización a la marca.
- En una intranet se puede utilizar para establecer tendencias en el uso de herramientas internas, control de stock (items que se repiten más o menos)...

# Comunicaciones unificadas



## USOS Y FUNCIONALIDADES

- Identificación única (DNI virtual)
- Creación de un *profile* al que se asocia todo el contenido generado por un usuario concreto (*lifestreaming*)
- Localización independientemente del lugar físico ('*presente en el sistema*' o '*ausente del sistema*')
- Gestión eficiente de los contactos

## EJEMPLOS

<http://openid.es/>: Servicio que genera un ID único (URL o XRI) para todas aquellas aplicaciones web que lo soporten.

<http://www.microsoft.com/latam/uc/default.aspx>: Información sobre Unified Communications de Microsoft, aplicación que auna correo electrónico y de voz, calendario, mensajería instantánea VoIP, audio, video y conferencia web.

<https://fuser.com/default.aspx>: Unifica servidores de correo y redes sociales (Outlook, Messenger, Hotmail, Facebook...) en una sola bandeja de entrada. No almacena contenidos, sólo los indexa.



## TÉRMINOS RELACIONADOS

**FOAF**

FOAF (Friend Of A Friend) es un proyecto de estandarización en lenguaje semántico que pretende identificar a un usuario en la web a través de todos sus perfiles activos en las redes sociales. Al ser semántico, se basa en que sean las máquinas las que reconozcan el ID como único (mediante una etiqueta en el código fuente) sin necesidad de que una persona o automata virtual se encarguen de clasificar explícitamente los contenidos creados.

La especificación puede ser consultada aquí:

<http://xmlns.com/foaf/0.1/>

## TECNOLOGÍA

Dependiendo del nivel de unificación que deseemos tenemos varias tecnologías que considerar:

- **Unificación del hardware:** Generalmente se trata de unificar la red telefónica (conmutada) y la red de datos (IP). Una solución es utilizar VoIP para las comunicaciones y, así, tener toda la red unificada en tecnología IP. Otra opción es compatibilizar ambas redes mediante software (Unified Communications de Microsoft).

- **Unificación del software (ID único):** Comunicaciones Unificadas de Microsoft, como referente, basa su tecnología en los protocolos entre servidores. Convierte básicamente todas las comunicaciones entrantes en tráfico SMTP, concentrándolo en el servidor de correo del usuario. A través de servicio de centralita gestiona la comunicación de cualquier usuario, interno o externo, a partir de la información disponible en el Directorio Activo.

Cisco compete con Microsoft por posicionarse a la cabeza de UC. El primero explota más las posibilidades hardware y el segundo, como se ha podido ver, apuesta por el software. IBM también desarrolla un paquete competitivo en este ámbito.

**OpenID** es una propuesta para crear un identificador único en la red. Google, Yahoo!, Flickr o WordPress ya lo soportan. La especificación de la tecnología se puede obtener en la siguiente wiki:

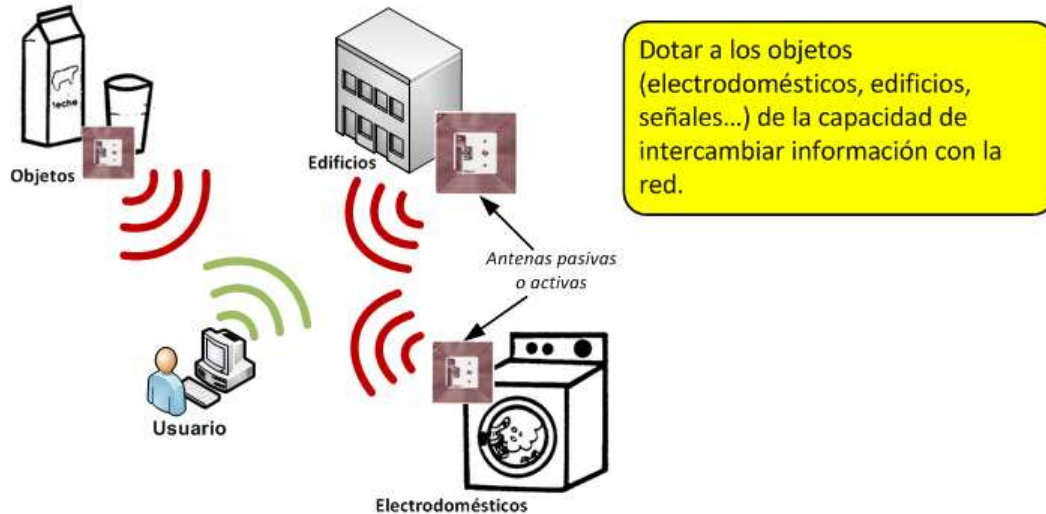
<http://wiki.openid.net/>

EMPRESA  
Y ADMIN.

Crea el contacto universal multisistema y multidispositivo. Abstrae al usuario de los sistemas y dispositivos. Se centra en lo importante, la comunicación entre personas. Eficiencia (costes) y eficacia (calidad) en la utilización de los sistemas de comunicación.

- Agenda de contactos internos universal e independiente del dispositivo
- Red de datos y voz unificada, gestión y tarificación única

# Objetos en red



## USOS Y FUNCIONALIDADES

- Reflejo en la red de los elementos del mundo real.
- Automatización de tareas
- Conversaciones M2M (Máquina a máquina)

## EJEMPLOS

[http://www.violet.net/index\\_en.html](http://www.violet.net/index_en.html): Página de la empresa Violet donde muestran sus productos sobre internet de las cosas para uso doméstico, RFID tags y dispositivos para su lectura con el PC.

## TÉRMINOS RELACIONADOS

No es necesario esperar a que cada objeto tenga una etiqueta RFID y un ID único en la red para poder trabajar con él en Internet. Aquí entra en juego la ya nombrada realidad aumentada: mediante una imagen, un sistema de reconocimiento y un dispositivo (por ejemplo, un móvil) podemos ‘reconocer’ un ítem y acceder a su información.

**Arduino**

Esta plataforma de hardware libre se perfila hoy por hoy como un interesante impulso del Internet de las cosas gracias a su interoperabilidad y la posibilidad de realizar prototipos sin un despliegue económico elevado. El software disponible para el desarrollo de Arduino está disponible tanto para Windows como para Linux. Algunos de los interfaces utilizados para programar una placa Arduino son puerto serie, USB, Bluetooth, XBee y ArCan. Aunque existen muchas iniciativas, la página del Proyecto Arduino:

<http://arduino.cc/>

## TECNOLOGÍA

La principal tecnología actual es el sistema de identificado por radiofrecuencia (**RFID**). Se basa en el uso de pequeñas antenas de tamaño reducido y extra plano que adheridas a los objetos son capaces de transmitir un identificador único. Esto, integrado en una red, permite a los sistemas que la componen interactuar con el objeto detectando su presencia en el entorno.

Los estándares de frecuencia son 125 kHz (original) y 134,5 kHz (internacional).

El dispositivo de auto-pago Via-T para autopistas, o la tarjeta bus TUZSA, son ejemplos de tecnología RFID.

Otra tecnología, más asociada al desarrollo domótico, es **ZigBee**. Ésta, de la familia del Bluetooth (WPAN) se diferencia con él en que requiere mucha menos electrónica (el 10% en comparación) pero mucho más código (un 50% más, como referencia). Basada en el estándar 802.15.4, su especificación se puede leer en castellano en Wikipedia:

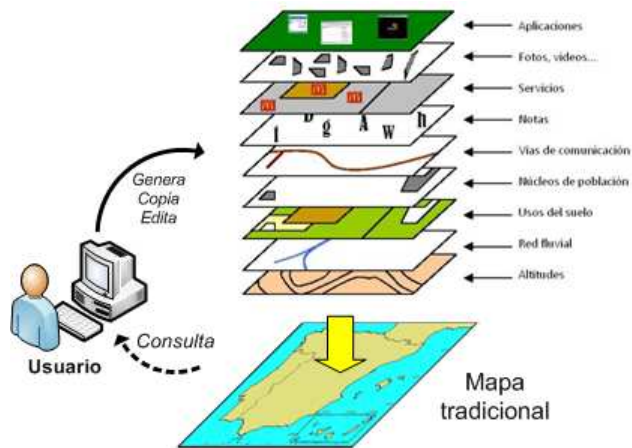
[http://es.wikipedia.org/wiki/ZigBee\\_\(especificaci%C3%B3n\)](http://es.wikipedia.org/wiki/ZigBee_(especificaci%C3%B3n))

EMPRESA  
Y ADMIN.

Introduce la capacidad de que los “objetos” se comuniquen y colaboren entre sí.

- Control automatizado de inventarios
- Domótica para el control de las oficinas
- Control de acceso de empleados/personal

# Georreferenciación



## USOS Y FUNCIONALIDADES

- Referencia espacial de los datos
- Filtrado de la información en función de la importancia debida a la distancia
- Contextualización espacial de los datos con contenidos relacionados o metainformación útil.

## EJEMPLOS

<http://earth.google.es/>: Integra wikipedia y panoramio, de modo que puedes obtener información relacionada con los lugares al clicar en ellos, y ver fotos del lugar. Incluye modelos 3D de edificios (que generan los usuarios). Incluye 'vuelos' sobre el mapa, enlaces a la página web de los servicios referenciados en sus mapas...

<http://www.openstreetmap.es/>: Proyecto dirigido a ofrecer recursos geográficos libres, debido a que la gran mayoría de los mapas existentes tienen restricciones legales o técnicas.

[http://gis.chicagopolice.org/CLEARMap\\_crime\\_sums/startPage.htm](http://gis.chicagopolice.org/CLEARMap_crime_sums/startPage.htm): Página de la policía de Chicago que te permite consultar crímenes por barrio, por tipo de crimen además de introducir crímenes sufridos mediante los mismos criterios incluso recibir alarmas dependiendo del lugar por el que te muevas. Todo ello referenciado y con posibilidad de ser consultado en un mapa de la ciudad.

<http://centrl.com/>: Aplicación que permite ingresar con tu perfil de las principales redes sociales, mostrar tu ubicación, ver la de otros y filtrar resultados en función de la distancia (te señala los servicios cercanos).



ESPECIALIZACIÓN	<p><b>GeoRSS</b></p> <p>Un caso particular de sindicación es aquella en la que se adjuntan coordenadas geográficas al feed, de modo que este puede ser visualizado en su contexto espacial. Una posible aplicación permitiría, una vez localizado al usuario, filtrar la información que se le ofrece en base a la relevancia que esta tuviera por proximidad a él.</p> <p>Existen al menos 3 estándares para syndicar mediante GeoRSS:</p> <ul style="list-style-type: none"> <li>- W3C Geo</li> <li>- Simple GeoRSS</li> <li>- Pro RSS (GML)</li> </ul>
TÉRMINOS RELACIONADOS	<p><b>Mashups</b></p> <p>El verdadero potencial de los sistemas SIG aparece al fusionar la información georreferenciada con otras aplicaciones o herramientas web. A esta fusión de herramientas existentes para crear funcionalidades nuevas se la conoce como mashups. Se generan utilizando APIs de aplicaciones existentes.</p> <p>Una recopilación de varios de ellos:  <a href="http://gmapsmania.googlepages.com/100thingstodowithgooglemapsmashups">http://gmapsmania.googlepages.com/100thingstodowithgooglemapsmashups</a></p> <p>Y un ejemplo-tutorial de la creación de un mashup:  <a href="http://franchu.net/2007/06/02/como-hacer-un-mashup-y-no-morir-en-el-intento/">http://franchu.net/2007/06/02/como-hacer-un-mashup-y-no-morir-en-el-intento/</a></p>

TECNOLOGÍA	<p><b>KML</b> es el lenguaje desarrollado por Google para definir sus aplicaciones sobre Google Earth. Ha sido adoptado por el Open Geospatial Consortium (<a href="http://www.osgeo.org">www.osgeo.org</a>), erigiéndose en estándar (a partir de su versión 2.2).</p> <p>Una referencia sobre KML:  <a href="http://code.google.com/intl/es/apis/kml/documentation/kmlreference.html">http://code.google.com/intl/es/apis/kml/documentation/kmlreference.html</a></p> <p>El precursor de KML fue GML.</p>
------------	---

EMPRESA Y ADMIN.	<p>Establece una relación directa entre dos dimensiones, la información y el espacio. Aproxima el mundo de la información al mundo de la persona, que transcurre en el espacio. Facilita la abstracción y asimilación de conocimientos y agiliza la toma de decisiones.</p> <ul style="list-style-type: none"> <li>-Inventario georreferenciado de documentos, activos...</li> <li>- Sistemas de información localizada en el espacio para empleados/clientes</li> <li>- Alarmas</li> </ul>
------------------	---

# Cloud computing



Trabajar con un escritorio online, basado en aplicaciones online y con los documentos almacenados en servidores. La idea es trasladar el PC a la red.

## USOS Y FUNCIONALIDADES

- Base de datos de aplicaciones ilimitada (WWW) accesible según las necesidades
- Acceso independiente de la ubicación
- Liberación de las tareas de mantenimiento y/o almacenamiento
- Escalabilidad de los recursos según las necesidades

**Riesgo:** La seguridad, la vulnerabilidad de los sistemas existentes y el peligro de confiar toda la información a la WWW.

## EJEMPLOS

<http://www.salesforce.com/es/>: Una de las principales proveedoras de SaaS en la Red. Tiene su versión PaaS en la plataforma Force.com

<http://aws.amazon.com/>: La página de Servicios Web de Amazon, que se ha posicionado como uno de los mas potentes y eficientes proveedores de Cloud Services.

[www.hotmail.com](http://www.hotmail.com): El correo web de Microsoft, nos permite almacenamiento de varios GB, la posibilidad de ordenar nuestras bandejas, gestionar nuestro correo, modificar nuestro perfil...

[www.zoho.com](http://www.zoho.com) : Suite tipo Office disponible online.

<http://www.google.com/apps/intl/en/business/index.html>: Google ofrece una serie de servicios online: Gmail, Google Calendar, Google Docs, Google Sites...

**WebOS**

ESPECIALIZACIÓN

En la línea de ‘software en la nube’, nace la idea de un sistema operativo basado en web. Con él, no tenemos que preocuparnos de que potencia o tipo de máquina remota (PC, móvil, PDA) estemos utilizando, accedemos desde cualquiera de ellos al sistema con el que trabajamos y que tenemos ordenado según nuestras necesidades.

- Ofrece independencia de la máquina desde la que se accede.
- Permite la colaboración remota y la compartición de recursos inmediata.
- Mejora la compatibilidad. No hay que preocuparse de las versiones de las aplicaciones, porque siempre se accede desde las mismas.

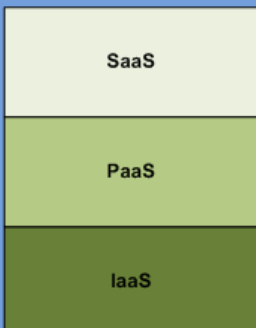
<http://es.eyeos.com>: WebOS desarrollado en España, con versiones Open Source y comercial.

La pila **LAMP** (Linux+Apache+MySQL+PHP/Perl/Python) es actualmente la plataforma preferida por los principales desarrolladores (SUN Microsystems destaca entre ellos).

Microsoft, Amazon y Google están enfocando sus esfuerzos en crear aplicaciones online (en la nube). El ejemplo más reciente es Google Wave, muy relacionado con Web 2.0 y todavía en periodo de prueba. La alternativa de Microsoft es Azure, PaaS orientado a desarrolladores que pretendan crear contenidos *cloud*. Amazon por su parte, lleva meses desarrollando sus AWS (Web Services) con EC2 como plataforma, que también pretenden perfilarse como PaaS.

La arquitectura de los productos ‘en la nube’ sigue un esquema de 3 capas:

TECNOLOGÍA



**SaaS (Software as a Service):** Se refiere a una aplicación completa ofrecida como solución, bajo demanda y soportada en la infraestructura del proveedor. Un ejemplo sería Salesforce.com.

**PaaS (Platform as a Service):** Incluye un paquete de desarrollo completo para la creación de aplicaciones (SO, servidor y entorno). Por ejemplo, Google App Engine.

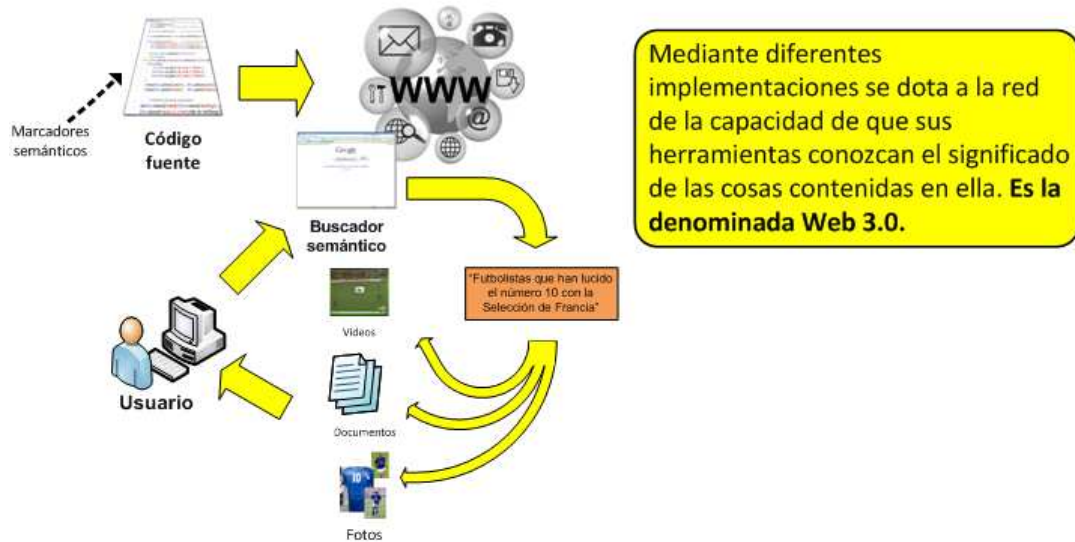
**IaaS (Infrastructure as a Service):** Las necesidades de capacidad de computación y almacenamiento son ofrecidas como servicio. Los AWS EC2 y S3 son ejemplos de esta capa.

EMPRESA Y ADMINISTRACIÓN

Consumir cualquier recurso TIC como servicio. Extiende el modelo SOA hasta las capas más bajas una solución TIC. Ofrece un modelo mucho más “elástico”.

- Gestión de las aplicaciones a través de la nube para evitar costes de mantenimiento de servidores, almacenamiento, licencias...
- Creación de una nube interna para unificar y centralizar el mantenimiento, actualización, etc. de aplicaciones

# Web semántica



## USOS Y FUNCIONALIDADES

- Búsquedas más precisas
- Expresión en lenguaje natural
- Máquinas más inteligentes, capaces de filtrar seleccionar y despreciar contenidos con criterio más parecido al razonamiento humano
- Red totalmente interconectada, elementos identificados unívocamente independientemente de la página que los contenga

## EJEMPLOS

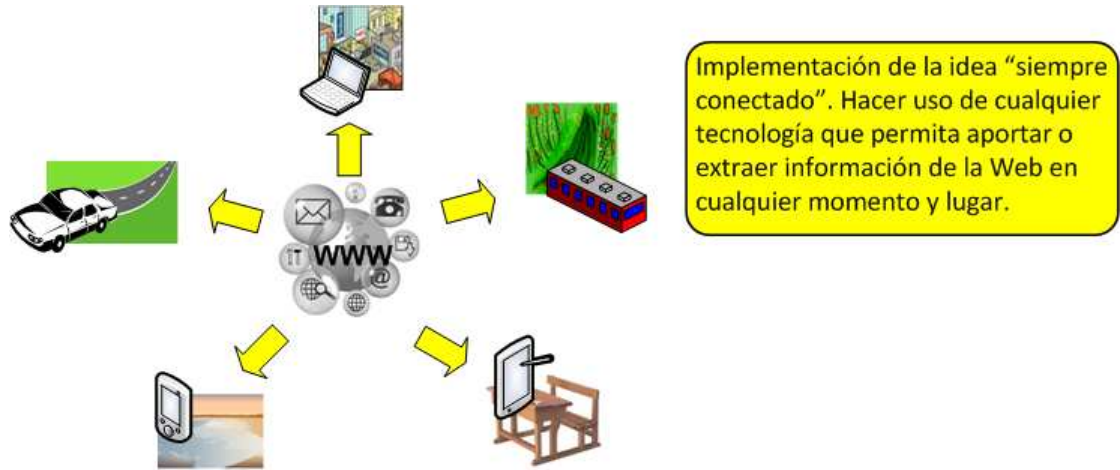
<http://dbpedia.org/>: Proyecto de semantización de la Wikipedia, que forma parte de la Linked Data Web.

<http://www.w3.org/2001/sw/>: Página del W3C llenada de ejemplos, presentaciones y entrevistas sobre web semántica.

TÉRMINOS RELACIONADOS	<p><b>Microformatos, ‘la web semántica con minúsculas’</b></p> <p>Ante la necesaria ‘semantización’ de las toneladas de contenido ya existente en la red, han surgido muchas propuestas para adaptar el código generado con los antiguos lenguajes de modo que las máquinas interpreten su semántica. El más extendido son los microformatos, ‘paquetes’ de código fácilmente integrables en HTML. <b>NO</b> es un nuevo lenguaje, y se utiliza para resolver problemas concretos. Su mayor problema es la reusabilidad, debido a que cada desarrollador puede crear su propio modelo. Para mejorar esto, existen organizaciones que promueven recomendaciones de formato.</p> <p><a href="http://microformats.org/">http://microformats.org/</a>: Es la organización que crea las especificaciones y se preocupa de crear un lenguaje de marcado consensuado que sea útil para la ‘semantización’ de la web.</p> <p><b>Linked Data Web ó Linking Open Data</b></p> <p>La ‘red de los datos enlazados’ es un esfuerzo, dentro del W3C, por interconectar bases de datos importantes a fin de crear un repositorio ‘global’ que sirva a las máquinas con inteligencia semántica para establecer relaciones entre los mismos. Es decir, si queremos llegar a la web semántica, las bases de datos deben ser abiertas (y no silos de información como son actualmente) de modo que los robots de la web puedan interrelacionar conceptos saltando de una a otra. El gráfico del proyecto ha ido creciendo desde su nacimiento, ésta es una versión de julio de 2009:</p> <p><a href="http://www4.wiwi.fu-berlin.de/bizer/pub/lod-datasets_2009-07-14.html">http://www4.wiwi.fu-berlin.de/bizer/pub/lod-datasets_2009-07-14.html</a></p>
TECNOLOGÍA	<p>Los lenguajes-estándares que están ayudando a construir la web semántica son los siguientes:</p> <ul style="list-style-type: none"> <li>- <b>XML (eXtended Markup Language)</b>: Su sistema de etiquetas permite mostrar algo de significado a las máquinas.</li> <li>- <b>XML schema</b>: Jerarquiza las relaciones entre funciones de XML.</li> <li>- <b>RDF (Resource Description Framework)</b>: Convierte las definiciones en conjuntos sujeto-predicado-objeto (tripletes). Junto con schema y OWL, permite relaciones semánticas entre los datos.</li> <li>- <b>RDF schema</b>: Jerarquiza los recursos RDF.</li> <li>- <b>OWL (Ontology Web Language)</b>: Define relaciones entre objetos.</li> <li>- <b>SPARQL (Simple Protocol And RDF Query Language)</b>: Lenguaje de recuperación para bases de datos definidas en RDF.</li> </ul> <p>En la página del W3C (<a href="http://www.w3.org/Consortium/activities">http://www.w3.org/Consortium/activities</a>) se puede encontrar abundante información y algunos ejemplos sobre los 9 grupos de trabajo definidos para estos estándares.</p> <p>Por otra parte, para apoyar el proceso de traducción, existen algunos estándares que normalizan el código existente (generalmente HTML) para que pueda ser interpretado por una máquina semántica:</p> <ul style="list-style-type: none"> <li>- <b>RDFa (RDF in attributes)</b>: Principalmente orientado a XHTML, aunque puede utilizarse en cualquier extensión XML, permite a las máquinas entender el código de manera semántica.</li> <li>- <b>eRDF (embeddable RDF)</b>: Embebido en el código, permite que un interprete extraiga el valor semántico (RDF) de un documento. Exclusivo de HTML y XHTML.</li> <li>- <b>GRDDL (Gleaning Resource Descriptions for Dialects of Languages)</b>: Recomendación del W3C para extraer RDF de XHTML utilizando XSLT.</li> </ul>
EMPRESA Y ADMIN.	<p>Dota de significado a los contenidos y establece relaciones lógicas entre ellos.</p> <ul style="list-style-type: none"> <li>- Mejora en la gestión de conocimiento, de identidades, unificación de documentos e incremento de la eficiencia de búsquedas a través de técnicas semánticas</li> </ul>



# Movilidad



## USOS Y FUNCIONALIDADES

- Acceso a la información desde cualquier lugar
- Creación de redes ad-hoc dinámicas entre dispositivos que generalmente no se encuentran interconectados (vehículos)

## EJEMPLOS

<http://www.fon.com/es/>: Comunidad para compartir WiFi. Una vez registrado, con tu *user* y *password* puedes acceder a la red de cualquier otro usuario de Fon en cualquier parte del mundo.

TÉRMINOS RELACIONADOS	<p><b>Ubicuidad</b></p> <p>La ubicuidad se refiere por definición a la capacidad de estar a la vez en varios sitios. Aplicado a la red, indica una tendencia que se está persiguiendo mucho desde hace unos meses: se considera que en poco tiempo la WWW será algo que nos envolverá y a lo que tendremos acceso de muy diversas maneras desde todos los objetos de nuestra vida cotidiana.</p> <p>Este concepto está también fuertemente relacionado con el Internet de las cosas.</p> <p><b>La Web Movil</b></p> <p>Dado el porcentaje de penetrabilidad de teléfonos móviles entre los ciudadanos (prácticamente total), los encargados de realizar estudios sobre tendencias (<a href="http://www.morganstanley.com/institutional/techresearch/pdfs/MS_Economy_Internet_Trends_102009_FINAL.pdf">http://www.morganstanley.com/institutional/techresearch/pdfs/MS_Economy_Internet_Trends_102009_FINAL.pdf</a>) prevén que en 2011 el acceso a servicios desde el móvil alcanzará unos niveles de uso similares a la actual banda ancha. De hecho ya poseemos la tecnología y un buen número de aplicaciones, y según estudios similares al nombrado, será el ratio de usuarios 3G unido a unas tarifas asequibles por parte de los operadores lo que determine cuánto falta para que consultemos en tiempo real los restaurantes que tenemos cerca, el horario de la película que queremos ver o las últimas actualizaciones de nuestros amigos en nuestra red social a través de nuestro teléfono.</p>
--------------------------	--

TECNOLOGÍA	<p>La movilidad está muy relacionada con las redes de acceso. El máximo exponente de movilidad es la tecnología móvil (<b>GSM, UMTS</b>). Los anchos de banda cada vez mayores unidos a las tarifas de datos cada día mas asequibles y las mejoras en la ergonomía de los dispositivos están potenciando la posibilidad de acceder a la Red en cualquier momento y lugar.</p> <p>Otra tecnología que facilita la movilidad es el acceso <b>Wi-Fi</b> (estandar IEEE 802.11). Las ciudades cada vez ponen más esfuerzos en ofrecer zonas de cobertura con conexión gratuita.</p> <p><b>Bluetooth</b> también ofrece conectividad móvil, aunque su alcance es más local y está orientado a eliminar los cables entre dispositivos y periféricos.</p> <p><b>WiMAX</b> es una tecnología inalámbrica de naturaleza similar a Wi-Fi, pero en un rango de frecuencias diferente. Su popularización se está retrasando debido a la falta de dispositivos comerciales importantes, aunque se espera que en los próximos años alcance relevancia.</p>
------------	--

EMPRESA Y ADMIN.	<p>Independiza la comunicación / colaboración del CUANDO y el DONDE.</p> <ul style="list-style-type: none"> <li>- Mejora de la sincronización de la Intranet para los trabajadores que se encuentran fuera de la oficina</li> <li>- Actualización de información de los proyectos a través de diversos dispositivos: móvil, PDA, portátil...</li> </ul>
---------------------	---

# Personalización



## USOS Y FUNCIONALIDADES

- El usuario 'acomoda' su entorno y las herramientas a su manera de trabajar
- Integración de feeds para minimizar las páginas a visitar
- Alertas, calendario... siempre a la vista y programados para avisar de eventos

## EJEMPLOS

<http://www.google.es/ig>: La página de inicio personalizable de Google.

<http://www.netvibes.com>: Página de inicio personalizable, con posibilidad de lectura de feeds, correo, inclusión de widgets...



**Widgets**

Los widgets son pequeñas aplicaciones que resuelven una tarea específica y que se pueden insertar en una página web. El modelo de WebParts de MOSS es una buena aproximación al concepto de personalización y al uso de widgets.

**TÉRMINOS RELACIONADOS**

<http://www.snap.com/snapshots.php>: Widget para enriquecer el contenido web. Entre otras funciones, muestra un preview en una ventana al pasar el ratón sobre un enlace.

<http://www.dapper.net/>: Permite recortar contenidos de la web y convertirlos en feeds, insertarlos en otra web... Basicamente importa el *copy-paste* en los navegadores.

<http://widgets.yahoo.com/>: Una biblioteca de widgets mantenida por Yahoo! Representativa del tipo de aplicaciones que hay disponibles.

**TECNOL.** La base de la personalización, los widgets, son generalmente código embebido en **HTML** que el usuario debe copiar y pegar. Muchos editores de contenido (blogs y wikis) poseen herramientas que permiten introducirlos para no tener que acudir al código fuente.

**EMPRESA  
Y ADMIN.**

El sistema conoce al usuario y se adapta a él. Evita el “ruido informacional y operacional” y mejora la eficiencia de la persona.

- Mejora de la eficiencia de cada empleado posibilitando customizar el entorno de trabajo

## A.6. Diagrama de oportunidad

Como conclusión al presente estudio, se ha elaborado un '*Diagrama de oportunidad*' en el que se enfrenta el *Retorno de Inversión* (ROI) previsto para cada una de las tecnologías con el tiempo.

Para alinear el estudio con los intereses de la empresa, se han sub-clasificado los elementos estudiados en tres apartados:

- *Gestión del conocimiento*: Se refiere a aquellos elementos que se aplican directamente a un foco principal de la actividad de la empresa (La gestión del conocimiento a través de plataformas colaborativas).
- *Extensión de proyectos web*: Se aplica a aquellos elementos identificados para complementar la actividad principal, sin desarrollarla completamente por sí mismos.
- *Plataformas de nueva generación*: Recoge las herramientas que suponen una manera diferente de abordar la actividad de la empresa, y que suponen un cambio más profundo.

Además, se ha utilizado un código de colores para enfatizar el interés por cada herramienta/tecnología:

- En amarillo, se indican los elementos que no alcanzan el interés suficiente para abrir una línea de investigación. Esto no significa que no resulten interesantes para la actividad de la empresa, para la cual podrán perfectamente ser considerados, sino que no se pretende dedicar recursos a desarrollarlos como herramienta propia.
- En verde, se agrupan aquellos elementos que han resultado más interesantes. Es grato advertir que muchos suponen líneas de trabajo en las que la empresa ya está trabajando. Los demás, se anotan para suponer futuras líneas de investigación.
- En rojo se señala el punto caliente, aquel que, por su posición en el gráfico y por la información recabada, ha resultado el más interesante y supondrá el objeto de estudio de la siguiente fase. Este elemento ha resultado ser el *Cloud Computing*.

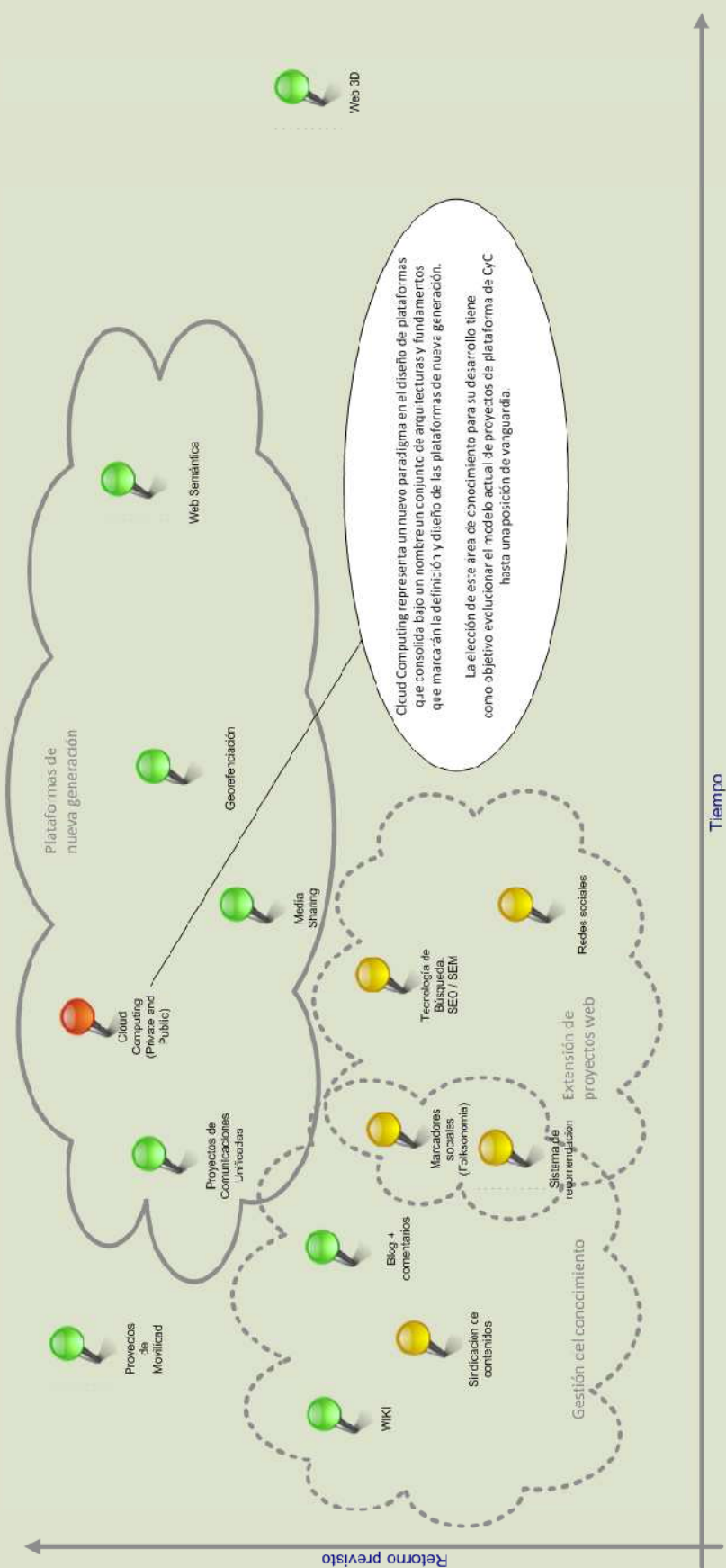
### CONTEXTO DEL DIAGRAMA

El siguiente diagrama representa una visión de oportunidad, de manera un tanto subjetiva, mediante la disposición de las distintas herramientas y fundamentos en un espacio retorno / tiempo.

Esta visión de oportunidad está condicionada y enfocada por dos factores:

- 1/ Actuales líneas de negocio del Grupo CyC, y por lo tanto activo disponible y evolucionable.
- 2/ Estrategias y apuestas colaterales de Grupo CyC.

Por lo tanto, este diagrama no puede ser tomado como una visión general de oportunidad de la Web X.O, sino como una reflexión contextualizada en el marco de una empresa concreta y unas estrategias determinadas.



## B. Aplicación de Cloud Computing a CyC

### B.1. Precedentes

La evolución en el desarrollo del *software* tomó en la pasada década una tendencia hacia diseños distribuidos en los que tanto el *hardware* como las aplicaciones dejaban de estar en local pasando a estar, parcial o totalmente, en remoto. De este modo las posibilidades no se limitan a lo que nuestra propia máquina pueda hacer, sino a los recursos que seamos capaces de reunir o localizar. El *Cloud*, como tecnología web, o de computación distribuida, mantiene elementos comunes con otras aparecidas anteriormente, entre las que caben destacar las siguientes:

#### Grid computing

Esta disciplina aprovecha la potencia conjunta de varias máquinas geográficamente distribuidas para resolver trabajos pesados de *software*. Así, mediante la asociación de máquinas más o menos potentes podemos obtener un poder de cómputo global mucho mayor y, sobre todo, mucho más económico que el que obtendríamos implementando un *supercomputador*. Los ordenadores se interconectan utilizando una red que, dependiendo del ámbito, podrá ser LAN, WAN, internet...

#### Utility computing

Se denomina así a cualquier iniciativa que tenga como fin ofrecer servicios, especialmente *software*, bajo demanda; siendo estos facturados por uso de manera similar al suministro de luz, gas, agua...

#### Virtualización

Podemos tomar una máquina física y ‘fragmentar’ sus recursos a nivel lógico, tanto *hardware* como *software*, de modo que cada uno de los fragmentos actúe desde el punto de vista del usuario como una máquina independiente con todas las funcionalidades. Así, entre otras ventajas, mejoramos el rendimiento, ya que recursos que permanecen latentes durante el tiempo de ejecución de una aplicación pueden ser utilizados para ejecutar otras.

#### Clustering

Es cuando reunimos varias máquinas físicas para utilizarlas como un solo computador lógico(n:1), o como varias máquinas virtuales distintas del número original de elementos *hardware* (n:m). Su relación con el *grid computing* es evidente.

Si en vez de considerar estas tecnologías/paradigmas por separado, tomamos sus fundamentos y los unimos, obtenemos un nuevo modo de computación: el *Cloud*

*Computing* o computación en la nube. Sintetizando, se trata de un modelo de computación distribuida donde varias máquinas con emplazamientos geográficos, compartidos o no, ponen su potencia al servicio de un(os) usuario(s) bajo demanda, y pagando sólo por el uso que se hace de esos recursos. Para distribuirlos desde el proveedor hasta dicho usuario, se implementan fragmentos de la potencia total disponible mediante virtualización.

Es el aumento del ancho de banda disponible, tanto en los hogares como en las empresas, y la madurez de los métodos de virtualización lo que ha hecho que sea ahora cuando se apuesta por este método de computación que, como acabamos de indicar, ya existía fundamentalmente por separado.

## B.2. Definición

Desde el punto de vista del usuario, podemos entenderlo como ‘*cualquier tipo de recurso de IT ofrecido como servicio*’. La nube o *Cloud*, metáfora de Internet, denominada así por la tendencia histórica de representarla como una nube, se refiere a un conjunto de recursos que no tenemos por qué comprender ni ubicar, de donde simplemente extraemos lo que necesitamos: ciclos de procesador, almacenamiento, herramientas de desarrollo, aplicaciones... Establecido el concepto, conviene definir las características que lo definen, y que nos permitirán identificarlo y diferenciarlo de otros métodos:

- *Elasticidad*: Los recursos a los que accedemos son fácilmente escalables, de modo que en todo momento tenemos garantizado el servicio aunque el volumen de demanda varíe.
- *Pago por uso*: Solamente se facturan los recursos consumidos, cuya disponibilidad viene comúnmente reflejada en un acuerdo de servicio (SLA).
- ‘*Piscina*’ de recursos: La nube es un repositorio de herramientas, ya sea en forma de *software* o *hardware*, y siempre tenemos una cantidad razonable de reservas u opciones alternativas por parte del proveedor.
- *Orientación a servicio*: Se nos ofrecen recursos encapsulados cuyo resultado es la solución de un problema concreto.
- *Abstracción de la infraestructura*: No debemos preocuparnos por las capas de la arquitectura que sostienen el servicio al cual tenemos acceso.
- *Ubicuidad*: La máquina desde la que accedamos no importa, el único requisito debe ser disponer de un navegador y estar registrado en el sistema que nos presta el servicio.

La arquitectura, vista con detalle más adelante, ofrece al usuario tres posibilidades:

*SaaS (Software as a Service)*: A través del navegador el usuario accede a aplicaciones completas de las que solo debe preocuparse por el resultado, puesto que almacenamiento, potencia de cómputo, diseño y funcionalidades son ofrecidas por el

proveedor. Éste también se encarga de almacenar el contexto, de modo que mediante autenticación el usuario puede acceder a la aplicación tal y como la dejó la última vez.

*PaaS (Platform as a Service)*: En este caso, lo que el usuario recibe es un entorno completo que le permita desarrollar aplicaciones, ya sean específicas o genéricas. De nuevo la potencia y la capacidad son cubiertas por el proveedor, y es optativo que la herramienta incorpore lo necesario para probar, hacer correr y/o depurar las aplicaciones.

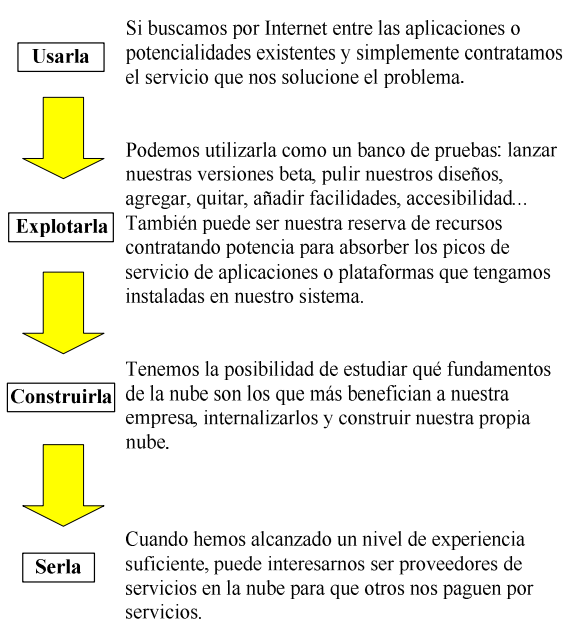
*IaaS (Infrastructure as a Service)*: El último escalón de servicios consiste en contratar expresamente la capacidad de cómputo y de almacenamiento y hacer con ella lo que necesitemos en cada momento. El proveedor nos garantiza el alojamiento y la disponibilidad a través del acuerdo de servicio, de modo que lo que evitamos es la adquisición de equipos físicos.

Las ventajas que se obtienen de utilizar una arquitectura basada en la nube son una alta eficiencia, representada por un incremento en el uso del hardware, del que se pueden obtener ratios de uso del 90% frente al 15% habitual en sistemas tradicionales, y la posibilidad de incrementar inmediatamente la capacidad de nuestros sistemas debido a la alta escalabilidad. Es decir, mejoraremos la eficiencia de nuestros recursos IT incurriendo en gastos de *OPEX* a la vez que reducimos la inversión en *CAPEX*, y lograremos agilidad en el redimensionamiento de nuestros sistemas sin invertir en nueva infraestructura, formar/dedicar más personal ni obtener más licencias de *software*.

### B.3. Tipos de nube según el ámbito

Se presentan en la *Figura B.1* cuatro niveles primarios de abstracción dependiendo de cuánto nos involucremos en el desarrollo de la nube.

#### ¿QUÉ PODEMOS HACER CON LA NUBE?



*Figura B.1. Usos de la nube*



Estos niveles permiten definir los distintos modelos de *Cloud Computing* que podemos llegar a realizar. Se distinguen tres tipos según su propiedad: públicas, privadas e híbridas.

Hablaríamos de *nube pública* siempre que la gestión de los recursos y la responsabilidad física del sistema quede en manos de un tercero, el proveedor, que se nos ofrece a través de Internet; y *privada* sería cuando todos los equipos y la arquitectura corren a cuenta de nuestra empresa, sin intercesión de un tercero en ningún momento. Cualquier solución a mitad camino entre estos dos extremos, en todas sus variantes sería una *nube híbrida*.

Algunas fuentes distinguen también *nubes gestionadas*, o nubes públicas con un nivel más de control, y que serían un tipo de nubes híbridas; otras entran en distinción nombrando las *nubes comunitarias* cuando la gestionan varios propietarios, de manera que se obtiene cierto grado de privacidad y los costes se reparten.

Adoptaremos una clasificación genérica en la que distingamos cuatro tipos de nube según la naturaleza explicada arriba: *públicas, privadas, híbridas y comunitarias*.

#### **B.4. Cloud pública vs. Cloud privada**

¿Puede privatizarse el modelo presentado sin que pierda las características que lo definen? Hay opiniones para todos los gustos. Si admitimos que sí, entonces merece la pena matizar qué ventajas se mantienen y cuáles de ellas pasan a un segundo plano o, simplemente, desaparecen.

##### La nube privada aventaja a la pública en:

- mayor privacidad de los datos (la misma a la que nos enfrentábamos al tener un sistema local)
- más control sobre el acceso a los equipos
- más velocidad por no tener que compartir recursos con terceros
- más ancho de banda (todo el que te permita tu LAN)
- no dependes de la disponibilidad de Internet (a no ser que tu nube esté distribuida geográficamente, lo cual ocurre la mayoría de las veces)

##### Pueden llegar a resultar iguales en:

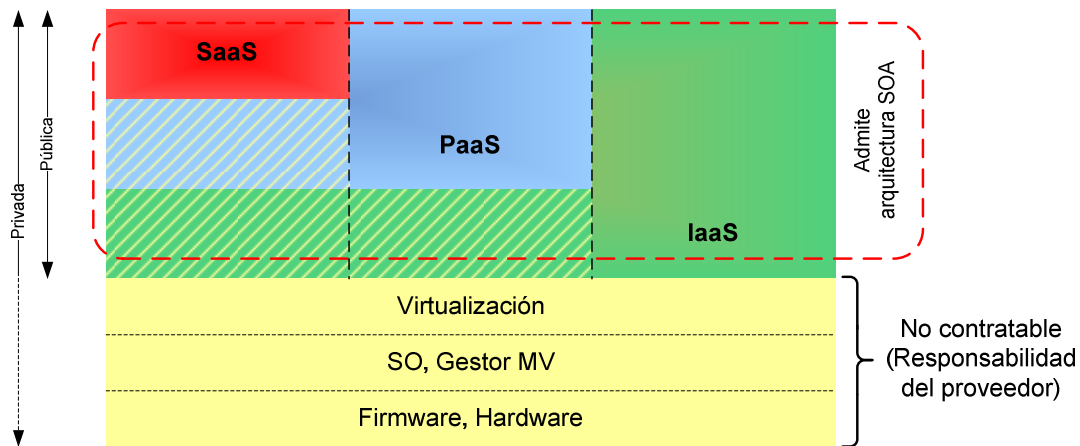
- elasticidad (escalabilidad), aunque dependes de la inversión inicial que realices en equipos
- orientación a servicios

##### Las desventajas al privatizar la nube son:

- menos ahorro al no contratar únicamente por uso (debes invertir en infraestructura)
- no te liberas de las tareas de mantenimiento y gestión, pues toda la responsabilidad sobre los equipos es tuya, no del proveedor

### B.5. Estructura bajo el Cloud

Si nos decidimos a crear una nube privada vamos a convertirnos en proveedores para nuestra propia empresa, por lo que a las capas de servicio establecidas hasta ahora (*Figura B.2*), las cuales seguiremos observando cómo usuarios, debemos añadirle la preocupación de construir aquellas que están bajo la nube (en amarillo) y que son el verdadero motor del paradigma de *Cloud Computing* y precisamente, la complejidad de la que se nos abstrae al hacer uso de la nube pública.



*Figura B.2. Capas de arquitectura Cloud*

Nótese que no se trata de una única pila sino de tres: SaaS, PaaS e IaaS, cuya base común es la relativa al proveedor. Más adelante se presentan por separado. Dependiendo de qué servicio implementemos añadimos al proveedor la responsabilidad de hacer transparente la capa de infraestructura y plataforma (en el caso de SaaS), o únicamente la infraestructura (para PaaS). Aparece en discontinuo la posibilidad de integrar SOA, esto es explicado con detalle a continuación.

## B.6. Cloud y SOA

Cloud Computing no dice **CÓMO** implementar el software sino que indica **DONDE**: en la nube. Por tanto es perfectamente compatible, incluso deseable, un desarrollo orientado a servicios. Podemos ‘posar’ una implementación SOA sobre nuestra nube (Ver Figura B.3).

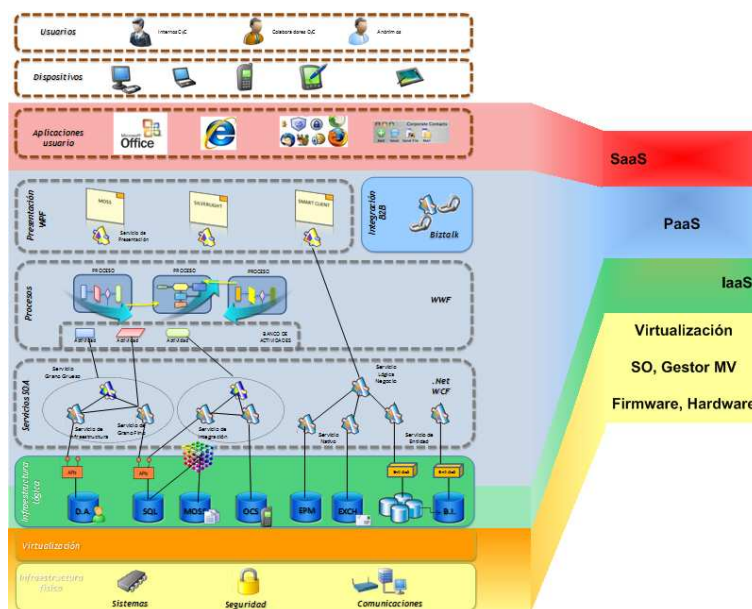


Figura B.3. Fusión SOA-Cloud

Es importante recordar que desde el punto de vista del usuario solo podemos implementar un *\*aaS* cada vez (Figura B.4), y que las capas inferiores al servicio quedarán integradas en la arquitectura del proveedor no pudiendo ser ofrecidas como servicio: para SaaS, la plataforma y la infraestructura son fijas, no ‘aaS’ y así sucesivamente.

Es **CLAVE** que seamos capaces de implementar una infraestructura virtual eficiente

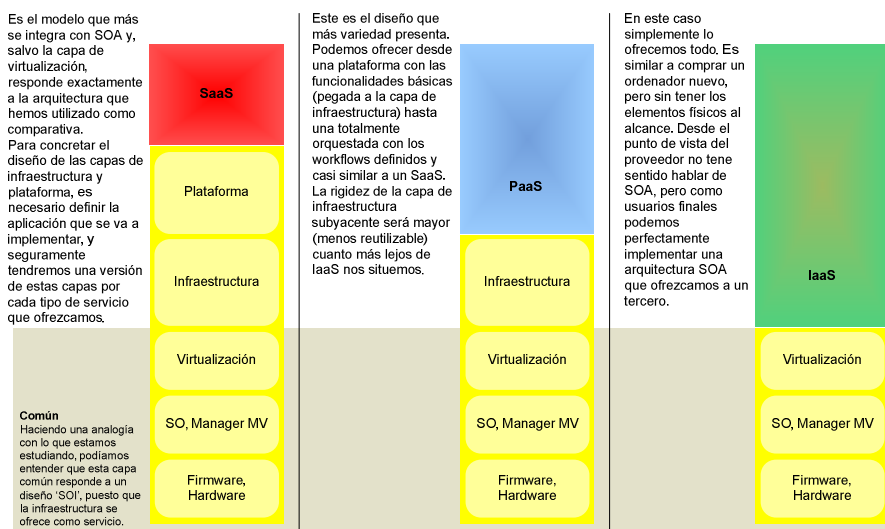


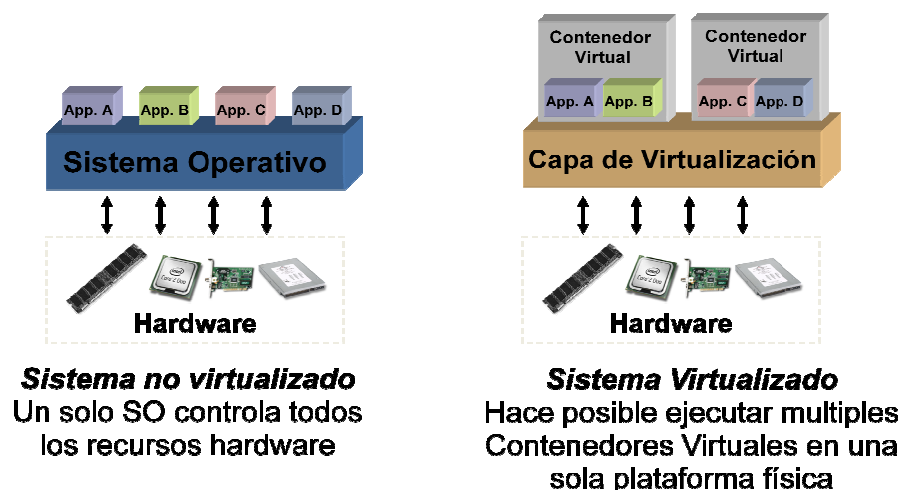
Figura B.4. Las tres posibles pilas Cloud

## C. Virtualización e hipervisor

### C.1. Introducción

La virtualización es un amplio término que se refiere a la abstracción de recursos de computación. En este documento nos centraremos en la virtualización de la plataforma, que separa el sistema operativo de los recursos de la plataforma *hardware*: virtualización total, paravirtualización y virtualización asistida por hardware.

En el pasado encontrábamos un sistema operativo por máquina, con lo que éste tenía control completo sobre los recursos de la misma (*Figura C.1a*). En la era de la virtualización, una sola máquina soporta múltiples instancias de sistema operativo: homogéneas y heterogéneas, lo que significa que una máquina física puede hospedar varias copias de *Windows/Linux* (*Figura C.1b*).



*Figura C.1a*

*Figura C.1b*

La virtualización es una capa software entre la máquina y el sistema operativo. Esencialmente, lo que esta capa hace es dividir los recursos de la máquina entre todos los sistemas operativos huéspedes. La capa de virtualización se encarga de multiplexar los recursos *hardware* entre varios sistemas operativos (SO). Cada SO percibe que controla el *hardware* completo pero, realmente, la máquina puede ahora hospedar varios sistemas operativos porque la capa de virtualización realiza la conmutación en el trasfondo.

### C.2. Fundamentos de la virtualización

#### C.2.1. Estados de máquina

Gerald J. Popek y Robert P. Goldberg, en su artículo "*Formal Requirements for Virtualizable Third Generation Architectures*", establecieron los requerimientos para que una arquitectura fuera virtualizable. A pesar de datar de 1974, y de que por aquel

entonces *IBM* ya trabajara en la virtualización de los grandes *mainframes*, las bases establecidas en dicho artículo siguen siendo vigentes hoy en día. Existen tres propiedades de interés cuando analizamos el entorno creado por un VMM (Gestor de máquina virtual o *Virtual Machine Manager*):

- *Equivalencia*: Un programa ejecutándose bajo el VMM debe mostrar un comportamiento esencialmente idéntico al que mostraría cuando se ejecuta directamente en una máquina física equivalente.
- *Control de recursos*: El VMM debe tener control total sobre los recursos virtualizados.
- *Eficiencia*: Un elevado porcentaje de las instrucciones de la máquina deben ser ejecutadas sin la intervención del VMM.

En la terminología de Popek & Goldberg, un VMM debe presentar las tres propiedades. Actualmente, se acepta que satisfaga las de equivalencia y control de recursos.

### C.2.2. Requerimientos para la virtualización

Para obtener sus requerimientos de virtualización, Popek & Goldberg clasifican las instrucciones de una arquitectura de tercera generación, la existente cuando escribieron el artículo, en 3 grupos:

- *Con privilegios (Privileged instructions)*: Aquellas que son atrapadas (*trap*) si el procesador se encuentra en modo usuario (*User mode*) y no lo son si se encuentra en modo sistema (*System mode*).
- *Sensibles al control (Control sensitive instructions)*: Aquellas que intentan cambiar la configuración de los recursos del sistema. Por ejemplo, cambiar la cantidad de memoria en uso.
- *Sensibles al comportamiento (Behavior sensitive instructions)*: Aquellas cuyo comportamiento o resultado depende de la configuración de los recursos, como el contenido del registro de ubicación (*relocation register*) o del modo del procesador.

#### Teorema 1:

*Para una máquina de tercera generación convencional, se podrá construir un VMM si el conjunto de instrucciones sensible (tipos 2 y 3) son un subconjunto del conjunto de instrucciones con privilegios.*

Esto garantiza la propiedad de control de recursos. Las instrucciones no privilegiadas deben, en cambio, ser ejecutadas en modo nativo, esto es, cumplir la propiedad de eficiencia. También se debe considerar la búsqueda de la propiedad de equivalencia.

### Teorema 2.

*Una máquina de tercera generación convencional será recursivamente virtualizable si es virtualizable y un VMM sin dependencias temporales puede ser construido sobre ella.*

#### **C.2.3. Definiciones**

*Máquina Virtual (MV):* Una máquina virtual pretende ser un duplicado eficiente y aislado de una máquina real.

*Trap:* Cuando una instrucción es atrapada (*trap*), el almacenamiento permanece sin cambios, excepto la posición cero donde se coloca la palabra de estado (PSW) que estaba en uso justo antes de que la instrucción fuera atrapada. Un '*trap*' salva automáticamente el estado actual de la máquina y entrega el control a una rutina predefinida que cambia el modo del procesador, el registro de ubicación y el contador de programa (PC) a los valores especificados.

*Virtual Machine Monitor:* El VMM será una pieza particular de software, que podríamos llamar programa de control, que muestra unas propiedades concretas. Los módulos del programa de control se engloban en tres grupos que se presentan aquí de manera vaga:

- El primer grupo es el *dispatcher (D)* o *despachador*. Su instrucción inicial está ubicada en la posición hacia la que el *hardware* 'atrapa' las instrucciones: el valor de P en la posición 1. El *dispatcher* puede ser considerado como el módulo de control más alto del programa de control. Decide a qué módulo se ha de llamar.
- El segundo grupo de esta especificación tiene un solo miembro, *el allocator (A) o repartidor*. Su tarea es decidir qué recursos del sistema deben de ser aprovisionados. En el caso de una única MV, el repartidor solo necesita mantener la MV y el VMM separados. Cuando son varias, su tarea es evitar conceder los mismos recursos a más de una MV a la vez.
- El tercer conjunto de módulos en el programa de control puede ser visto como interpretes para todas las demás instrucciones que se atrapan, un intérprete por cada instrucción con privilegios. El propósito de cada intérprete es simular el efecto de la instrucción que es atrapada.

### **C.3. Historia de la virtualización**

La virtualización es un concepto maduro que fue inicialmente desarrollado en los 60 para particionar los enormes *mainframes* de entonces. Hoy, los ordenadores basados en arquitectura x86 presentan los mismos problemas de rigidez e infrautilización que los *mainframes* mostraban entonces. Para solucionar esto, *VMware* inventó la virtualización de las plataformas x86 en los 90, superando varios desafíos en el proceso que llevó a cabo.



### C.3.1. Virtualización de mainframes

La virtualización fue implementada por primera vez hace más de 40 años por *IBM* como un método para particionar de manera lógica los *mainframes* en máquinas virtuales independientes. Estas particiones hicieron posible la ‘*multitarea*’: ejecutar múltiples aplicaciones y procesos al mismo tiempo. Dado que los *mainframes* eran recursos caros en ese momento, el particionado fue presentado como una manera de amortizar la inversión.

La primera máquina que soportó la virtualización total fue la ‘*IBM’s VM*’, que comenzó su vida como parte del proyecto *System/360*. La idea del *System/360* (S/360) era proveer una arquitectura estable y un paquete de actualización para los consumidores de *IBM*. Una gran variedad de máquinas fueron producidas con la misma arquitectura básica, con lo que las pequeñas empresas podían comprar un *miniordenador* si era todo lo que necesitaban y actualizarse a un *mainframe* con el mismo *software* tiempo después.

Una de las claves de mercado que *IBM* identificó entonces fue que la gente deseaba consolidar sus máquinas S/360. Una compañía con unos pocos *miniordenadores* S/360 podía ahorrar dinero actualizándolo a un único *mainframe* S/360, asumiéndose que el *mainframe* pudiera proporcionarles las mismas características.

El *Model 67* introdujo el concepto de conjunto de instrucciones auto-virtualizadoras. Esto significaba que podía ser fácilmente particionado y presentarse como un número de versiones menos potentes de sí mismo. Podía incluso ser recursivamente virtualizado, particionando a su vez cada máquina virtual.

### C.3.2. La necesidad de la virtualización x86

La virtualización fue abandonada durante los 80 y los 90, cuando las aplicaciones cliente-servidor y los baratos servidores (y PCs) x86 establecieron el modelo de computación distribuida. En lugar de compartir recursos de manera centralizada, como el modelo *mainframe*, las organizaciones comenzaron a utilizar sistemas distribuidos de bajo coste para construir islas de capacidad de cómputo. La adopción masiva de *Windows* y la emergencia de *Linux* como sistema operativo de servidor en los 90, posicionaron a los servidores x86 como el estándar de la industria.

## C.4. Arquitectura x86

El término general x86 se refiere al conjunto de instrucciones de arquitectura *CISC* del CPU con más éxito comercial de la historia, usado en procesadores *Intel*, *AMD*, *VIA* y otros. Deriva del número de modelo de las primeras generaciones de CPU, retro-compatibles con el *chip* original de 16 bits de *Intel*, el 8086 de 1978, la mayoría de los cuales acababan en “86”.

Dado que el término x86 se volvió popular tras la introducción del 80386 en 1985, éste frecuentemente abarca también la compatibilidad binaria con el conjunto extendido de instrucciones de este *chip*, es decir, la versión de 32 bits. A veces se

enfatisa como x86-32 para distinguirlo del original de 16 bits (x86-16) o del nuevo de 64 bits (x86-64, también escrito x64).

Hoy en día, referirse a *hardware* x86 suele implicar también las capacidades de 64 bit, al menos para PCs y servidores. Sin embargo, para evitar problemas de compatibilidad por error, la etiqueta *software* x86 implica únicamente 32 bit, mientras que x86-64 o x64 se utiliza sólo para *software* exclusivo de 64 bit.

Los únicos competidores significativos de x86 en PCs eran los conjuntos de instrucciones *Motorola 68k* (tipo *CISC*) y *PowerPC* (tipo *RISC*). Sin embargo, el 7 de agosto de 2006, *Apple Inc.* migró a un CPU x86 garantizando el monopolio de este conjunto de instrucciones en el campo de los PCs y *notebooks*. También ha obtenido un crecimiento significativo en servidores y estaciones de trabajo. Los mercados sin presencia significativa de x86 se reducen a procesadores embebidos de bajo coste localizados en aplicativos y juguetes, y poco más.

Una cantidad incontable de *software* es desarrollado para la plataforma x86, incluidos la mayoría de los sistemas operativos modernos, desde *MS-DOS* y *Microsoft Windows* hasta *Linux*, *BSD*, *Solaris OS* y *Mac OS X*, haciendo el conjunto de instrucciones de la arquitectura x86 indispensable a escala global y prácticamente irremplazable.

Es más barato, más seguro y más fácil para el consumidor comprar nuevo *hardware* para cubrir necesidades nuevas que tratar de instalar las aplicaciones en el existente. Pero el *datacenter* creado de esta forma presenta los siguientes *handicaps*:

- 1 máquina → 1 sistema operativo → varias aplicaciones.
- Cada aplicación puede afectar a las demás.
- La gran desventaja: la utilización de la máquina es muy baja, la mayoría del tiempo menor del 25%, ente el 10-18% generalmente (*Figura C.2*).
- Se incrementan los costes de infraestructura física: instalaciones, potencia, refrigeración, etc.
- Se incrementan los costes de gestión de IT: configuración, despliegue, actualizaciones, etc.
- *Failover* y protección ante desastres insuficiente.

La solución para todos estos problemas es virtualizar las plataformas x86. Así se aúnan los beneficios de un alto índice de utilización de *hardware* con la ejecución de varios sistemas operativos, y por tanto aplicaciones, en entornos virtualizados separados. Cada aplicación se ejecuta en su propio sistema operativo y cada sistema operativo desconoce que está compartiendo el hardware subyacente con otros.

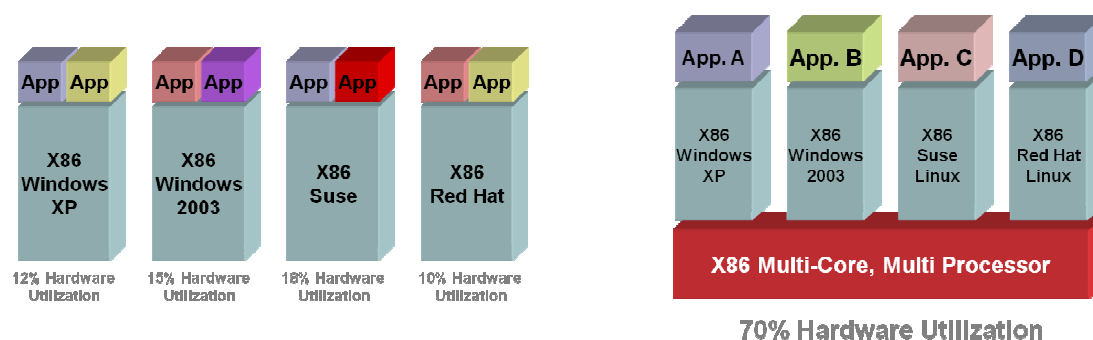


Figura C.2. Rendimiento del hardware sin y con virtualización

#### C.4.1. IA-32 (x86)

El conjunto de instrucciones IA-32 contiene 18 instrucciones sensibles o sin privilegios. Pueden dividirse en dos grupos:

- *Sensibles a registro:* Leen o escriben registros o posiciones de memoria como el registro de reloj o el de interrupción: PUSHF, POPF, SGDT, SIDT, SLDT, SMSW.
- *Protección del sistema:* Se refieren al sistema de protección del almacenamiento, memoria o redireccionamiento: CALL, JMP, INT n, LAR, LSL, MOV, POP, PUSH, RET, STR, VERR, VERW.

No obstante, la arquitectura x86 es realmente una buena candidata a ser virtualizada.

#### C.4.2. Niveles de privilegio

El mecanismo de protección segmentada del procesador x86 reconoce 4 niveles de privilegio, numerados del 0 al 3. A mayor número, menos privilegios. La Figura C.3 muestra cómo estos niveles pueden ser interpretados como un anillo de protección. El centro, reservado para el código, datos o pila más privilegiados, es usado por los segmentos que contienen el *software* crítico, normalmente el *kernel* del sistema operativo. Los anillos exteriores son utilizados por *software* menos crítico. Los sistemas que utilizan anillos basados en 2 niveles suelen utilizar los niveles 0 y 3.

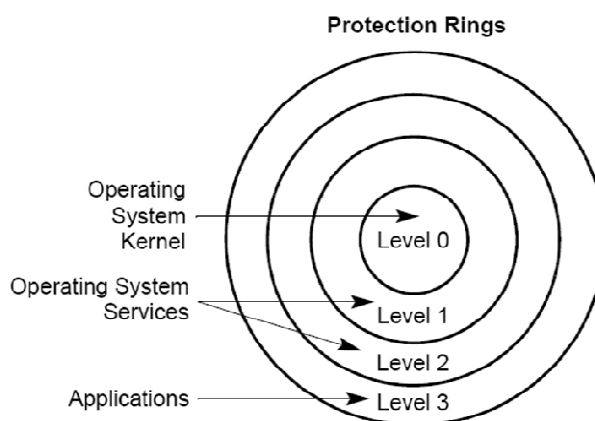


Figura C.3. Anillo de privilegios del procesador x86

El procesador utiliza los niveles de privilegio para prevenir que un programa o tarea que opera en un nivel de privilegios inferior alcancen un segmento con mayores privilegios, excepto situaciones específicas. Cuando el procesador detecta una violación de los niveles de privilegio, lanza una excepción de protección general (#GP).

Para llevar a cabo revisiones del control de privilegios entre el segmento de código y el segmento de datos, el procesador reconoce los siguientes tres tipos de niveles de privilegios:

- *Nivel de privilegio actual o Current privilege level (CPL)*: El CPL es el nivel de privilegio del programa o tarea en ejecución. Se guarda en los *bits* 0 y 1 de los registros CS y SS. Normalmente, el CPL es igual que el nivel de privilegios del segmento de código del cual es tomada la instrucción. El procesador cambia el CPL cuando el control del programa es transferido a un segmento de código con diferente nivel de privilegios.
- *Nivel de privilegio de descriptor o Descriptor privilege level (DPL)*: El DPL es el nivel de privilegios de un segmento o puerta. Se almacena en el campo DPL de descriptor de segmento. Cuando el código en ejecución intenta acceder a un segmento, el DPL se compara con su CPL y el RPL del selector de segmento. El DPL se interpreta de manera diferente, dependiendo del tipo de segmento al que se accede.
- *Nivel de privilegio solicitado o Requested privilege level (RPL)*: El RPL es un nivel de privilegio para ‘sobrescribir’ que se asigna al selector de segmento. Se almacena en los *bits* 0 y 1 del selector de segmento. El procesador compara el RPL junto al CPL para determinar si se permite el acceso a un segmento. Incluso si el programa o la tarea que solicita el acceso tienen suficientes privilegios para acceder al segmento, se deniega el acceso si el RPL no tiene nivel de acceso suficiente. Es decir, si el RPL del selector de segmento es numéricamente mayor (menos privilegios) que el CPL, el RPL lo sobrescribe, y viceversa. El RPL puede utilizarse para asegurarse de que código privilegiado no accede a un segmento en nombre de una aplicación a menos que dicha aplicación posea los mismos privilegios.

Los niveles de privilegio se comprueban cuando el selector de segmento de un descriptor es cargado al registro de segmento. La comprobación para el acceso de datos es diferente a la realizada para el segmento de código.

#### Ejemplo: Lectura de los descriptores de segmento

- Registros *Code Segment* (CS) y *Stack Segment* (SS):
  - Los 14 *bits* superiores de estos registros contienen el puntero de segmento y el selector de la tabla de descriptores.
  - Los dos *bits* más bajos del CS y el SS contienen el CPL.

- Instrucciones que acceden explícita o implícitamente al selector CS/SS, incluidas CALL, MOV al SS y POP al SS, no son *atrapadas* cuando se ejecutan en modo usuario.
  - Al ejecutar POP SS el SO huésped detectará que no está ejecutándose en modo privilegiado (nivel 0) sino en el nivel 1:
    - La propiedad de equivalencia puede llegar a ser violada
    - Se viola la propiedad de control de recursos

#### C.4.3. Virtualización de servidor

##### Virtual Machine Monitor (VMM)

Cada máquina virtual interacciona con su sistema anfitrión a través del gestor de máquina virtual (VMM). Al ser el principal enlace entre la MV y el sistema operativo anfitrión, el VMM proporciona una función crucial.

- Cuando se emplea virtualización total, el VMM presenta un conjunto completo de *hardware* emulado al sistema operativo huésped. Esto incluye CPU, placa base, memoria, disco, controladores de disco y tarjetas de red. Por ejemplo, *Microsoft Virtual Server 2005* emula una tarjeta de red *Intel 21140* y el *chipset Intel 440BX*.
- La siguiente tarea significativa del VMM es proporcionar aislamiento. El VMM tiene control total de los recursos de la máquina física, dejando a las MVs acceder únicamente a sus recursos virtuales emulados. El VMM no contiene mecanismos para la comunicación entre MVs, lo que obliga a dos máquinas virtuales que pretendan comunicarse a hacerlo a través de una red.
- Otra importante tarea del VMM es gestionar el acceso a los recursos del sistema anfitrión. Esto es importante, porque prevé que la infratilización de una MV repercuta en el rendimiento de otras MVs en el mismo anfitrión. A través de la consola de configuración del sistema, los recursos *hardware* como CPU, red y acceso a disco pueden ser repartidos, con el máximo porcentaje posible asignado a cada MV. Esto permite al VMM un acceso distribuido apropiado a la vez que garantiza que las MVs críticas tendrán acceso a la cantidad de recursos que necesitan para completar sus operaciones.

##### SO anfitrión/Hipervisor

La función principal del sistema operativo anfitrión o *hipervisor* es trabajar con el VMM coordinando el acceso a los recursos de la máquina física. Esto incluye organizar el acceso al CPU así como a los *drivers* para comunicarlos con las unidades físicas del anfitrión, así como con las tarjetas de red.

El término *hipervisor* se utiliza para describir un *shell* ligero cuyo único propósito es proveer servicios de *hosting* a las MVs. El *hipervisor* difiere de un SO

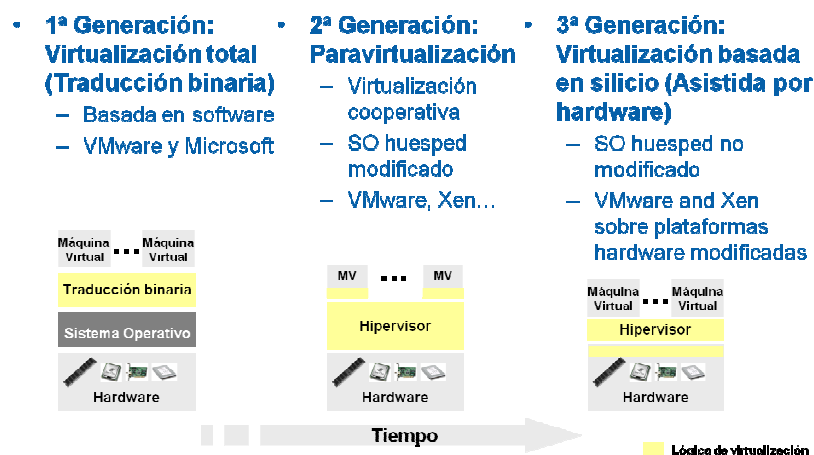
tradicional en que éste último está diseñado para cumplir otras funciones en la red. Al estar hecho a medida para realizar *hosting* de MVs, el *hipervisor* generalmente ofrece mayor rendimiento y presentará menos vulnerabilidades de seguridad porque ejecuta menos servicios y contiene únicamente código esencial. Los *hipervisores* escritos para virtualización asistida por *hardware* pueden penetrar mucho más profundamente en la arquitectura y ofrecer como resultado mayores niveles de rendimiento.

Como cualquier SO tradicional, un *hipervisor* contiene su propio código de sistema operativo; por tanto, mantener actualizaciones de seguridad sigue siendo importante. A diferencia de los tradicionales, los *hipervisores* son específicos de cada fabricante, por lo que las actualizaciones deben provenir directamente de éste. Por ello, el soporte para los dispositivos proviene de los fabricantes de virtualización, lo que hace que las empresas deban asegurarse de que los productos de virtualización son compatibles con el *hardware* disponible. Cuando se alojan MVs en sistemas operativos tradicionales como *SUSE Linux Enterprise Server* o *Windows Server* se puede observar que a pesar de que el SO anfitrión es más pesado, en tamaño, que un *hipervisor*, aquél provee flexibilidad adicional con los dispositivos *hardware*. En la integración de una SAN, por ejemplo, si el SO anfitrión no reconoce el HBA del canal de fibra (FC), el administrador puede descargar los *drivers* apropiados del sitio web del fabricante. Con el *hipervisor*, el administrador necesitará pedir el *driver* al proveedor de virtualización, o admitir que el dispositivo no tiene soporte.

Tanto los *hipervisores* como los sistemas operativos tienen sus ventajas y sus desventajas. Los sistemas operativos proveen mayor soporte de dispositivos que los *hipervisores*, pero además requieren atención para asegurarse que contienen todos los parches y actualizaciones de seguridad. Los *hipervisores* ocupan un espacio mínimo de disco y almacenamiento, pero sus actualizaciones y parches deben provenir directamente del fabricante de virtualización.

## C.5. Evolución de la virtualización

En la *Figura C.4* se presentan las tres generaciones de virtualización de las plataformas x86. A continuación se presentan sus características principales.

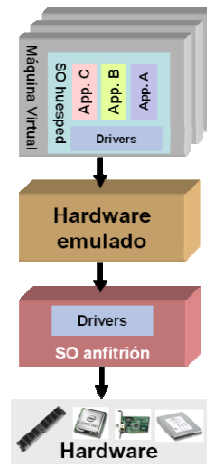


*Figura C.4. Evolución de la virtualización*



### C.5.1. Virtualización total

Fue la primera generación en ofrecer virtualización de servidor x86/x64. En ella, todo el *hardware* es emulado, incluso el CPU. Como ejemplo de este tipo de virtualización, dos emuladores populares y open-source son *QEMU* y *Bochs*. La arquitectura genérica puede verse en la *Figura C.5*.



*Figura C.5. Arquitectura de la virtualización total*

Hay diferentes aproximaciones a este tipo de virtualización: la emulación y la traducción binaria.

- *Emulación hardware*: La emulación *hardware* se presenta con un precio sobre el rendimiento cuando el VMM traduce las instrucciones entre el *hardware* emulado y los actuales *drivers* del sistema:
  - Usualmente, menos de un 2% para dispositivos emulados, como la RAM.
  - Del 8% al 20% para dispositivos de entrada/salida como tarjetas de red y discos duros, por lo que la emulación supone un alto precio.
- *Traducción binaria*: La capa de emulación (el VMM) se comunica con el sistema operativo que pretende comunicarse con el *hardware* del ordenador.
  - El VMM modifica la imagen binaria huésped en tiempo de ejecución para tomar el control del procesador cuando el *software* huésped trata de llevar a cabo una operación con privilegios.
  - El VMM puede emular la operación con privilegios y devolver el control al *software* huésped.
  - El huésped no se da cuenta de que se está ejecutando en un entorno emulado.

#### Ventajas

- La capa de emulación aísla las MVs del SO anfitrión y de otras MVs y controla el acceso de cada MV a los recursos del sistema, previniendo que una MV inestable interfiera en el rendimiento del sistema.

- Emulando un conjunto consistente de hardware, las MVs adquieren la capacidad de moverse de manera transparente entre anfitriones con hardware diferente sin problemas.
  - Es posible ejecutar un sistema operativo que fue desarrollado para otra arquitectura en tu propia arquitectura
  - Una MV ejecutándose en un servidor Dell puede ser realojada en un servidor HP

Los sistemas operativos diseñados para entornos x86/x64 están escritos para tener acceso total al nivel 0, donde se ejecutan las instrucciones privilegiadas del SO. Éstas incluyen el *kernel* del SO y el acceso de los *drivers* al *hardware* del sistema. Las aplicaciones se ejecutan en el nivel 3. (Figura C.6).

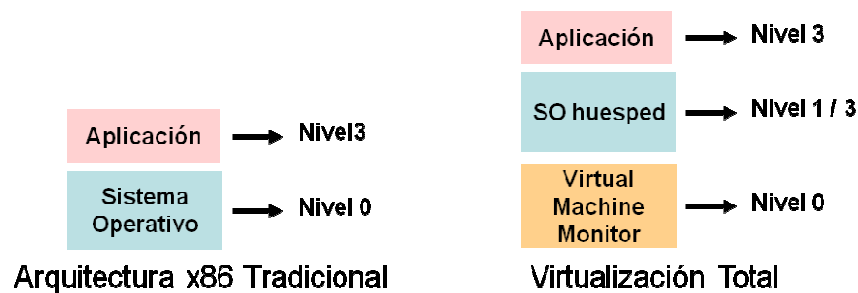
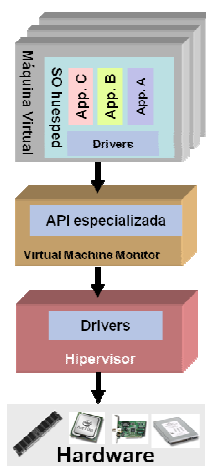


Figura C.6. Niveles de ejecución tradicional frente a virtualizados

En un entorno virtualizado, el VMM se ejecuta en el nivel 0 junto al SO anfitrión y sus *drivers*. Las MVs no pueden acceder al nivel 0 sin introducir conflictos, por lo que el VMM ejecuta todas las MVs en el nivel 1. Dado que las instrucciones con privilegios del huésped esperan ejecutarse en el nivel 0, el VMM debe proporcionar una traducción con el fin de ‘engañar’ al *kernel* del SO huésped haciéndole creer que tiene acceso al nivel 0. Si el SO huésped no solicita acceso al nivel 0, entonces la traducción no es necesaria y el rendimiento se incrementa sustancialmente. Aquí es donde entra en juego la paravirtualización.

### C.5.2. Paravirtualización

La paravirtualización fue desarrollada como una manera de solventar los requerimientos de la emulación para las instrucciones con privilegios de las MVs. Con la paravirtualización, un API de virtualización y los *drivers* son cargados en el sistema operativo huésped (*Figura C.7*). Esto le permite ejecutarse siendo consciente de la arquitectura de virtualización y, por lo tanto, ejecutando las operaciones de *kernel* en el nivel 1. El resultado final es que las instrucciones con privilegios no deben ser traducidas. Las diferencias de arquitectura entre paravirtualización y virtualización total se dan entre la MV y el VMM.



*Figura C.7. Arquitectura de la paravirtualización*

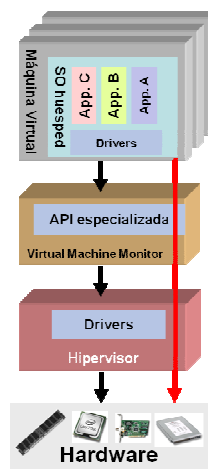
La paravirtualización requiere la existencia de *drivers* de virtualización de dispositivos en la MV huésped, el SO huésped, el VMM y el *hipervisor*. Al incluir las APIs en el *kernel* del SO huésped, éste aprende cómo tratar las instrucciones con privilegios; por lo que no es necesaria la traducción por parte del VMM. Además, los *drivers* de virtualización para dispositivos de red o almacenamiento se programan con intención de comunicarse con los *drivers* del VMM y el *hipervisor*. Por ello, el VMM no tiene que presentar un *driver* alternativo al SO huésped para, a continuación, traducir sus instrucciones al *hipervisor*. Eliminando los grandes requerimientos del VMM para la emulación reduces su carga de trabajo, pues ya simplemente debe aislar y coordinar el acceso de cada MV a los recursos físicos de *hardware*.

El otro beneficio de la paravirtualización es el acceso al *hardware*. Con los *drivers* apropiados en su *kernel*, el SO huésped es ahora capaz de comunicarse directamente con el *hardware* del sistema. Nótese que esto no significa que la MV tenga acceso directo a todo el *hardware* del sistema. En la mayoría de los casos, parte del *hardware* estará disponible, mientras que otros dispositivos aparecerán con representaciones genéricas, según determinen los *drivers* de virtualización presentes en la MV. Para determinar qué elementos del *hardware* están paravirtualizados y cuales son accesibles directamente, se debe consultar con cada proveedor de *software* de virtualización. Actualmente, existen dos métodos para alcanzar la paravirtualización:

- *Recompilar el kernel del SO*: Los *drivers* de virtualización deben residir en el *kernel* del SO huésped. Se necesita un SO modificado que incluya la API específica, lo que implica recompilar el sistema operativo. Algunos proveedores, como *Novell*, han adoptado la paravirtualización y ofrecen distribuciones modificadas de sus SOs, mientras que otros, como *Microsoft*, no disponen de versiones modificadas.
- *Instalar drivers de virtualización*: En algunos sistemas operativos no es posible utilizar la paravirtualización completa, porque ésta requiere versiones especializadas del SO. Para asegurar una buena adaptación en diversos entornos, la paravirtualización puede aplicarse a los dispositivos por separado. Por ejemplo, las instrucciones generadas por las placas de red o las tarjetas de interfaz gráfica pueden ser modificadas antes de abandonar la máquina virtualizada utilizando *drivers* de virtualización.

### C.5.3. Virtualización asistida por hardware

En este tipo de virtualización, el SO huésped se ejecuta en nivel 0. El VMM utiliza extensiones de procesador (como *Intel-VT* o *AMD-V*) para interceptar y emular las operaciones con privilegios en el huésped. La virtualización asistida por *hardware* elimina muchos de los problemas que dificultan la creación del VMM haciendo que éste se ejecute en un nivel más privilegiado que el 0, el nivel -1 creado virtualmente. La *Figura C.8* muestra su arquitectura.



*Figura C.8* Arquitectura de la virtualización por hardware

La virtualización asistida por *hardware* es una firme candidata a alzarse como estándar de virtualización de servidores en el futuro. Mientras que la primera generación de *hardware* que ofrece virtualización asistida presenta mayor rendimiento de CPU y aislamiento de máquina virtual avanzado, las futuras implementaciones prometen extender ambas características a otros elementos del *hardware*, como la memoria. Esto se manifestará en forma de espacio de direccionamiento dedicado asignable a cada MV. La tecnología '*AMD-V's forthcoming nested paging support*' eliminará el cuello de botella de paginación presente en el actual método de paginación, *Shadow Paging*, e

incrementará el rendimiento de la memoria. *Intel* ofrecerá la misma funcionalidad, llamada *Extended Page Tables* (EPT), en futuras implementaciones de sus *chips* VT.

Los CPUs que soportan virtualización asistida por *hardware* son totalmente conscientes de la presencia de la pila de virtualización. Con la opción activada a través de la BIOS, el sistema automáticamente reserva espacio físico de direccionamiento exclusivamente para las máquinas virtuales. Esto proporciona verdadero aislamiento para los recursos de las MVs. Además, señalar la existencia de un bus de dispositivos E/S ‘*pass-through*’ en la pila de virtualización. Esto es significativo porque las máquinas virtuales pueden utilizar dicho bus para acceder a los dispositivos de entrada/salida, tales como disco o red, de manera directa, en lugar de a través de recursos de *hardware* emulados. Téngase en mente que mientras el bus ‘*pass-through*’ puede proporcionar un camino de datos limpio para los recursos físicos de *hardware*, toda la información de control es procesada por el VMM, que evita que una MV tome control total de un recurso de *hardware*.

#### Pros y contras

- Pros: Permite ejecutar SOs sin modificar.
- Contras: Velocidad y flexibilidad. Un SO sin modificar no sabe que está ejecutándose en un entorno virtualizado y, por tanto, no puede aprovechar las ventajas de la virtualización. Esto puede ser resuelto con virtualización parcial.

El sistema operativo huésped es provisto con acceso al nivel 0. Permitiendo al VMM ejecutarse sobre este nivel, los sistemas operativos huéspedes pueden procesar instrucciones con privilegios sin necesidad de ninguna traducción por parte del VMM. Esto elimina el requerimiento previo de traducción para instrucciones con privilegios por parte del VMM. Cuando una plataforma *AMD-V* o *Intel VT* detecta la presencia del VMM, ésta le permite ejecutarse en el nivel -1, al que denomina nivel *super-privilegiado*. El VMM mantiene el control de procesador, memoria y acceso al *hardware* del sistema a fin de coordinar el acceso a los recursos. Al mismo tiempo, el VMM también reserve espacio de direccionamiento *hardware* específico para cada MV, lo que produce aislamiento *hardware* entre MVs.

## **C.6. Conclusión**

Futuras implementaciones de los *chips* *AMD-V* e *Intel VT* mejorarán el soporte de paginación de memoria. La virtualización total, la paravirtualización y la primera generación de virtualización asistida por *hardware* cuentan con las *Shadow Page Tables* (SPT) para traducir el acceso de las máquinas virtuales a la RAM. Para gestionar la memoria, el VMM mantiene una SPT por cada MV. Cuando una MV intenta escribir en memoria, el VMM intercepta la petición, la traduce y la almacena en la SPT asociada a la MV. El resultado es un *overhead* significativo (del 25% al 75%) de la paginación de memoria. Las ‘*AMD-V's Nested Page Tables* (NPT)’ y las ‘*Intel VT's Extended Page Tables* (EPT)’ permiten traducción directa entre direcciones de memoria del SO huésped y direcciones físicas. NPT y EPT permiten a los sistemas operativos huéspedes de las

MVs modificar directamente sus propias tablas de paginación físicas y manipular sus propios errores de página. Con el VMM actuando esencialmente como un puente entre la MV y la memoria física, el cuello de botella creado por las SPT dejará de existir.



## D. Comparativa de hipervisores

### D.1. Análisis inicial

Como punto de partida para el estudio de mercado de los *hipervisores* se acudió a fuentes tomadas de internet, en particular a un estudio publicado en el portal *TechRepublic* titulado “*What’s the best virtualisation suite?*”. El texto, a pesar de resultar ligeramente obsoleto por no recoger las últimas versiones que se ofertan actualmente, sienta las bases estableciendo tres fabricantes a considerar: *VMware*, *Microsoft* y *Citrix*; además de cierta cantidad de teoría que contextualiza el estado del arte.


De otra de las fuentes consultadas, *VMgurú*, tomamos una tabla más actualizada de los *hipervisores* disponibles (*Tabla D.1*). En esta primera tabla comparativa se presentan las principales alternativas de virtualización, a saber: el ESX de *VMware*, Hyper-V de *Microsoft* y Xen de *Citrix*. Este último fue una versión libre hasta ser comprado en 2007 por la empresa *Citrix*, sin embargo otras distribuciones libres basadas en Xen han sido desarrolladas, como por ejemplo ‘*xVM Server*’ de *Sun* y *Red Hat*. La tabla tiene varios atractivos:

- Estar realizada por una consultora independiente, lo que en un primer momento garantiza neutralidad.
- Presentar características genéricas que permiten valorar la potencia de cada *hipervisor*.
- Mostrar varias versiones de cada *hipervisor*, incluidas las gratuitas, que permiten deducir la evolución y esfuerzos de cada fabricante de una versión a la siguiente.

Tras el análisis de la tabla, conversaciones entre los distintos responsables y otros criterios, se decidió realizar la siguiente criba:

- Establecer *Microsoft* como un posible candidato. Si bien demuestra ser el fabricante más novel en virtualización, su evolución ha sido la más rápida, su experiencia es garantía de confianza y, sobre todo, todo el *software* utilizado por CyC en sus proyectos es *Microsoft*, por lo que se supuso que la migración más sencilla de los antiguos sistemas a los nuevos sería a través de este *hipervisor*.
- Con el fin de no precipitarse en la decisión y ahondar más en el conocimiento de la virtualización, otro fabricante sería considerado para una segunda fase. Entre los restantes, por su experiencia, larga trayectoria y mejor comparativa de características, se optó por *VMware*.

Tabla D.1. Comparativa de hipervisores

 v1.3, 04-06-2009	VMware ESXi 3.5 (without vCenter Server)	VMware ESXi(i) 3.5 (with vCenter Server)	VMware vSphere 4 (with vCenter Server)	Microsoft Hyper-V Server 2008	Microsoft Server 2008 Enterprise w Hyper-V 1.0	Microsoft Server 2008 Enterprise w Hyper-V 2.0	Citrix XenServer 4 (with essentials pack)	Citrix XenServer 5.0 (with essentials pack)
Free	✓			✓				
<b>Business continuity</b>								
High Availability		✓	✓		✓ (MSCS)	✓ (MSCS)	✓ *1 (EvRun)	✓ *1 (EvRun)
Fault tolerance (zero downtime HA)			✓ *2				✓ *1 (EvRun)	✓ *1 (EvRun)
Disaster recovery		✓ (SRM)	✓ (SRM)					✓ (WS)
Storage replication support for disaster recovery	✓			✓	✓	✓	✓	
Live migration		✓ *3	✓ *3			✓	✓ *3	✓
<b>Storage</b>								
Local storage	✓	✓	✓	✓	✓	✓	✓	✓
Shared storage (FC, iSCSI)	✓	✓	✓	✓	✓	✓	✓	✓
Live storage migration (no downtime)		✓	✓					
<b>Back-up</b>								
Support for back-up and recovery		✓	✓		✓	✓	✓	✓
Back-up proxy		✓	✓					
<b>Networking</b>								
VLAN support (802.1q)	✓	✓	✓	✓	✓	✓	✓	✓
Link aggregation (803.2ad)	✓	✓	✓	✓ *5	✓ *5	✓ *5	✓	✓
<b>Core</b>								
Bare metal deployment	✓	✓	✓	✓	✓	✓		✓
Intel-VT or AMD-V needed	No	No	Yes	Yes	Yes	Yes	Yes	Yes
Max CPU support (logical/cores)	32	32	64	24	24	32	2-Unlimited *3	Unlimited
Max Memory support	256GB	256GB	1TB	32GB	32GB	1TB	4GB-128GB *3	Unlimited
<b>Virtual machines</b>								
Max vCPU	4	4	8	4	4	4 *6	8	8
Max memory	64GB	64GB	255GB	31,5GB	31,5GB	64GB	32GB	32GB
Max network I/O	9Gb/s	9Gb/s	40Gb/s	[unknown]	[unknown]	[unknown]	[unknown]	[unknown]
Max storage I/O	100.000	100.000	300.000+	[unknown]	[unknown]	[unknown]	[unknown]	[unknown]
<b>Operating system support (servers)</b>								
Windows NT	✓	✓	✓					
Windows 2000	✓	✓	✓	SP4, max 1 CPUs	SP4, max 1 CPUs	SP4, max 1 CPUs	✓ (no x64)	✓ (no x64)
Windows 2003	✓	✓	✓	SP2, max 2 CPUs	SP2, max 2 CPUs	SP2, max 2 CPUs	✓	✓
Windows 2008	✓	✓	✓	Max 4 CPUs	Max 4 CPUs	Max 4 CPUs	✓	✓
Mandrake Linux	✓	✓	✓					
Ubuntu Linux	✓	✓	✓					
SUSE Linux	✓	✓	✓	10 SP1/2, 1 CPU	10 SP1/2, 1 CPU	10 SP1/2, 1 CPU	✓	✓
RedHat Linux	✓	✓	✓				✓	✓
FreeBSD	✓	✓	✓					
Netware	✓	✓	✓					
SUN Solaris	✓	✓	✓					
<b>Management</b>								
Role based management	✓	✓	✓	✓	✓	✓	✓	✓
P2V migration		✓	✓	✓	✓	✓	✓	✓
Zero downtime host maintenance		✓	✓			✓	✓	✓
Auto VM placement while running		✓	✓				✓ *1	✓
Auto VM placement at startup		✓	✓				✓	✓
Live VM migration (no downtime)		✓	✓			✓	✓	✓

## D.2. Microsoft vs. VMware

La estrategia en esta segunda fase consistió en un análisis más profundo, corroborando las características que cada *hipervisor* mostraba, explorando sus sitios web, echando un primer vistazo a sus versiones de prueba y contactando con sus departamentos comerciales para escuchar sus diferentes ofertas.

El resultado se recogió en un informe que consta de visión técnica, económica y estratégica.

### D.2.1. Informe Técnico

Con las diferentes características señaladas por ambos fabricantes, se elaboró una tabla comparativa propia (*Tabla D.2*) que sirviera para analizar el potencial de cada herramienta.

*Tabla D.2. vSphere frente a Hyper-V*

		<b>vSphere 4</b>	<b>Hyper-V R2</b>
1	Memory Overcommit	Si	No
2	Clustered File System	Si	Si (CSV)
3	Automated VM Failover	Si (High Availability)	Si (Failover Clustering)
4	Built-in online/offline VM Patching	Si (Update Manager)	Si (SCCM/SCVMM)
5	Integrated Disk Backup with Data Deduplication	Si (Data Recovery)	Si (Data Protection Manager)
6	Storage Thin Provisioning	SI	Si
7	VM Live Migration	Si (varias simultáneamente)	Si
8	Dynamic Resource Scheduling	Si	Si (PRO mediante SCVMM y SCOM)
9	Power Management	Si	No (Pero se puede alcanzar)
10	Host Profiles	Si	Si
11	Cross-Hypervisor Management	No	Si
12	Application Management	No	Si
13	End-to-End OS Monitoring	No	Si
14	Application/Service Monitoring	No	Si
15	Physical Management	No	Si
16	Management Extends to Desktop	No	Si
17	Ultra-Thin Hypervisor	Si (Ligero)	No (Pesado)
18	Hot Add	Si	No (solo disco)
19	VM Fault Tolerance	Si	No
20	Firewall Virtual Appliance	Si	No
21	Storage Live Migration	Si	No (Quick)
22	Distributed Network Switch	Si	No

A continuación, se explican las características de algunos elementos de la tabla:

- 1.- Posibilidad de declarar más memoria de la total física disponible aprovechando que las MV no utilizan el 100% de la memoria declarada la mayor parte del tiempo.
- 2.- Acceso al mismo LUN desde varios nodos del *cluster* simultáneamente (posibilita el *failover* automático).

3.- Se refiere a la característica de continuidad de negocio que permite que cuando una MV del *cluster* cae, es automáticamente levantada en otro equipo del *cluster* sin pérdida de datos ni funcionalidad.

5.- *Data Deduplication* es una característica por la que si varias instancias de la MV tienen datos idénticos, por ejemplo un mismo correo, únicamente se guarda una copia de los mismos para todas las instancias. Hyper-V no indica específicamente que lo soporte, probablemente debido a que *Data Protection Manager* es un *software* independiente.

6.- Discos dinámicos (almacenamiento compartido).

7.- Migración de máquinas en caliente.

8.- Balanceo de carga.

9.- Balanceo de carga para lograr apagar máquinas. *Microsoft* asegura poder alcanzarlo mediante el uso de SCOM y la generación de un *script* apropiado.

10.- Posibilidad de predefinir perfiles del host posteriormente reutilizables.

11, 12, 13, 14, 15 ,16.- A través de *System Center* se pueden gestionar y monitorizar tanto *hipervisores* de diferentes fabricantes, como aplicaciones y sistemas operativos corriendo dentro de las MV.

17.- Se refiere al tamaño del *hipervisor* en disco, sin considerar actualizaciones y *patches*.

18.- Añadir/quitar CPU/RAM/disco asignado a una máquina en caliente.

19.- Asegura que las migraciones, especialmente las debidas a *Live Migration*, que proporcionan alta disponibilidad (HA), se producen sin pérdida de datos y garantizando la continuidad de las máquinas.

21.- *Quick* se refiere a que la opción existente de *Microsoft* no permite migrar realmente “en caliente” (*live*).

22.- Control centralizado de la red virtual dentro del *cluster*, que permite no tener que preocuparse por la influencia de las migraciones de las MV en la configuración de la red.

## Sistemas operativos soportados

### *vSphere*

*Tabla D.3. Sistemas operativos soportados por vSphere*

48	Windows	NT 4.0, 2000, Server 2003, Server 2008, XP, Vista, 98, 95 3.1
	MS-DOS	6.22
	Red Hat	Enterprise Linux 5, 4, 3 y 2.1
	SUSE	Linux Enterprise Server 11, 10, 9 y 8
	Ubuntu	Linux 8.04 y 7.1
	Novell	NetWare 6 y 5.1
	OS/2	Warp 4
	Sun	Solaris 10
	SCO	Open Server 5, Unixware 7
	Open	Enterprise Server 2
	CentOS	5 y 4
	Debian	5 y 4
	FreeBSD	7 y 6
	Asianux	3

### *Hyper-V*

*Tabla D.4 Sistemas operativos soportados por Hyper-V*

13	Windows	2000, Server 2003, Server 2008, XP, Vista
	Red Hat	Enterprise Linux 5
	SUSE	Linux Enterprise Server 10

### **D.2.2. Informe Económico**

Este análisis (*Tabla D.5*) no supone un presupuesto, sino un vistazo al precio de cada ‘paquete’ de *software*. Se pueden consultar análisis más detallados y una discusión crítica sobre los mismos en los blogs “*Virtualboy*” y “*VMware: Virtual Reality*” y otros tantos en los que, con la intención de desmontar la maquinaria publicitaria de la competencia, personas muy cercanas a las empresas de virtualización argumentan sobre costes, funcionalidades y otras características de sus respectivos productos.

Tabla D.5. Análisis de precios de vSphere e Hiper-V

	VMware vSphere w/ vCenter Std.				Hyper-V R2 with VMM		
	Std	Adv	Ent	Ent+	Std	Ent	DC
Max host processors	N/A	N/A	N/A	N/A	4	8	64
Max host processor cores	64	64	64	64	N/A	N/A	N/A
Max cores/processor	6	12	6	12	N/A	N/A	N/A
Max virtual SMP	4	4	4	8	4	4	4
Max host RAM (GB)	256	256	256	1024	32	2048	2048
Max RAM/vm	255	255	255	255	64	64	64
Failover nodes	N/A	32	32	32	N/A	16	16
Memory overcommit	Yes	Yes	Yes	Yes	No	No	No
Transparent page sharing	Yes	Yes	Yes	Yes	No	No	No
Live Migration	No	Yes	Yes	Yes	No	Yes	Yes
Simultaneous Live Mig.	Yes	Yes	Yes	Yes	No	No	No
Live guests per host	320	320	320	320	384	384	384
Live guests/cluster node	N/A	160*	160*	160*	N/A	64	64
Distributed Resource Scheduler	No	No	Yes	Yes	No	No	No
Snapshots per VM	32	32	32	32	50	50	50
Thin Provisioning	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Storage Live Migration	No	No	Yes	Yes	No	No	No
Distributed Switch	No	No	No	Yes	No	No	No
Direct I/O	Yes	Yes	Yes	Yes	No	No	No
Max. partition size (TB)	2	2	2	2	Notes	Notes	Notes
Application firewall (vShield)	No	Yes	Yes	Yes	No	No	No
Virtual instance rights	N/A	N/A	N/A	N/A	1	4	Unl.
Hypervisor licensing	/proc	/proc	/proc	/proc	/host	/host	/proc
Pricing	\$795	\$2,245	\$2,875	\$3,495	\$999	\$3,999	\$2,999

Las anotaciones a esta tabla son:

Versión *stand-alone* de Hyper-V: *Hyper-V Server 2008 R2* con las mismas funcionalidades técnicas que las versiones *Enterprise* y *Datacenter* GRATIS, es decir, la licencia que se paga en Hyper-V corresponde al software *Windows Server 2008 R2*.

La suites de gestión (*vCenter* y *System Center* respectivamente) no están incluidas y son necesarias.

### D.2.3. Informe Estratégico

- La solución de *VMware* resulta más integrada que la de *Microsoft* en cuanto a los servicios que ofrece, dado que las funcionalidades de este último vienen ofrecidas a través de varios *software* que se deben orquestar para que trabajen juntos, y cuyas licencias deben ser adquiridas aparte.



- El software de gestión de *Microsoft, System Center*, es una *suite* muy potente con un elevado número de funcionalidades que amplían la solución inicial. En este caso, es la suma de las piezas la que lo hacen tan potente, si bien de nuevo se requiere orquestarlas todas juntas.
- *Microsoft* está poniendo gran empeño en lograr excelencia en virtualización, pero todavía está emergiendo. *VMware* posee amplia experiencia, aunque sus licencias son más caras.
- La versión *stand-alone* (sin *Windows Server*) de Hyper-v es gratuita.
- Empiezan a existir conflictos de licenciamiento con el sistema empleado por *VMware* (licencias por núcleo) debido a la aparición de procesadores *8-core* y *12-core*. Desde *VMware* están trabajando por obtener un modelo de licencia más eficiente.

### **D.3. Conclusiones**

A la vista de los datos presentados y las tareas llevadas a cabo para recopilarlos, las diferentes reuniones y consultas y aplicando criterios subjetivos, se optó por estudiar la solución de *VMware* para su instalación como *software* de virtualización de la infraestructura de CyC.

## E. Tipos de almacenamiento

### E.1. Introducción

En esta guía se analizarán cuatro tipos de arquitecturas de almacenamiento:

- *Almacenamiento de conexión directa (DAS)*: El más sencillo de los tipos de almacenamiento de datos, es el que se encuentra en el propio servidor o conectado directamente a éste.
- *Almacenamiento conectado a la red (NAS)*: Un servidor de archivos dedicado conectado a una red de área local que ejecuta un sistema operativo que se dedica de forma específica a la distribución de archivos.
- *Red de área de almacenamiento (SAN)*: Una red dedicada para el tráfico de almacenamiento entre servidores y un *array* de almacenamiento en disco o dispositivo de cinta.
- *NAS/SAN híbrido o unificado (puerta de acceso NAS)*: la combinación de una puerta de acceso NAS con una SAN y un híbrido de NAS/SAN, con una interfaz de administración sencilla para ocultar la complejidad.

A continuación se enumeran algunos de los protocolos de red que se encuentran en los subsistemas de almacenamiento actuales:

- *Fibre Channel (FC)*: El protocolo *Fibre Channel* de baja latencia se ha diseñado de forma específica para entornos y redes de almacenamiento mayores y más exigentes. Proporciona un alto nivel de rendimiento y fiabilidad cuando se emplea entre servidores y dispositivos de almacenamiento.
- *IP SCSI*: iSCSI es un método de conexión de servidores y almacenamiento a través de *Ethernet*. Una red *Ethernet* de 1 Gb/seg. dedicada proporcionará el suficiente ancho de banda para conectar un dispositivo de almacenamiento a varios servidores.
- *Serial-attached SCSI (SAS)*: Con la introducción de los sistemas DAS compartidos, SAS es una alternativa emergente para el almacenamiento de aplicaciones en red. Constituye una manera sencilla y asequible de compartir almacenamiento entre un número fijo de servidores de conexión directa.

A la vez, son diversas las tecnologías de unidad de disco que se pueden implantar en una red de almacenamiento (*Tabla E.1*), cada una de ellas con ventajas complementarias distintas. A continuación se exponen algunas de las empleadas en los sistemas de almacenamiento actuales:

- *Serial ATA (SATA)*: Si el rendimiento no es un factor determinante y su negocio requiere almacenamiento rentable de gran capacidad para la distribución de

archivos, el archivo de datos o información de referencia, la conexión de tecnología avanzada serie (SATA) es la opción perfecta. Los discos SATA resultan mucho más rentables por gigabyte que los discos SCSI, SAS o *Fibre Channel*.

- *Serial-attached SCSI (SAS)*: Con un rendimiento mejorado en comparación con los discos SATA, las unidades de disco SCSI de conexión serie (SAS) ofrecen la velocidad, la fiabilidad y la alta disponibilidad que requieren el almacenamiento y las aplicaciones en línea. Las unidades de disco SAS individuales cuentan con menor capacidad que los discos SATA, pero su rendimiento es mucho mayor y se pueden agrupar en diversas configuraciones RAID que consiguen llegar prácticamente a cualquier nivel de capacidad.
- *Fibre Channel (FC)*: Las unidades de disco *Fibre Channel* (FC) se han diseñado fundamentalmente para el rápido procesamiento de los datos en sistemas de almacenamiento de gran capacidad, funcionamiento intensivo y alta disponibilidad que requieren la máxima escalabilidad. Con el mayor coste por *gigabyte* de todas las opciones descritas, las unidades de disco FC son una buena alternativa para las aplicaciones de misión crítica más exigentes.

*Tabla E.1. Comparativa entre tecnologías de unidad de disco*

Comparación de tecnologías de unidad de disco

	Serial ATA (SATA)	SCSI	Serial-attached SCSI (SAS)	Fibre Channel
Precio por GB	Bajo	Moderado	Moderado	Elevado
Capacidad	La más alta	Moderada	Moderada	Moderada
Fiabilidad	Moderada	Elevada	Elevada	Elevada
Rendimiento	Bajo	Elevada	Elevada	Elevada
Escalabilidad	Moderada	Baja	Moderada	Elevada
Acceso a datos	Moderado	Frecuente	Frecuente	Frecuente
Mejor uso	Almacenamiento de archivos, archivo/copia de seguridad, aplicaciones secundarias	Transacciones empresariales y aplicaciones principales	Transacciones empresariales y aplicaciones principales	Transacciones empresariales y aplicaciones principales

## E.2. Almacenamiento de conexión directa (DAS)

El tipo más sencillo de almacenamiento de datos es el denominado de conexión directa (DAS). En una configuración DAS (*Figura E.1*), uno o más componentes de almacenamiento de datos, por ejemplo, discos duros o unidades de cinta, se instalan en un equipo o se conectan directamente a él, con frecuencia por medio de un enlace SCSI o SAS. Una opción de creciente popularidad en el almacenamiento es DAS compartido, donde un número fijo de servidores se conectan directamente a un sistema de almacenamiento en lugar de emplear una red de almacenamiento o estructura. En el almacenamiento de conexión directa (DAS), cada servidor se configura con su propio dispositivo de almacenamiento.



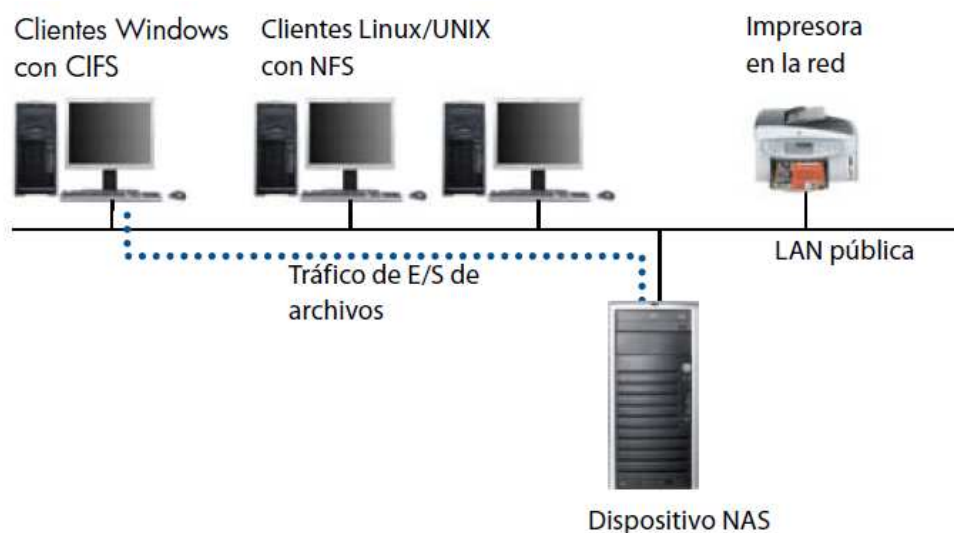
- Por almacenamiento de conexión directa se puede hacer referencia a las unidades dentro de un servidor o a una caja de almacenamiento externa

*Figura E.1. Estructura DAS*

### E.3. Almacenamiento conectado a la red (NAS)

El almacenamiento conectado a la red (NAS) es fundamentalmente un servidor de archivos dedicado que ejecuta un sistema operativo diseñado y configurado de forma específica para gestionar el tráfico de E/S de archivos para clientes de red (*Figura E.2*). El acceso del equipo cliente al servidor NAS se realiza por lo general mediante una conexión *Ethernet*. El servidor NAS aparece en la red como un único nodo con su propia dirección IP. Los archivos almacenados en el sistema NAS están disponibles para los clientes en la red LAN a través de una conexión *Ethernet* mediante protocolos como CIFS/SMB (clientes *Windows*) o NFS (*Linux* y *UNIX*). Muchos sistemas NAS también admiten protocolos como HTTP o FTP para el acceso a archivos basado en internet.

Los productos NAS van desde dispositivos de calidad de oficina en casa de bajo coste a puertas de acceso NAS empresariales completamente agrupadas que se conectan a una SAN. En el almacenamiento conectado a la red (NAS), todos los clientes disponen de acceso al mismo almacenamiento a través de la red de área local (LAN).



*Figura E.2. Estructura NAS*

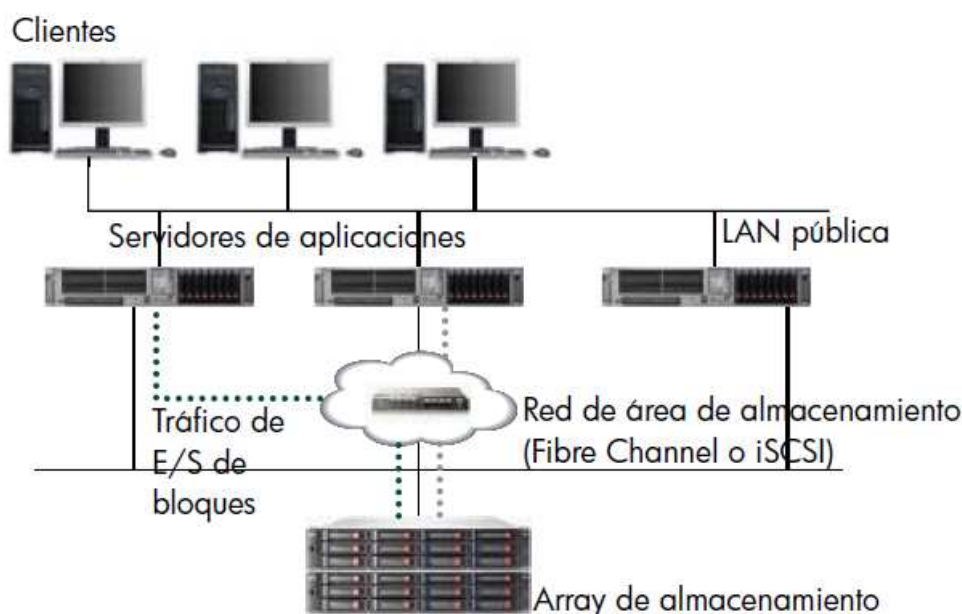
#### E.4. Red de área de almacenamiento (SAN)

Una red de área de almacenamiento (SAN) es simplemente eso, una red que se dedica al almacenamiento. La red SAN (*Figura E.3*) está separada de la LAN y proporciona a los servidores acceso al almacenamiento. La red se ha diseñado para gestionar las comunicaciones del almacenamiento.

En su forma más sencilla, una SAN consta de los siguientes elementos:

- Almacenamiento compartido, por lo general, un *array* de discos.
- La red dedicada que permite la comunicación entre servidores y almacenamiento.
- El *software* que configura el entorno compartido.
- Los componentes que protegen los datos en la red SAN.

Para garantizar un rendimiento óptimo, la red entre servidores y almacenamiento es por lo general del tipo *Fibre Channel*, iSCSI de varios *gigabits* u otro enlace de alta velocidad. Una red de área de almacenamiento (SAN) es una red de alta velocidad dedicada en exclusiva al almacenamiento, que está separada de la LAN pública.



*Figura E.3. Estructura SAN*

##### Software de configuración:

En una SAN, una vez que los servidores se encuentran conectados físicamente al *array* de discos, se precisa *software* de configuración que proporcione dicha capacidad de almacenamiento a aplicaciones específicas que se ejecutan en el servidor. El *software* de configuración podría ser una aplicación que se ejecute en uno de los servidores conectados, o bien un *software* que se ejecute directamente desde el dispositivo de almacenamiento.

##### Configuración de discos físicos:

El primer paso de la configuración implica agrupar discos físicos en un grupo RAID en función de los requisitos de rendimiento, redundancia y capacidad. La

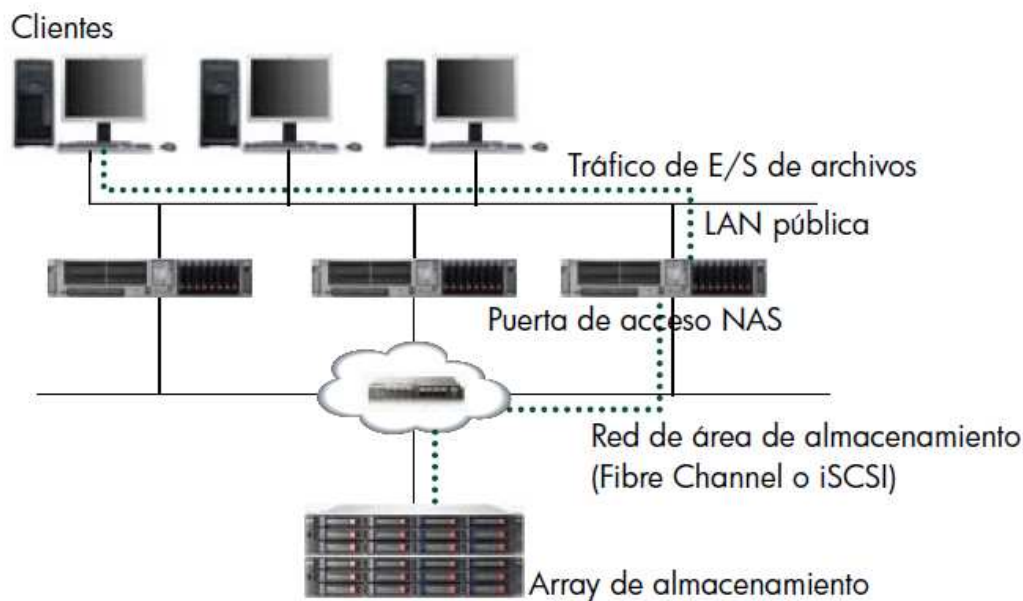
tecnología RAID protege a los discos en la SAN de fallos de unidad física y puede mejorar el rendimiento de las aplicaciones diseminando las lecturas /escrituras de datos en varios discos.

#### Configuración de discos lógicos:

Una vez que se han creado los grupos RAID, porciones de ellos se pueden asignar a uno o más servidores. El *software* toma porciones de la capacidad de disco y reparte dicha capacidad al servidor correspondiente. Por ejemplo, puede haber 2,5 *terabytes* de almacenamiento protegido en una configuración RAID 5, con tres servidores conectados a la SAN. Las capacidades apropiadas se seleccionan y asignan a cada servidor.

### **E.5. Puerta de acceso NAS con una SAN**

Un uso común de la red de área de almacenamiento es almacenar datos de archivo y proporcionar acceso a dichos datos a través de una puerta de acceso NAS. La puerta de acceso no tiene almacenamiento incorporado; en su lugar, se conecta a una SAN y actúa como traductor entre los protocolos NAS de nivel de archivo, como NFS y CIFS, y los protocolos SAN de nivel de bloque, como *Fibre Channel* e iSCSI. El uso de una puerta de acceso NAS a una SAN (*Figura E.4*) combina las ventajas de las dos tecnologías y proporciona tanto E/S de archivos como de bloques.



*Figura E.4. Estructura híbrida NAS/SAN*

