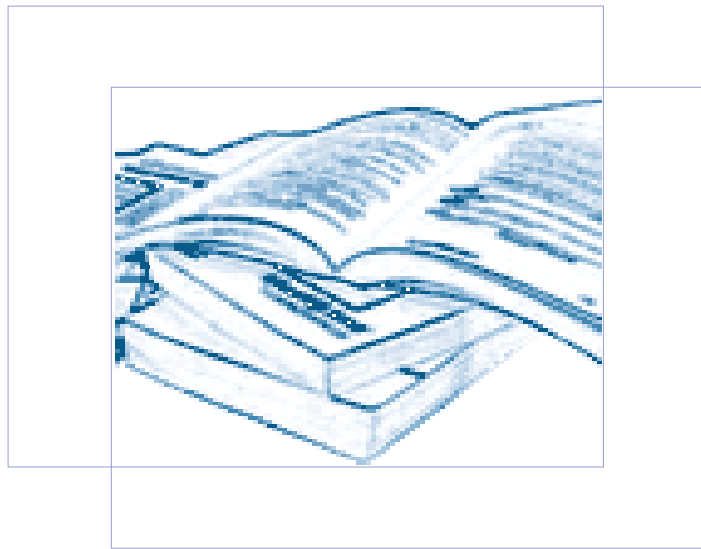


ANEXOS



- ❖ Anexo 1: MANUALES DE INSTALACIÓN, CONFIGURACIÓN Y USO DE PHP EN LINUX Y EN WINDOWS
- ❖ Anexo 2: MANUALES DE INSTALACIÓN, CONFIGURACIÓN Y USO DE ASP EN LINUX Y EN WINDOWS
- ❖ Anexo 3: MANUALES DE INSTALACIÓN, CONFIGURACIÓN Y USO DE JSP EN LINUX Y EN WINDOWS
- ❖ Anexo 4: TABLAS

CONTENIDO

ASPECTOS A TOMAR EN CUENTA PARA LAS INSTALACIONES.....	4
INSTALACIONES EN LINUX	4
INSTALACIONES EN WINDOWS	5
ANEXO 1	
MANUALES DE INSTALACIÓN, CONFIGURACIÓN Y USO DE PHP EN LINUX Y EN WINDOWS	7
INSTALACIÓN, CONFIGURACIÓN Y USO DE PHP EN LINUX.....	7
INSTALACIÓN VÍA RPM(REDA HAT PACKAGE MANAGER)	7
INSTALACIÓN POR MEDIO DE BINARIOS.....	10
INSTALACIÓN Y CONFIGURACIÓN DE POSTGRESQL	10
INSTALACIÓN Y CONFIGURACIÓN DE APACHE WEB SERVER.....	13
INSTALACIÓN DE PHP COMO MÓDULO DE APACHE WEB SERVER.....	14
VERIFICACIÓN DE LAS INSTALACIONES.....	17
INSTALACIÓN, CONFIGURACIÓN Y USO DE PHP EN WINDOWS.....	20
INSTALACIÓN Y CONFIGURACIÓN DE POSTGRESQL	20
INSTALACIÓN DE CYGWIN.....	20
INSTALACIÓN DE CYGIPC	22
INSTALACIÓN Y CONFIGURACIÓN DE APACHE WEB SERVER.....	23
INSTALACIÓN Y CONFIGURACIÓN DE PHP	24
VERIFICACIÓN DE LAS INSTALACIONES.....	27
ANEXO 2	
MANUALES DE INSTALACIÓN, CONFIGURACIÓN Y USO DE ASP EN LINUX Y EN WINDOWS	32
INSTALACIÓN, CONFIGURACIÓN Y USO DE ASP EN LINUX.....	32
INSTALACIÓN DE CHILI!SOFT ASP.....	33
INSTALACIÓN Y CONFIGURACIÓN DE POSTGRESQL	35
CONFIGURACIÓN DEL CONTROLADOR ODBC PARA POSTGRESQL.....	35
VERIFICACIÓN DE LAS INSTALACIONES.....	37
INSTALACIÓN, CONFIGURACIÓN Y USO DE ASP EN WINDOWS	41
INSTALACIÓN Y CONFIGURACIÓN DE POSTGRESQL	41
INSTALACIÓN Y CONFIGURACIÓN DE INTERNET INFORMATION SERVER	41
INSTALACIÓN Y CONFIGURACIÓN DE ASP.....	42
VERIFICACIÓN DE LAS INSTALACIONES.....	43


ANEXO 3

MANUALES DE INSTALACIÓN, CONFIGURACIÓN Y USO DE JSP EN LINUX Y EN WINDOWS	47
INSTALACIÓN, CONFIGURACIÓN Y USO DE JSP EN LINUX	47
INSTALACIÓN VÍA RPM	47
INSTALACIÓN POR MEDIO DE BINARIOS.....	48
INSTALACIÓN Y CONFIGURACIÓN DE POSTGRESQL	48
INSTALACIÓN Y CONFIGURACIÓN DEL JDK (ENTORNO JAVA PARA JSP).....	49
CONFIGURACIÓN DE JDBC.....	49
INSTALACIÓN Y CONFIGURACIÓN DE APACHE WEB SERVER.....	51
INSTALACIÓN Y CONFIGURACIÓN DE TOMCAT	52
CREACIÓN E INSTALACIÓN DE MOD_JK	52
VERIFICACIÓN DE LAS INSTALACIONES.....	55
INSTALACIÓN, CONFIGURACIÓN Y USO DE JSP EN WINDOWS	58
INSTALACIÓN Y CONFIGURACIÓN DE POSTGRESQL	58
INSTALACIÓN Y CONFIGURACIÓN DEL JDK (ENTORNO JAVA PARA JSP).....	58
CONFIGURACIÓN DE JDBC.....	59
INSTALACIÓN Y CONFIGURACIÓN DE APACHE WEB SERVER.....	60
INSTALACIÓN Y CONFIGURACIÓN DE TOMCAT	60
CREACIÓN E INSTALACIÓN DE MOD_JK	61
VERIFICACIÓN DE LAS INSTALACIONES.....	63
ANEXO 4	
TABLAS	67
Tabla 1: Comparación de los Sistemas Operativos Windows 2000 Server y Linux Red Hat(6.0 -7.x)	69
Tabla 2: Combinación de Herramientas para programar Aplicaciones Web.	81
Tabla 3: Servicios proporcionados en Internet y sus respectivos puertos.	82
Tabla 4: Evaluación de Servidores Web gratuitos y comerciales que trabajan sobre WINDOWS / LINUX y soportan: PHP/ASP/JSP.....	83
Tabla 5: Directivas de configuración del archivo httpd.conf de Apache Web Server (Ejemplos en Linux)	90
GLOSARIO	93
BIBLIOGRAFÍA	98

ASPECTOS A TOMAR EN CUENTA PARA LAS INSTALACIONES

Las instalaciones de los paquetes en Linux, se pueden realizar de 3 formas:

- Instalación junto con el sistema operativo.
- Instalación vía **RPM**(Red Hat Packages Manager – Administrador de Paquetes Red Hat).
- Instalación por medio de binarios para compilar.

 **Nota:** En este documento se describen las dos últimas formas de instalar: vía RPM y binarios para compilar.

Las instalaciones de los paquetes en Windows se realizarán mediante instaladores ejecutables y copiando archivos.

INSTALACIONES EN LINUX

Si se instala vía RPM se debe tomar en cuenta los siguientes aspectos:

- Verificar que no se encuentren instaladas ni en funcionamiento otras versiones de cualquiera de los paquetes(base de datos, lenguaje del lado del servidor, servidor web).
- Verificar que se instalen todos los paquetes y sub-paquetes(librerías, módulos, etc.) necesarios.
- El sistema operativo debe ser Red Hat Linux, ya que éste cuenta con el RPM.
- Verificar que el tamaño en disco sea suficiente, tanto para las descargas e instalaciones.
- Las versiones de los paquetes instaladores dependerán de la versión de Linux que se instale, en este caso se deberá primero verificar con que versiones se cuenta.

Si se instala por medio de binarios para compilar:

- Verificar que no se encuentren instaladas ni en funcionamiento otras versiones de cualquiera de los paquetes(base de datos, lenguaje del lado del servidor, servidor web).
- Crear directorios o verificar la existencia de los mismos para descargas e instalaciones.
- Descargar las versiones indicadas de los sitios señalados(versiones probadas).
- Verificar que se cuente con utilidades para descomprimir y desempaquetar archivos **.tar.gz**.
- Instalar mediante la consola de comandos.
- Leer los archivos de configuración y la documentación que trae consigo cada instalador.
- Verificar que se tenga permisos para ejecutar las instalaciones.

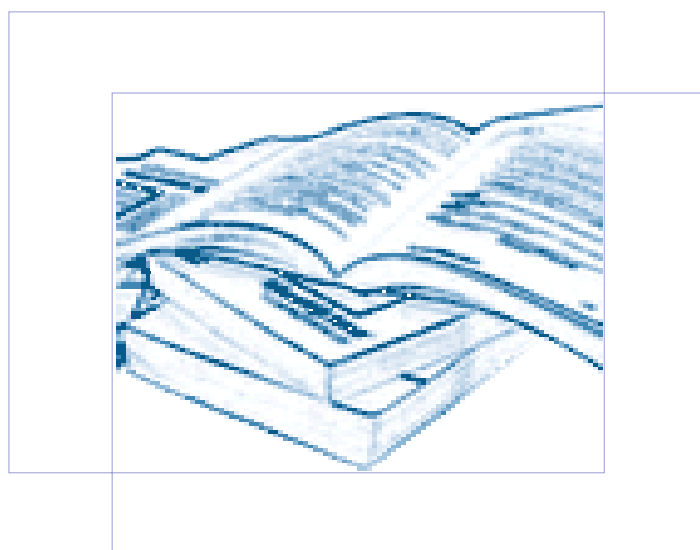
- Verificar que el tamaño en disco sea suficiente tanto para las descargas e instalaciones.
- Instalar de preferencia en Red Hat Linux.
- Verificar que las versiones a instalar sean similares a las indicadas en este documento. Para versiones inferiores o superiores el proceso de instalación puede variar.

INSTALACIONES EN WINDOWS

Al instalar en Windows se debe tomar en cuenta los siguientes aspectos:

- Verificar que no se encuentren instaladas ni en funcionamiento otras versiones de cualquiera de los paquetes(base de datos, lenguaje del lado del servidor, servidor web).
- Descargar las versiones indicadas de los sitios señalados(versiones probadas).
- Verificar que se cuente con utilidades para descomprimir archivos **.zip**.
- Leer los archivos de configuración y la documentación que trae consigo cada instalador.
- Verificar que el tamaño en disco sea suficiente tanto para las descargas e instalaciones.
- Instalar de preferencia en Windows 2000 Server.
- Verificar que las versiones a instalar sean similares a las indicadas en este documento. Para versiones inferiores o superiores el proceso de instalación puede variar.

ANEXO 1



MANUALES DE INSTALACIÓN, CONFIGURACIÓN Y USO DE PHP EN LINUX Y EN WINDOWS

Para el uso de PHP, tanto en Linux como en Windows, se necesita de una base de datos y un servidor web. En este Anexo se detalla el proceso de instalación, configuración y uso de PHP, PostgreSQL y Apache Web Server.

INSTALACIÓN, CONFIGURACIÓN Y USO DE PHP EN LINUX

INSTALACIÓN VÍA RPM(Red Hat Package Manager)

Para instalar vía RPM se debe seleccionar del CD de instalación los siguientes paquetes:

- php-4.2.2-80.5.rpm
- postgresql-server.rpm y postgresql-7.2.2-1.1.rpm
- apache-2.0.40.8.rpm

Adicionalmente, se instalarán de forma automática los siguientes paquetes: postgresql-devel.rpm, postgresql-docs.rpm, postgresql-jdbc.rpm, postgresql-odbc.rpm, postgresql-libs.rpm, postgresql-perl.rpm y postgresql-python.rpm.

Las instalaciones se realizan como **root**.

Luego de realizar la instalación es importante verificar los directorios donde se han instalado cada uno de los paquetes; de esta forma podemos conocer donde encontrar los archivos de configuración.

Con la instalación vía RPM, la mayoría de configuraciones se realizan automáticamente, sólo hace falta editar los archivos: **hosts**, **httpd.conf** de Apache Web Server y **pg_hba.conf** de PostgreSQL. Estos archivos se encuentran en los siguientes directorios: **/etc/hosts**, **/etc/httpd/conf** y **/var/lib/pgsql/data** respectivamente.

En el archivo **hosts** debe constar el mismo nombre de máquina que esté en la directiva **ServerName** dentro de **httpd.conf**.

En el archivo **httpd.conf**, además de editar la directiva **ServerName**, se deben editar las directivas: **ServerRoot**, **AddType**, **Action**, **DirectoryIndex**, **DocumentRoot**.

```
#ServerName localhost (nombre del dominio, de la máquina o dirección IP)
*****

#ServerRoot "/etc/httpd/"
*****

#AddType application/x-httpd-php .php php4 .phtml
#AddType application/x-httpd-php-source .phps
#Action application/x-httpd-php "/usr/bin/php.exe"
*****

#DirectoryIndex index.html index.htm index.shtml index.php index.php4 index.phtml
*****

#DocumentRoot "/var/www/html"
#<Directory "/var/www/html">
```

Todas las directivas de configuración de **httpd.conf** se encuentran detalladas en el **Anexo 4, Tabla 5**.

En el archivo **pg_hba.conf** debe constar el nombre de máquina, dirección IP, máscara y método de encriptación tanto de la máquina servidor como de las máquinas cliente que se van a conectar a PostgreSQL.

```
# cd /var/lib/pgsql/data
# vi pg_hba.conf
# TYPE          DATABASE    USER        IP-ADDRESS    IP-MASK       METHOD
localhost       all         all         127.0.0.1    255.255.255.255  trust
# exit
```

Luego de instalar PostgreSQL, se ingresa como usuario **postgres** para iniciar el servidor de la base de datos, las siguientes líneas indican los pasos:

```
# su postgres
# cd /var/lib/pgsql/bin
# ./initdb -D /var/lib/pgsql/data
```

El script **initdb** se ejecuta únicamente la primera vez que arranca el motor de PostgreSQL y crea las bases de datos básicas del sistema. Al ingresar como usuario **postgres** se mantiene la seguridad del sistema.

Para comenzar a trabajar con PostgreSQL, se debe primero iniciar el motor de **SQL** con el siguiente comando:

```
#!/postmaster -i -D /var/lib/pgsql/data
```


El script **postmaster** se ejecuta cada vez que se necesite trabajar con la base de datos. Para ejecutar este script es necesario ingresar como usuario **postgres**.

En este punto ya se puede crear bases de datos con sus respectivas tablas y acceder a éstas por medio del cliente **psql**. Los siguientes pasos indican el proceso, para ello se debe abrir otra consola e ingresar como usuario **postgres**:

```
# su postgres
# CREATEDB prueba;
# psql prueba
prueba => CREATE TABLE usuarios {
    nombre varchar(10),
    clave varchar(10),
    tipo_usuario varchar(10)
};
prueba => \q
```

Una vez que se ha creado una base de datos, se debe crear el usuario **apache** y darle privilegios para trabajar sobre las tablas:

```
#createuser apache;
# psql prueba
prueba => GRANT SELECT ON usuarios TO apache;
prueba => \q
# exit
```

 **Nota:** Para verificar las instalaciones, se debe seguir los mismos pasos de la sección [VERIFICACIÓN DE LAS INSTALACIONES](#).

INSTALACIÓN POR MEDIO DE BINARIOS

Al realizar este tipo de instalación se debe seguir en orden los siguientes pasos:

1. Instalación y configuración de la base de datos: PostgreSQL
2. Instalación y configuración del servidor web: Apache Web Server
3. Instalación y configuración de PHP

INSTALACIÓN Y CONFIGURACIÓN DE POSTGRESQL

Paquete: Servidor de base de datos PostgreSQL para Linux

Sitio Web: <ftp://ftp4.es.postgresql.org/Mirror/www.postgresql.org/v7.1.3/postgresql-7.1.3.tar.gz>

Nombre del archivo instalador: postgresql-7.1.3.tar.gz

Directorio de descarga del instalador: /usr/src

1. Crear la cuenta del superusuario de PostgreSQL. Este superusuario debe ser un usuario común, sin privilegios de **root**. Para crear el usuario **postgres** ingresamos al sistema como **root** y en el prontuario procedemos como sigue:

```
# su
# /usr/sbin/adduser postgres
# passwd postgres
```

2. Crear directorios para el código fuente y para la instalación y dar permisos al usuario **postgres**.

```
# cd /usr/src
# mkdir pgsq|
# chown postgres:postgres pgsq|
# cd /usr/local
# mkdir pgsq|
# chown postgres:postgres pgsq|
# exit
```

Los directorios **/usr/src** y **/usr/local** son para código fuente y para programas de usuarios respectivamente. El usuario **postgres** debe ser propietario de ambos directorios.

3. Empezar el proceso de compilación/instalación de PostgreSQL. Para ello ingresamos con el

usuario **postgres**.

```
# su postgres
# cd /usr/src/pgsql
# gunzip postgresql-7.1.3.tar.gz
# tar -xvf postgresql-7.1.3.tar
# cd /usr/src/pgsql/postgresql-7.1.3
# ./configure --prefix=/usr/local/pgsql --with-odbc --enable-locale
```

En el script **configure** se indica el directorio donde se instalará PostgreSQL en el parámetro **prefix**, además se indica que será compilado con soporte **ODBC**.

4. Empezar el proceso de compilación.

```
# gmake all > make.log 2>&1 &
# tail -f make.log
```

Si hasta este punto todo va bien, **tail** muestra la última línea del archivo **.log**:

```
"All of PostgreSQL is successfully made"
Ready to Install
```

Presionar Ctrl +C para salir del archivo y volver al prompt.

5. Instalar los binarios(código fuente compilado) y archivos adicionales.

```
# gmake install > make.install.log 2>&1 &
# tail -f make.install.log
```

En este punto saldrá el siguiente mensaje indicando el fin de la instalación:

```
Thank you for choosing PostgreSQL, the most advanced open source database engine.
```

Presionar Ctrl +C para salir del archivo y volver al prompt.

6. Instalar la documentación de PostgreSQL.

```
# cd /usr/local/pgsql/postgresql-7.1.3/doc
# make install
```

Con **make install**, se realiza la compilación y creación de los archivos de documentación.

7. Indicar al sistema donde encontrar las librerías de PostgreSQL. Para ello se debe editar el archivo **ld.so.conf** del directorio **/etc**, ingresando como el superusuario **root**.

```
# su
# echo /usr/local/pgsql/lib >> /etc/ld.so.conf
# /sbin/ldconfig
# exit
```

La ejecución del programa **ldconfig** actualiza los enlaces de las librerías dinámicas.

8. Actualizar el archivo **.bash_profile** del usuario **postgres**. Este archivo se encuentra en el directorio **home** de dicho usuario. Para actualizar el archivo se debe añadir las variables de entorno que se indican a continuación y luego grabar los cambios.

```
# su postgres
# vi ~/.bash_profile
    PATH=$PATH:/usr/local/pgsql/bin
    MANPATH=$MANPATH:/usr/local/pgsql/man
    PGLIB=/usr/local/pgsql/lib
    PGDATA=/usr/local/pgsql/data
    LC_COLLATE=C
                                                    LC_CTYPE=C

    LC_COLLATE=C
    export PATH MANPATH PGLIB PGDATA LC_COLLATE LC_CTYPE LC_COLLATE
```

El directorio **data** dentro de **/usr/local/pgsql** debe ser creado en caso de que no exista.

9. Iniciar y arrancar PostgreSQL. Este paso arranca el motor de PostgreSQL y crea las bases de datos básicas del sistema. Se debe ingresar como usuario **postgres** para mantener la seguridad del sistema.

```
# su postgres
# cd /usr/local/pgsql/bin
# ./initdb -D /usr/local/pgsql/data
# ./postmaster -i -D /usr/local/pgsql/data
```

El script **initdb** inicia el servidor de PostgreSQL y el script **postmaster** lanza el servidor de **SQL** de

PostgreSQL. El script **initdb** se ejecuta sólo la primera vez, luego de instalar; el script **postmaster** se ejecuta cada vez que se necesite trabajar con PostgreSQL.

10. Iniciar el motor de PostgreSQL.

```
# ./postgresql start
```

En este punto el estado del servicio **postgresql** debe estar iniciado, por lo tanto ya se puede crear bases de datos propias.

Para ver el estado del servicio usamos el comando **status**, para detener el servicio **stop** y para reiniciar **restart**.

11. Permitir el acceso vía Web a PostgreSQL. En este paso se debe añadir información al archivo **pg_hba.conf** del directorio **/usr/local/pgsql/data/** y luego guardar los cambios.

```
# cd /usr/local/pgsql/data/  
# vi pg_hba.conf  
# TYPE          DATABASE      USER          IP-ADDRESS    IP-MASK       METHOD  
localhost      all           all           127.0.0.1     255.255.255.255  trust  
# exit
```

En el archivo **pg_hba.conf** se indica la **dirección IP** y la **máscara** del servidor y las máquinas cliente que se van a conectar a PostgreSQL.

INSTALACIÓN Y CONFIGURACIÓN DE APACHE WEB SERVER

Paquete: Apache Web Server para Linux

Sitio Web: http://www.apache.org/dist/httpd/apache_1.3.20.tar.gz

Nombre del archivo instalador: `apache_1.3.20.tar.gz`

Directorio de descarga del instalador: `/usr/src`

1. Descomprimir y desempaquetar el código fuente en el directorio **/usr/src**. Se debe ingresar al sistema como el superusuario **root**.

```
# su
```

```
# cd /usr/src
# gunzip apache_1.3.20.tar.gz
# tar -xvn apache_1.3.20.tar
# cd apache_1.3.20
# ./configure --prefix=/usr/local/apache
```

En este punto, antes de terminar la instalación de Apache, es necesario instalar PHP, ya que PHP será módulo de Apache. Luego de instalar y configurar PHP se continuará con la instalación y configuración de Apache.

INSTALACIÓN DE PHP COMO MÓDULO DE APACHE WEB SERVER

Paquete: PHP para Linux

Sitio Web: <http://www.php.net/get/php-4.3.0.tar/from/es.php.net/mirror>

Nombre del archivo instalador: php-4.3.0.tar.gz

Directorio de descarga del código fuente: /usr/src

1. Descomprimir el código fuente en el directorio **/usr/src**. Se debe ingresar como **root**.

```
# su
# cd /usr/src
# gunzip php-4.3.0.tar.gz
# tar -xvf php-4.3.0.tar
# cd ../php-4.3.0
# ./configure --with-pgsql=/usr/local/pgsql --with-apache=../apache_1.3.20 --enable-tvacks-vars
--enable-sysvsem --enable-sysvshm --enable-url-includes
```

En el script **configure** se indica que PHP será compilado con soporte para PostgreSQL y para Apache.

2. Compilar e Instalar PHP.

```
# make
# make install
```

En este paso, **make** se encarga de compilar PHP con los parámetros indicados en el script **configure** y **make install** realiza la instalación.

Una vez instalado PHP, a continuación retomamos la instalación de Apache, tomando PHP como módulo.

3. Activar PHP como módulo de Apache.

```
# cd ../apache_1.3.20
# ./configure --prefix=/usr/local/apache --activate-module=src/modules/php/libphp4.a
```

Con el parámetro **activate-module**, activamos a PHP como módulo de Apache.

4. Compilar e instalar Apache.

```
# make
# make install
```

En este paso, **make** se encarga de compilar Apache con los parámetros indicados en el script **configure** y **make install** realiza la instalación.

Antes de proceder a configurar Apache, debemos realizar un paso adicional con PHP: copiar el archivo **php.ini-dist** al directorio **/usr/local/lib** y renombrarlo como **php.ini**.

5. Copiar y renombrar el archivo **php.ini.dist**.

```
# cd ../php-4.3.0
# cp php.ini-dist /usr/local/lib/php.ini
# exit
```

El archivo **php.ini** contiene las variables de configuración de PHP.

Ya tenemos instalado Apache y PHP como módulo del mismo. Ahora hay que hacer algunos cambios en el archivo de configuración de Apache.

6. Editar el archivo de configuración de Apache: **httpd.conf** del directorio **/usr/local/apache/conf**. El archivo **httpd.conf** contiene las directivas de configuración que son las que dan instrucciones a Apache para su funcionamiento. Las principales directivas de configuración son: **ServerName**, **ServerRoot**, **AddType**, **Action**, **DirectoryIndex**, **DocumentRoot**. Estas directivas se encuentran en diferentes secciones; para que los cambios surtan efecto hay que descomentarlas (eliminar el **#**) y luego guardar el archivo y reiniciar el servidor.

```

# cd /usr/local/apache/conf
# vi httpd.conf

#ServerName localhost (nombre del dominio, de la máquina o dirección IP)
*****

#ServerRoot "/usr/local/apache"
*****

#AddType application/x-httpd-php .php php4 .phtml
#AddType application/x-httpd-php-source .phps
#Action application/x-httpd-php "/usr/local/php"
*****

#DirectoryIndex index.html index.htm index.shtml index.php index.php4 index.phtml
*****

#DocumentRoot "/var/www/html"
#<Directory "/var/www/html">

```

Luego de guardar los cambios del archivo **httpd.conf**, el servidor Apache está listo para ser iniciado.

7. Iniciar Apache. Este paso inicia el demonio **httpd**. Se debe ingresar como superusuario **root**.

```

# su
# /usr/local/apache/bin/httpd start

```

Para realizar cualquier cambio en la configuración de Apache, luego de que se ha iniciado la primera vez, se necesita primero **detener** el servidor, editar **httpd.conf** y luego **reiniciar** el servidor. Los comandos para detener y reiniciar el servidor son:

```

# su
# /usr/local/apache/bin/httpd stop
# /usr/local/apache/bin/httpd restart

```

Las directivas editadas en el paso **6** son básicas para que Apache sepa que hacer con un archivo cuya extensión sea: **.php**, **php4** o **phtml**. Sin embargo hay otras directivas que deben ser editadas para configurar completamente Apache. Todas las directivas de configuración de Apache junto con su definición se detallan en el **Anexo 4, Tabla 5**.

VERIFICACIÓN DE LAS INSTALACIONES

Para verificar que las instalaciones de PostgreSQL, Apache y PHP funcionan, creamos una página web con el nombre **index.php** en el directorio de publicación de Apache. Esta página contendrá un script de conexión a una base de datos de nombre **prueba** con una tabla **usuarios**.

Los mismos pasos deben ser seguidos si se instala vía RPM o por medio de binarios, pero se debe tomar en cuenta que las rutas no son las mismas. En ambos casos el directorio de publicación de Apache es: **/var/www/html**.

Nota: Tomar en cuenta que debe estar iniciado el servidor web Apache y la base de datos PostgreSQL.

1. Crear la base de datos y tablas en PostgreSQL. Una vez creada la base de datos, se debe crear el usuario **apache** y darle privilegios para trabajar sobre las tablas.

```
# su postgres
# postmaster -i -D /usr/local/pgsql/data
# su postgres (en otra consola)
# CREATEDB prueba;
# psql prueba
prueba => CREATE TABLE usuarios {
           nombre varchar(10),
           clave varchar(10),
           tipo_usuario varchar(10)
           };
prueba => \q
#createuser apache;
# psql prueba
prueba => GRANT SELECT ON usuarios TO apache;
prueba => \q
```

2. Crear la página **index.php**. Esta página desplegará los datos(excepto las contraseñas) ingresados en la tabla **usuarios**. La página debe ser creada dentro del directorio de publicación de Apache.

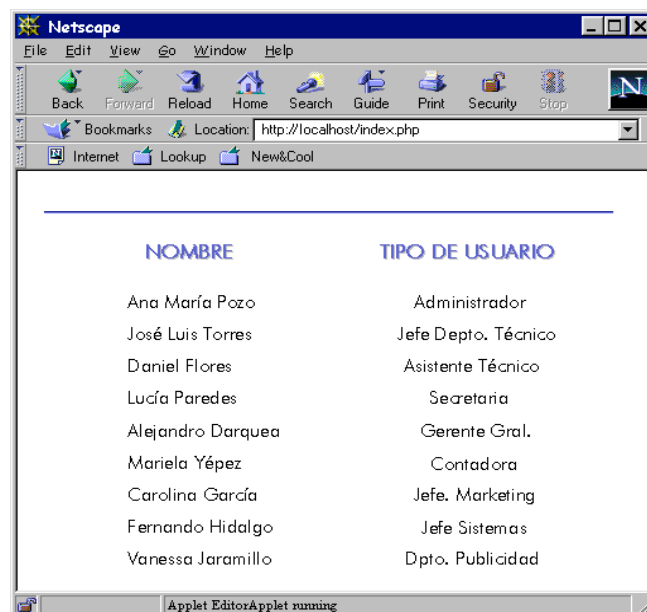
```
<HTML>
  <HEAD>
    <TITLE>Página de prueba</TITLE>
  </HEAD>
```

```

<BODY>
<?php  /***** Conexión con PostgreSQL *****/
$conexion = pg_pconnect("host=localhost, port=5432, dbname=prueba");
if (!$conexion) {
    echo "<CENTER>Problemas de conexión con la base de datos.</CENTER>";
    exit;
}
$sql="SELECT nombre, tipo_usuario FROM usuarios; ";
$resultado_set = pg_Exec ($conexion, $sql);
$filas = pg_NumRows($resultado_set);
echo "NOMBRE."          ". "TIPO DE USUARIO";
for ($j=0; $j < $filas; $j++){
    echo pg_result($resultado_set, $j, 0)."          ".pg_result($resultado_set, $j, 1);
    echo "<BR>";
}
pg_close($conexion); /***** Cierre de la conexión con PostgreSQL *****/
?>
</BODY>
</HTML>

```

3. Desplegar la página **index.php** en el navegador. Para desplegar la página **index.php**, ir al navegador y escribir la siguiente dirección: **http://localhost/index.php**. Donde **localhost** es el nombre de la máquina, dirección **IP** o dominio. Si todo ha salido bien, el resultado puede verse en la figura siguiente.



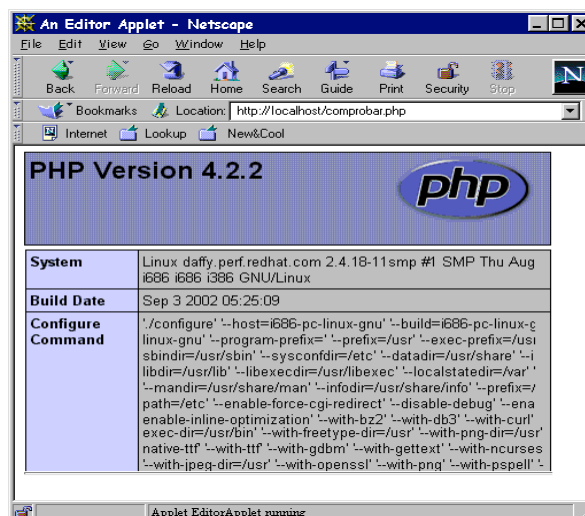
También se puede comprobar las instalaciones de Apache y PHP individualmente. Para comprobar que Apache ha sido instalado y configurado correctamente, se debe iniciar Apache, abrir un navegador y escribir la siguiente dirección: **http://localhost**, si se despliega la siguiente página Apache está listo para trabajar.



Para comprobar que PHP ha sido instalado y configurado correctamente se debe crear una página web con extensión **.php** (comprobar.php) que contenga el siguiente código.

```
<HTML> <HEAD><TITLE>Comprobación de PHP</TITLE> </HEAD>
<BODY> <? phpinfo(); ?> </BODY>
</HTML>
```

Dicha página debe ser guardada en el directorio de publicación, y luego en un navegador se debe escribir la dirección: **http://localhost/comprobar.php**, si aparece una página similar a la siguiente, PHP está listo para trabajar.



INSTALACIÓN, CONFIGURACIÓN Y USO DE PHP EN WINDOWS

Las instalaciones se deben hacer en el siguiente orden:

1. Instalación y configuración de la base de datos: PostgreSQL
2. Instalación y configuración del servidor web: Apache Web Server
3. Instalación y configuración de PHP

INSTALACIÓN Y CONFIGURACIÓN DE POSTGRESQL

Para la instalación del servidor de base de datos PostgreSQL en Windows es necesario instalar la utilidad **Cygwin**. La versión de PostgreSQL que se instala bajo **Cygwin** es la 7.3.3.

🖱 **Nota:** *Cygwin es una "capa de emulación que expone suficiente funcionalidad del API de Unix", en otras palabras es una API compuesta de una DLL(cygwin1.dll) y varias herramientas que traen la mayoría de las funciones de Unix para poder correr programas de Unix en Windows mediante una consola tipo MS-DOS.*

INSTALACIÓN DE CYGWIN

Paquete: Cygwin - API de Unix para Windows.

Sitio Web: <http://cygwin.com/setup.exe>

Nombre del archivo instalador: setup.exe

Directorio de descarga del archivo instalador: C:\Downloads\

1. Crear en el Panel de control el usuario **postgres**, sin contraseña. La instalación de **Cygwin** se debe hacer iniciando la sesión como dicho usuario.
2. Descargar el instalador de la utilidad **Cygwin**. Ir al sitio web indicado y pulsar en el icono de descarga. Guardar el archivo instalador **setup.exe** en el directorio C:\Downloads\.
3. Ejecutar el instalador **setup.exe** para la descarga de **Cygwin** y los paquetes necesarios. En este caso, se descargará el paquete **postgresql**. Para realizar la descarga aparecerá un wizard con las siguientes pantallas:

Pantalla de inicio - Seleccionar Siguiente.

Choose a Download Source - Seleccionar Download from Internet (opción recomendada), esta opción permite copiar los instaladores a un directorio local para instalar posteriormente.

Select Local Package Directory - C:\Cygwin\Download\. Es el directorio donde se almacenarán los archivos de los instaladores. Si el directorio no existe el wizard lo creará.

Select Your Internet Connection - Seleccionar Direct Connection. Establece la forma de conexión a Internet.

Choose A Download Site - Permite escoger un sitio (mirror) de una lista de sitios que contienen el instalador **Cygwin** para descargar.

Select Packages - Permite seleccionar los paquetes a descargar, en este caso, seleccionar la sección **Database** y luego el paquete **postgresql**.

👉 **Nota:** Si en el paso *Choose a Download Source* se selecciona *Install from Internet*, la instalación se llevará a cabo automáticamente después de descargar los paquetes seleccionados. Caso contrario esperamos a que se termine de descargar y seguimos con el paso 4.

En este punto, esperamos a que se termine de descargar todo, luego saldrá un mensaje indicando que la descarga se ha completado.

4. Ejecutar nuevamente el instalador **setup.exe**, esta vez para instalar los paquetes descargados. Se abrirá el mismo wizard, pero en esta ocasión se debe seguir los siguientes pasos:

Pantalla de inicio - Seleccionar **Siguiente**.

Choose a Download Source - Seleccionar **Install From Local Directory**. Esta opción debe ser escogida porque los instaladores están almacenados localmente.

Select Root Install Directory - C:\Cygwin. Es el directorio en el que se quiere instalar **Cygwin**.

Select Local Package Directory - C:\Cygwin\Download\. Seleccionar el directorio donde se copiaron anteriormente los archivos instaladores.

Select Packages - Permite seleccionar los paquetes a instalar, en este caso, seleccionar la sección **Database** y luego **postgresql**.

En este punto se debe esperar hasta que salga el mensaje de instalación completa.

5. Indicar que se creen los accesos directos en el escritorio y en el menú Inicio. En este punto **Cygwin** está instalado, se puede abrir el shell de Unix desde el siguiente icono:



Se abrirá una consola de **Cygwin** tipo MS-DOS, en la que, dependiendo de los paquetes instalados, se podrá trabajar como si se estuviese en entornos Unix.

👉 **Nota:** Adicionalmente, luego de instalar *Cygwin*, se debe instalar *CyglPC*. *CyglPC* es una utilidad requerida para correr PostgreSQL.

INSTALACIÓN DE CYGIPC

Paquete: CyglPC.

Sitio Web: <http://www.neuro.gatech.edu/users/cwilson/cygutils/V1.1/cygipc/>

Nombre del archivo instalador: `cygipc-1.14-1.tar.bz2`

Directorio de descarga del instalador: `C:\Downloads\`

1. Descargar el archivo binario **cygipc-1.14-1.tar.bz2** del sitio web indicado y guardarlo en el directorio: `C:\Cygwin`, éste directorio es donde se instaló anteriormente **Cygwin**.
2. Como usuario **postgres**, abrir una consola de **Cygwin** y comprobar que el prontuario sea: **postgres@nombre_pc**.
3. Descomprimir el archivo binario **cygipc-1.14-1.tar.bz2**. Ir al directorio raíz de **Cygwin** y ejecutar el comando **tar**, las siguientes líneas indican la secuencia:

```
postgres@nombre_pc$ cd /
postgres@nombre_pc$ tar -xvf cygipc-1.14-1.tar.bz2
```

Este paso descomprimirá el archivo **cygipc-1.14-1.tar.bz2** y creará varios directorios y archivos, entre esos archivos se encuentra **ipc-daemon.exe** que es necesario para el funcionamiento de

PostgreSQL en Windows.

4. Ejecutar **ipc-daemon.exe**. Ir al directorio **C:\Cygwin\usr\local\bin**.

```
postgres@nombre_pc $ cd C:/Cygwin/usr/local/bin
postgres@nombre_pc $ ipc-daemon.exe
```

5. Abrir otra consola de **Cygwin** e iniciar la base de datos PostgreSQL y el servidor SQL.

```
postgres@nombre_pc $ initdb -D /usr/share/postgresql/data
postgres@nombre_pc $ postmaster -i -D /usr/share/postgresql/data
```

El comando **initdb** se ejecuta sólo la primera vez, luego de la instalación; el comando **postmaster** se ejecuta cada vez que se necesite trabajar con PostgreSQL, antes de acceder a las bases de datos con el cliente **psql**.

6. Editar el archivo **pg_hba.conf**. En este archivo debe constar el nombre de máquina, dirección IP, máscara y método de encriptación tanto de la máquina servidor como de las máquinas cliente que se van a conectar a PostgreSQL. Por ejemplo:

#	TYPE	DATABASE	USER	IP-ADDRESS	IP-MASK	METHOD
	localhost	all	all	127.0.0.1	255.255.255.255	trust

7. Crear una base de datos. En este punto ya se puede crear bases de datos con sus respectivas tablas. Para acceder a una base de datos, se utiliza el cliente **psql** de PostgreSQL. Mediante el cliente **psql** podemos ejecutar cualquier instrucción SQL.

```
postgres@nombre_pc $ createdb prueba
postgres@nombre_pc $ psql prueba
$prueba => select * from prueba;
```

INSTALACIÓN Y CONFIGURACIÓN DE APACHE WEB SERVER

Paquete: Apache Web Server para Windows

Sitio Web: <http://www.apache.org/dist/httpd/apache-1.3.12-win32-src.zip>

Nombre del archivo instalador: apache-1.3.12-win32-src.zip

Directorio de descarga del instalador: C:\Downloads\Apache\

1. Extraer **apache-1.3.12-win32-src.zip** en el directorio **C:\Downloads\Apache**. Este paso creará el archivo ejecutable **apache-1.3.12-win32** dentro de dicho directorio.
2. Ejecutar **apache-1.3.12-win32**. En este paso, luego de aceptar los términos de la licencia, se desplegará una pantalla en la que se debe ingresar los siguientes datos:

Nombre de dominio: nombre_del_dominio.com

Nombre del servidor: www. nombre_del_dominio.com

E-mail del administrador: cuenta_admin@ nombre_del_dominio.com

En la misma pantalla debemos elegir: **Correr Apache Web Server como un servicio para todos los usuarios**. Esta elección permite que Apache se inicie cuando se inicia la máquina y sigue en funcionamiento aún cuando una sesión termina.

3. Elegir tipo de instalación. Dependiendo del instalador se elegirá Instalación **Típica** o **Completa**, no **Personalizada**.
4. Elegir el directorio de instalación. El directorio de instalación de Apache por defecto es: **C:\Archivos de Programa\Apache Group**. De preferencia no cambiar para que no existan conflictos o errores posteriores.
5. Instalar Apache. En este paso se puede ver una barra de progreso de la instalación, así como unas pantallas MS-DOS, lo cual es normal, además el proceso de instalación preguntará si se desea sobrescribir algunos archivos, a lo que respondemos **NO**.

En este punto ya podemos realizar la instalación y configuración de PHP, para luego configurar Apache con soporte a PHP y posteriormente arrancar el servidor de forma adecuada.

INSTALACIÓN Y CONFIGURACIÓN DE PHP

Paquete: PHP para Windows

Sitio Web: <http://www.php.net/get/php-4.0.4-Win32.zip/from/es.php.net/mirror>

Nombre del archivo instalador: php-4.0.4-Win32.zip

Directorio de descarga del instalador: C:\Downloads\

1. Extraer **php-4.0.4-Win32.zip** en el directorio **C:\php**. Este paso extraerá varios archivos y carpetas, entre esos archivos se encuentra el archivo **php4.ini-dist**.
2. Copiar el archivo **php4.ini-dist** al directorio **C:\windows** o donde se encuentren los archivos de extensión **.ini**. Una vez copiado el archivo, dentro del directorio, lo renombramos a **php4.ini**.
3. Editar el archivo **php4.ini**. En este archivo es donde se controlan las variables de configuración de PHP. Aquellas variables que debemos editar son: **extension_dir** y **doc_root** de la sección "**Paths and Directories**"; y en la sección "**Dynamic Extensions**"; debemos activar las librerías que necesitemos. Al editar el archivo, la configuración debe quedar como sigue:

```
*****Paths and Directories*****
;extension_dir=C:/php/extensions
;doc_root= C:/Archivos de Programa/Apache Group/Apache/htdocs

*****Dinamic Extensions*****
;extension=php_pgsql.dll
;extension=php3_calendar.dll
;extension=php3_dbase.dll
;extension=php3_gd.dll
;extension=php3_dbm.dll
;extension=php3_zlib.dll
;extension=php3_filepro.dll
;extension=php3_imap4r1.dll
;extension=php3_ldap.dll
;extension=php3_crypt.dll
;extension=php3_odbc.dll
```

La variable **extension_dir** debe contener el directorio donde se encuentran las librerías dinámicas(archivos **.dll**). La variable **doc_root** debe contener el directorio raíz donde se publicarán las páginas web. Para que la edición tenga efecto quitamos el (;) del principio de las líneas editadas.

En la sección **Dinamic Extensions** debemos quitar el (;) de aquellas extensiones que necesitemos que estén activas en PHP. Por ejemplo, la extensión **php_pgsql.dll** debe estar activada para que exista conexión entre PHP y PostgreSQL. Luego grabamos el archivo para que los cambios surtan efecto.

👉 **Nota:** Todos los “\” deben ser cambiados por “/” en **php4.ini** y en **httpd.conf** debido a que serán archivos utilizados por Apache. El servidor web Apache trabaja con la sintaxis de UNIX.

4. Copiar los archivos **msvcrt.dll** y **php4ts.dll** ubicados en el directorio **C:\php\dlls** al directorio **C:\windows\system**. En **C:\windows\system** se encuentran todos los archivos **.dll** del sistema. Si el sistema indica que ya existe el archivo **msvcrt.dll**, no debemos sobrescribirlo con el nuevo (dejamos el archivo anterior).

En este punto la instalación de PHP está lista, ahora debemos proceder con la configuración de Apache.

5. Configurar Apache. En este paso podemos editar las principales directivas de configuración antes de iniciar Apache. Para ello vamos al directorio **C:\Archivos de Programa\Apache Group\Apache\conf** y abrimos el archivo **httpd.conf** con el Bloc de Notas. Editamos lo siguiente: **Servername, ServerRoot, AddType, Action, DirectoryIndex, DocumentRoot**.

```
#ServerName localhost (nombre del domino, de la máquina o dirección IP)
*****

#ServerRoot C:/Archivos de Programa/Apache Group/Apache
*****

#ScriptAlias /php/ "C:/php/"
#AddType application/x-httpd-php .php .php4 .phtml
#AddType application/x-httpd-php-source .phps
#Action application/x-httpd-php "/php/php.exe"
*****

#DirectoryIndex index.html index.htm index.shtml index.php index.php4 index.phtml
*****

#DocumentRoot "C:/Archivos de Programa/Apache Group/Apache/htdocs"
#<Directory "C:/Archivos de Programa/Apache Group/Apache/htdocs">
```

Descomentamos eliminando el **#**, luego grabamos para que los cambios surtan efecto. Apache en Windows se configura igual que en Linux, las directivas editadas en el paso **5** son básicas para que Apache sepa que hacer con un archivo cuya extensión sea: **.php**, **php4** o **phtml**. Sin embargo hay otras directivas que deben ser editadas para configurar completamente Apache. Todas las directivas de configuración de Apache junto con su definición se detallan en el **Anexo 4, Tabla 5**.

👁 **Nota:** En el **Anexo 4, Tabla 5** del se dan ejemplos de configuración para Linux. Para la configuración de **http.conf** en Windows, se debe tomar en cuenta que todos los "\" deben ser cambiados a "/".

6. Iniciar el servidor Apache. Luego de editar el **httpd.conf**, el servidor está listo para ser iniciado. Ir a **Inicio – Programas – Apache Web Server – Apache Start**. Con este paso el servidor estará iniciadg. Para detener el servidor, lo hacemos mediante la opción **Inicio – Programas – Apache Web Server – Stop Apache** g **apache -k shutdown**(en una consola DOS).

VERIFICACIÓN DE LAS INSTALACIONES

Para verificar que las instalaciones de PostgreSQL, Apache y PHP funcionan, creamos una página web con el nombre **index.php** en el directorio **C:\Archivos de Programa\Apache Group\Apache\htdocs**. Esta página contendrá un script de conexión a una base de datos de nombre **prueba** con una tabla **usuarios**.

👁 **Nota:** Tomar en cuenta que debe estar iniciado el servidor web Apache y la base de datos PostgreSQL.

4. Crear la base de datos y tablas en PostgreSQL. Una vez creada la base de datos, se deba crear el usuario **apache** y darle privilegios para trabajar sobre las tablas.

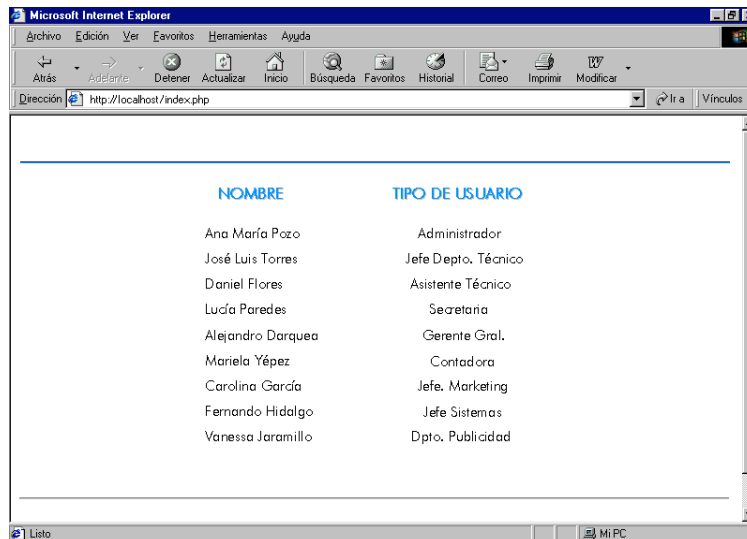
```
postgres@nombre_pc$ postmaster -i -D /usr/share/postgresql/data
postgres@nombre_pc $ CREATEDB prueba; (en otra consola)
postgres@nombre_pc $ psql prueba
prueba => CREATE TABLE usuarios {
           nombre varchar(10),
           clave varchar(10),
           tipo_usuario varchar(10)
           };
prueba => \q
#createuser apache;
# psql prueba
prueba => GRANT SELECT ON usuarios TO apache;
```

5. Crear la página **index.php**. Esta página desplegará los datos(excepto las contraseñas) ingresados

en la tabla **usuarios**. La página debe ser creada en el directorio **C:\Archivos de Programa\Apache Group\Apache\htdocs**.

```
<HTML>
<HEAD>
  <TITLE>Página de prueba</TITLE>
</HEAD>
<BODY>
<?php  /***** Conexión con PostgreSQL *****/
$conexion = pg_pconnect("host=localhost, port=5432, dbname=prueba");
if !$conexion) {
  echo "<CENTER>Problemas de conexión con la base de datos.</CENTER>";
  exit;
}
$sql="SELECT nombre, tipo_usuario FROM usuarios; ";
$resultado_set = pg_Exec ($conexion, $sql);
$filas = pg_NumRows($resultado_set);
echo "NOMBRE."          ". "TIPO DE USUARIO";
for ($j=0; $j < $filas; $j++){
  echo pg_result($resultado_set, $j, 0)."          ".pg_result($resultado_set, $j, 1);
  echo "<BR>";
}
pg_close($conexion);  /***** Cierre de la conexión con PostgreSQL *****/
?>
</BODY>
</HTML>
```

6. Desplegar la página **index.php** en el navegador. Para desplegar la página **index.php**, ir al navegador y escribir la siguiente dirección: **http://localhost/index.php**. Donde **localhost** es el nombre de la máquina, dirección **IP** o dominio.



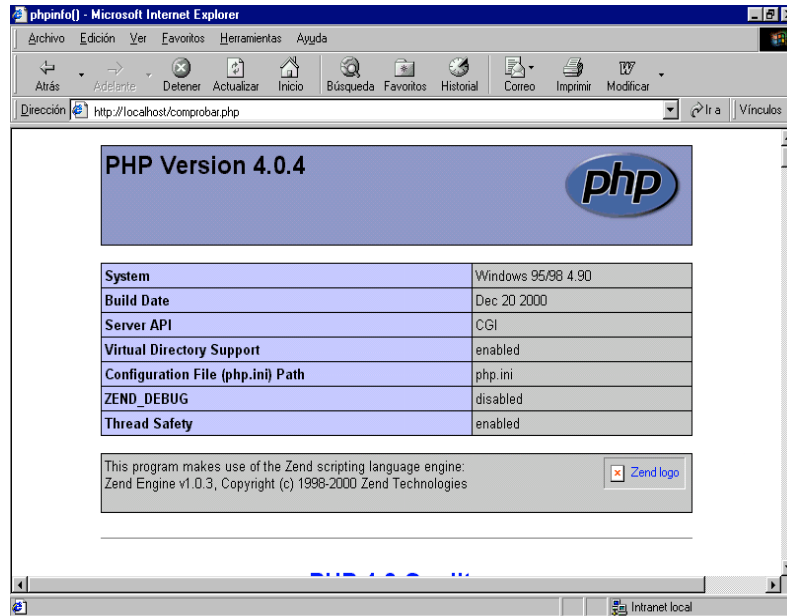
También se puede comprobar las instalaciones de Apache y PHP individualmente. Para comprobar que Apache ha sido instalado y configurado correctamente, se debe iniciar Apache, abrir un navegador y escribir la siguiente dirección: **http://localhost**, si se despliega la siguiente página Apache está listo para trabajar.



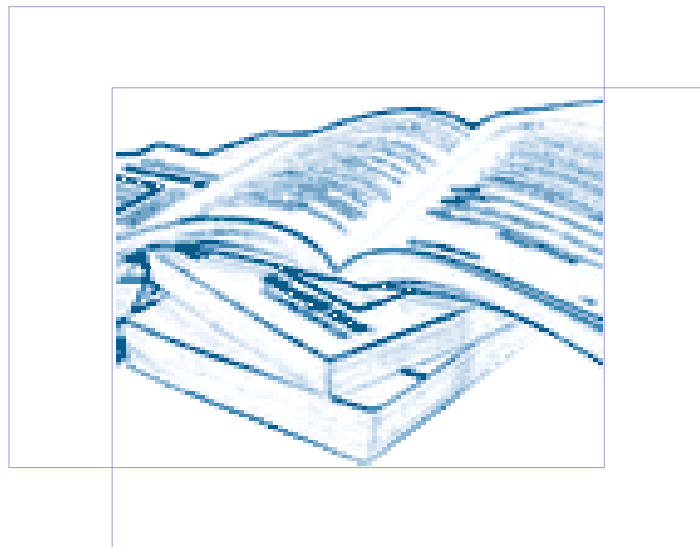
Para comprobar que PHP ha sido instalado y configurado correctamente se debe crear una página web con extensión **.php** (comprobar.php) que contenga el siguiente código.

```
<HTML> <HEAD><TITLE>Comprobación de PHP</TITLE> </HEAD>
<BODY> <? phpinfo(); ?> </BODY>
</HTML>
```

Dicha página debe ser guardada en el directorio de publicación, y luego en un navegador se debe escribir la dirección: **http://localhost/comprobar.php**, si aparece una página similar a la siguiente, PHP está listo para trabajar.



ANEXO 2



MANUALES DE INSTALACIÓN, CONFIGURACIÓN Y USO DE ASP EN LINUX Y EN WINDOWS

En este Anexo, se detalla el proceso de instalación, configuración y uso de ASP y de los paquetes complementarios para que ASP funcione adecuadamente.

INSTALACIÓN, CONFIGURACIÓN Y USO DE ASP EN LINUX

Para instalar ASP en Linux, se necesita del software **Chili!Soft ASP** que permite integrar ASP con Apache Web Server. Cabe destacar que Chili!Soft ASP es un software comercial, aunque también cuenta con una versión gratuita y una para desarrollo, pero la primera es temporal y la segunda tiene características limitadas. Por lo tanto si se necesita instalar ASP para un sitio real en Internet se debe adquirir la versión comercial.

En este Anexo se detalla la instalación de Chili!Soft ASP en la versión gratuita, la única diferencia en el proceso de instalación entre la versión gratuita y la versión comercial es la licencia de tipo evaluación por tiempo limitado.

El orden a seguir se indica a continuación:

1. Instalación de Chili!Soft ASP
2. Instalación y configuración de la base de datos: PostgreSQL
3. Configuración del controlador ODBC para PostgreSQL

Al instalar Chili!Soft ASP, se instala Apache Web Server, el servidor de ASP, se configura el archivo **httpd.conf** y los controladores ODBC para las bases de datos soportadas, todo lo anterior por defecto, si no se tiene otro servidor web instalado.

Nota: Para la siguiente instalación, se recomienda Red Hat Linux 7.2, y Apache Web Server 1.3.19 o 1.3.22 con soporte para módulos **DSO**(Dynamic Shared Object), ya que son las versiones que han sido certificadas por Sun Chili!Soft ASP.

INSTALACIÓN DE CHILI!SOFT ASP

Paquete: Chili!Soft ASP 3.6. 2 para Linux

Sitio Web: <http://www.chilisoft.com/downloads>

Nombre del archivo instalador: `casp-3.6.2-linux.tar`

Directorio de descarga del instalador: `/usr/src`

1. Ingresar al sistema como **root**.
2. Desempaquetar el archivo instalador **casp-3.6.2-linux.tar** que se encuentra en el directorio **/usr/src**.

```
$ cd /usr/src
$ tar -xvf casp-3.6.2-linux.tar
```

Este paso creará varios archivos, entre ellos el script **install.sh** necesario para la instalación.

3. Ejecutar el script **install.sh**.

```
$ ./install.sh
```

Al ejecutar este comando se desplegará un wizard para guiar la instalación. Las pantallas que aparecerán son las siguientes:

Aceptar los términos de la licencia - En este caso la licencia será de evaluación por tiempo limitado. Teclar **"yes"** para aceptarla.

Escoger el directorio de instalación - Por defecto el directorio es **/opt/casp**, aunque puede ser cambiado.

Ingresar el número de licencia - El wizard preguntará por el número de licencia, si lo conocemos tecleamos **"y"** caso contrario **"n"**. En nuestro caso telceamos **"n"** y obtendremos una licencia de evaluación.

Configurar Apache Web Server por defecto – Si todavía no se tiene instalado un servidor web, este paso permite indicar si más adelante se desea que Chili!Soft ASP instale y configure Apache Web Server. Si se instala, Chili!Soft ASP configurará el archivo **httpd.conf** durante la instalación. Apache Web Server 1.3.19 DSO o 1.3.22 DSO son las únicas versiones del servidor que pueden

ser configuradas por Chili!Soft ASP. En nuestro caso para aceptarlo tecleamos "y".

Escoger el tipo de configuración para Apache Web Server - Se puede seleccionar el tipo de configuración para el httpd.conf, las opciones son: teclear 1 para **Default configuration**, 2 para **Specify only the Web server listen port** y 3 para **Customize the configuration**. En este caso tecleamos 1 para obtener una configuración básica y funcional. Este paso se realiza si se aceptó el anterior.

Seleccionar el lenguaje - El lenguaje por defecto es el inglés, para aceptarlo presionamos "Enter", caso contrario tecleamos un número de los lenguajes que se presentan en la lista.

Habilitar Chili!Beans - Permite habilitar o no el soporte para en entorno **JRE** y **JavaBeans**. Seleccionamos la opción: **Disable Java support**, debido a que sólo nos intereza desarrollar en ASP.

Seleccionar Servidor Web - Permite seleccionar un servidor web de la lista o detecta si existe un servidor web instalado. En nuestro caso seleccionamos Apache Web Server, ya que es el servidor que necesitamos, además anteriormente indicamos a Chili!Soft que sea configurado por defecto.

Indicar configuración por defecto - Digitar 1 para indicar que Apache sea configurado con Chili!Soft ASP por defecto.

Verificar Información acerca del servidor web - Digitar "y" para indicar que la información que muestra Chili!Soft ASP acerca del servidor web es correcta, caso contrario digitar "n".

Seleccionar tipo de configuración para el servidor de ASP - Las opciones son: digitar 1 para **Default Configuration**, 2 para **Custom Configuration**. Para obtener una configuración básica digitamos 1.

Seleccionar tipo de configuración para consola de administración ASP - Debido a que en el paso anterior digitamos 1, en este paso también lo hacemos para que Chili!Soft ASP configure por defecto la consola de administración de ASP. Con esta opción se establecerá el nombre de usuario "admin" y la clave "root". Para proteger el servidor de ASP, es recomendable que se cambie el usuario y la clave desde la consola de administración tan pronto como sea posible.

Información sobre configuración de consola de administración ASP - Para aceptar la información sobre como ha sido configurada la consola de administración de ASP, se debe presionar **“Enter”**, para que la instalación continúe. Al terminar, el wizard escribe un resumen de la instalación en el archivo `install_summary` dentro del directorio: `/opt/casp/logs`.

4. Iniciar el servidor de ASP. Para iniciar el servidor de ASP, se debe hacer uso de la consola de administración de ASP. Para ingresar a la consola de administración se utiliza la URL:

```
http://localhost:port
```

Donde **localhost** es el **ServerName** establecido en `httpd.conf`, y **port** es el puerto configurado por la consola de administración, por defecto es **5100**.

Dentro de la consola podemos hacer nuestras propias configuraciones o aceptar las que realizó por defecto el wizard de instalación. Una vez aceptadas las configuraciones, podemos verificar que Apache en conjunto con ASP estén funcionando correctamente, para ello utilizamos la siguiente URL:

```
http://localhost/caspsamp/diagnostics.htm
```

INSTALACIÓN Y CONFIGURACIÓN DE POSTGRESQL

Paquete: Servidor de base de datos PostgreSQL para Linux

Sitio Web: <ftp://ftp4.es.postgresql.org/Mirror/www.postgresql.org/v7.1.3/postgresql-7.1.3.tar.gz>

Nombre del archivo instalador: `postgresql-7.1.3.tar.gz`

Directorio de descarga del instalador: `/usr/src`

Ver la sección [INSTALACIÓN Y CONFIGURACIÓN DE POSTGRESQL](#). Todos los pasos deben ser seguidos, en el paso **3** se compila con el módulo `--with-odbc`, dicho módulo servirá para conectarnos a PostgreSQL mediante un controlador **ODBC** y permitirá que las páginas ASP trabajen con la base de datos utilizando **ADO**(ActiveX Data Object).

CONFIGURACIÓN DEL CONTROLADOR ODBC PARA POSTGRESQL


Paquete: Controlador ODBC para PostgreSQL en Linux

Nombre del archivo instalador: libodbcpsql.so

Directorio del instalador: /opt/casp/odbc/opensource/lib

Durante la instalación de Chili!Soft ASP, se instalan automáticamente los controladores ODBC para las bases de datos soportadas, entre éstos controladores se encuentra el controlador para PostgreSQL 6.5.2 y superiores. El controlador ODBC para PostgreSQL es: **libodbcpsql.so** y se encuentra en: **/opt/casp/odbc/opensource/lib**.

Para que PostgreSQL funcione correctamente hace falta configurar el controlador respectivo, agregando un **DSN**. Antes de crear el DSN, se debe crear primero la base de datos con sus respectivas tablas.

 **Nota:** En la sección [VERIFICACIÓN DE LAS INSTALACIONES](#), en el paso 1, se indica como crear una base de datos en PostgreSQL.

1. Agregar un **DSN** de sistema. Luego de creada la base de datos con sus tablas, ir a la consola de Administración del servidor de ASP(<http://localhost:5100>), en la sección **server management** escoger el enlace **Databases** y **Add New DSN**. Se desplegará un formulario en el que se deberá llenar los siguientes datos:

DSN Name: nombre del DSN, ej: postgresql_asp

Description: descripción del DSN, ej: Conexión a PostgreSQL mediante ASP

DataBase Type: seleccionar PostgreSQL

Para que el DSN quede configurado se debe grabar.

Una vez que se ha creado el DNS, Chili!Soft ASP configura automáticamente los parámetros para el controlador ODBC. Entre los parámetros que se configuran están:

Data Source Name: postgresql_asp

DataBase Type: PostgreSQL

Controlador: libodbcpsql.so

ServerName: IP del servidor de Base de Datos, si está en blanco se asumirá que PostgreSQL está instalado localmente.

Port: 5432, es el puerto por defecto para PostgreSQL.

DataBase: es el nombre DSN

User: nombre de usuario requerido para conectarse a la base de datos

Password: clave requerida para conectarse a la base de datos.

2. Probar el DSN. En la sección **Databases**, en la pestaña **Data Source Names**, se listan los DSN creados, de éstos seleccionar el que hemos creado anteriormente y presionar el enlace **test** que se encuentra en frente.

Si todo está correcto se desplegará una pequeña pantalla con un mensaje de creación satisfactoria del DSN.

3. Configurar número de conexiones mediante **ADO**. Chili!Soft ASP utiliza el control ADO(ActiveX Data Object) para conectarse a bases de datos desde las páginas ASP. En la sección **Databases**, en la pestaña **ADO Settings**, ingresar el número(25 por defecto) de conexiones que se desea que estén disponibles en el campo **Connection pool size**. Para que los cambios surtan efecto se debe grabar y reiniciar el servidor de ASP.

En este punto ya se puede crear una cadena de conexión para ser usada dentro de las páginas ASP. Existen 3 formas para crear una cadena de conexión, mediante: DSN de sistema, DSN de archivo y DSN de usuario. Debido a que en el paso 1 creamos un DSN de sistema, es éste el que usaremos para crear la cadena de conexión, por ejemplo:

```
cadena_conexion= "DATABASE=prueba; UID=adminbdd; PWD=; DSN=postgresql_osp"
```

VERIFICACIÓN DE LAS INSTALACIONES

Para verificar que las instalaciones de PostgreSQL, Apache y ASP funcionan, creamos una página web con el nombre **index.php** en el directorio de publicación de Apache. Esta página contendrá un script de conexión a una base de datos de nombre **prueba** con una tabla **usuarios**.

El directorio de publicación de Apache en Linux es: **/var/www/html**

Nota: Tomar en cuenta que debe estar iniciado el servidor web Apache y la base de datos PostgreSQL.

1. Crear la base de datos y tablas en PostgreSQL. Una vez creada la base de datos, se debe crear el usuario **apache** y darle privilegios para trabajar sobre las tablas.

```
# su postgres
# postmaster -i -D /usr/local/pgsql/data
# su postgres (en otra consola)
# CREATEDB prueba;
# psql prueba
prueba => CREATE TABLE usuarios {
           nombre varchar(10),
           clave varchar(10),
           tipo_usuario varchar(10)
        };
prueba => \q
#createuser apache;
# psql prueba
prueba => GRANT SELECT ON usuarios TO apache;
prueba => \q
```

2. Agregar un **DSN** de sistema para la base de datos **prueba**. Ir a la consola de Administración del servidor de ASP(<http://localhost:5100>), en la sección **server management** escoger el enlace **Databases** y **Add New DSN**. Llenar los siguientes datos:

DSN Name: conexion_postgresql

Description: Conexión a PostgreSQL mediante ASP

DataBase Type: PostgreSQL

3. Crear la página **index.asp**. Esta página desplegará los datos(excepto las contraseñas) ingresados en la tabla **usuarios**. La página debe ser creada dentro del directorio de publicación de Apache.

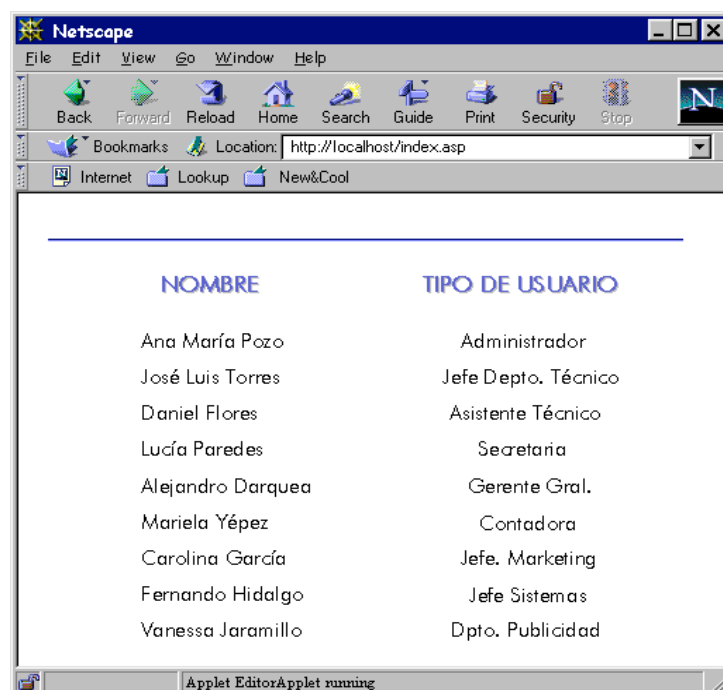
```
<HTML>
  <%@ LANGUAGE="VBSCRIPT" %>
  <HEAD>
    <TITLE>Página de prueba</TITLE>
  </HEAD>
  <BODY>
    /***** Conexión con PostgreSQL *****/
  <%
    Dim conexion, resultado, cadena_conexion
```

```

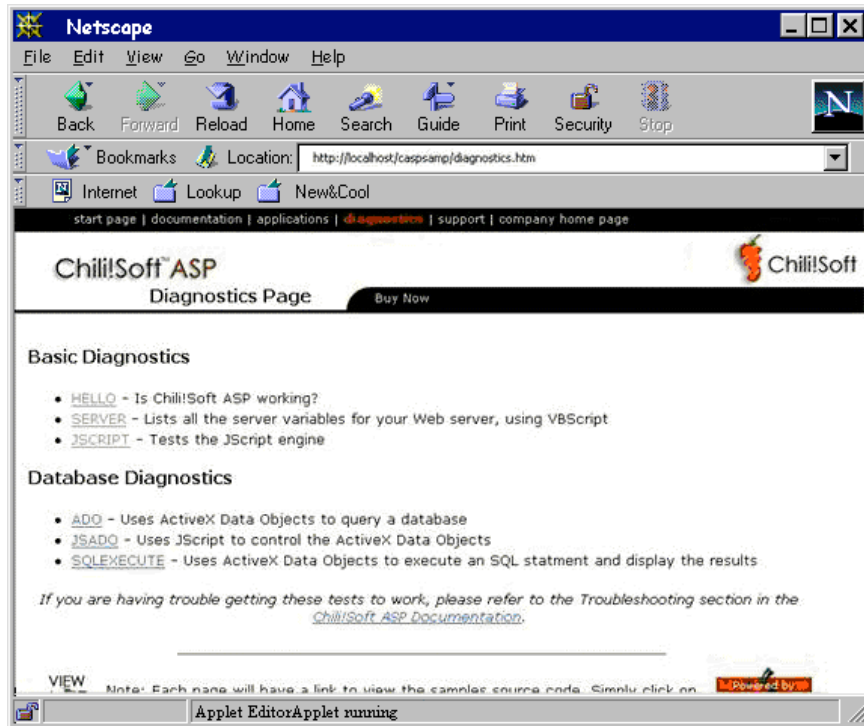
cadena_conexion= "DATABASE=prueba; UID=adminbdd; PWD=; DSN=postgresql_esp"
Set conexion = Server.CreateObject ("ADODB.Connection")
Set resultado = Server.CreateObject ("ADODB.RecordSet")
conexion.Open cadena_conexion
sql = "SELECT usuario, tipo_usuario FROM usuarios"
resultado.Open sql, conexion
Do While not resultado.EOF
    Response.Write "NOMBRE" & " " & "TIPO DE USUARIO"
    Response.Write resultado ("usuario") & " " & resultado ("tipo_usuario ")
    Response.Write "<BR>"
    resultado.MoveNext
Loop
resultado.Close
conexion.Close
Set resultado = Nothing
Set conexion = Nothing
%>
</BODY>
</HTML>

```

4. Desplegar la página **index.asp** en el navegador. Para desplegar la página **index.asp**, ir al navegador y escribir la siguiente dirección: **http://localhost/index.asp**. Donde **localhost** es el nombre de la máquina, dirección IP o dominio. Si todo ha salido bien, el resultado puede verse en la figura siguiente.



Para verificar que Apache en conjunto con ASP estén funcionando correctamente, escribimos utilizamos la siguiente URL: <http://localhost/caspsamp/diagnostics.htm> en un navegador, si apareció la siguiente página, el servidor de ASP estará listo para trabajar.



INSTALACIÓN, CONFIGURACIÓN Y USO DE ASP EN WINDOWS

En este proceso se debe seguir un orden específico, el cual se indica a continuación:

1. Instalación y configuración de la base de datos: PostgreSQL
2. Instalación y configuración del servidor web: Internet Information Server
3. Instalación y configuración de ASP

INSTALACIÓN Y CONFIGURACIÓN DE POSTGRESQL

Para la instalación de PostgreSQL en Windows se deben instalar las utilidades: **Cygwin** y **CyglPC**. La instalación de dichas utilidades se encuentra detallada en las secciones: [INSTALACIÓN DE CYGWIN](#) e [INSTALACIÓN DE CYGIPC](#) respectivamente. Todos los pasos deben ser seguidos.

INSTALACIÓN Y CONFIGURACIÓN DE INTERNET INFORMATION SERVER

Paquete: Internet Information Server 5.0

El Internet Information Server puede ser instalado en Windows 2000 Server, Profesional, en Windows XP Server, Profesional, en Windows NT. En este Anexo, se documenta el proceso de instalación en Windows 2000 Server. Cabe destacar que el proceso no varía para las otras versiones.

La instalación puede realizarse de dos formas: durante la instalación de Windows 2000 por defecto o manualmente después de instalar el sistema.

1. Instalar el servidor **IIS**. En el primer caso, durante la instalación de Windows 2000, habrá que verificar que el servidor(IIS) y los paquetes adicionales se encuentren seleccionados. En el segundo caso se debe ir a **Panel de Control, Agregar o quitar programas, Agregar o quitar componentes de Windows**, seleccionar **Servicios de Internet Information Server** y mediante el botón **Detalles**, elegimos qué componentes adicionales instalar. Entre los componentes se encuentran las extensiones de Frontpage, documentación, servicios adicionales de IIS, un servidor de FTP, un servidor de SMTP, Index Server, Site Server Express, Servicio de Transacción, Servidor de Certificados, servidor NNTP, etc. Para instalar el servidor web y los componentes adicionales elegimos el botón **Siguiente** e insertamos el CD de instalación.

2. Configurar el servidor web. Para configurar el IIS, ir a **Mi PC**, seleccionar la opción **Administrar**, se abre la Consola de Administración de Microsoft o **MMC**, en la lista de la izquierda, aparece **Servicios y aplicaciones** y elegimos **Servicios de Internet Information Server**. Podemos configurar el Sitio web predeterminado, tipos de archivos a servir por defecto, directorios virtuales, seguridad, etc.
3. Definir el **Sitio web predeterminado**. Por defecto el **Sitio web predeterminado** es **C:\inetpub\wwwroot**. Para publicar nuestras propias páginas es mejor crear un subdirectorio dentro del directorio anterior, por ejemplo: **C:\inetpub\wwwroot\aplicaciones** .
4. Establecer el tipo de archivo que se servirá por defecto. El archivo por defecto puede llamarse **default.htm**, **index.htm**, **default.asp**, **index.asp**; incluso se pueden definir todos los tipos anteriores. En caso de que no se defina ninguno, el servidor buscará la existencia del archivo **default.htm**. Esta configuración debe hacerse dentro de la consola **Servicios de Internet Information Server**, en la carpeta **Sitio web predeterminado**, mediante la opción **Propiedades** del menú emergente, escogiendo la pestaña **Documentos**.
5. Iniciar el servidor web. Para comprobar que Internet Information Server se ha iniciado se debe realizar lo siguiente:
 - En el escritorio, doble clic en Microsoft Internet Explorer.
 - Si no se está conectado con Internet, aparecerá un cuadro de diálogo que indica que no ha podido conectar con el sitio de Internet <http://www.microsoft.com>. Clic en Aceptar.
 - En el menú Archivo de Microsoft Internet Explorer, clic en Abrir, escribir el nombre de la máquina y, después, clic en Aceptar.

INSTALACIÓN Y CONFIGURACIÓN DE ASP

En realidad, para poder desarrollar páginas de tipo ASP en Windows, no hace falta realizar ninguna instalación ya que el entorno de desarrollo ASP es una DLL(asp.dll) que se instala conjuntamente con el servidor web. Esta DLL forma parte del Internet Information Server desde la versión 4.0.


Tampoco se necesita de ninguna configuración, es suficiente con instalar un editor de ASP, tener

conocimientos de Visual Basic y ADO para programar los scripts y acceder a la base de datos respectivamente.

VERIFICACIÓN DE LAS INSTALACIONES

Para verificar que las instalaciones de PostgreSQL, Internet Information Server y ASP funcionan, creamos una página web con el nombre **index.asp** en el directorio de publicación del Internet Information Server. Esta página contendrá un script de conexión a una base de datos de nombre **prueba** con una tabla **usuarios**.

El directorio de publicación de Internet Information Server es: `\inetpub\wwwroot`

 **Nota:** Tomar en cuenta que debe estar iniciado el servidor web Internet Information Server y la base de datos PostgreSQL.

1. Crear la base de datos y tablas en PostgreSQL.

```
postgres@nombre_pc$ postmaster -i -D /usr/share/postgresql/data
postgres@nombre_pc $ CREATEDB prueba; (en otra consola)
postgres@nombre_pc $ psql prueba
prueba => CREATE TABLE usuarios {
           nombre varchar(10),
           clave varchar(10),
           tipo_usuario varchar(10)
           };
prueba => \q
```

2. Agregar un DSN de sistema para la base de datos **prueba**. Llenar los siguientes datos:

DSN Name: conexion_postgresql

Description: Conexión a PostgreSQL mediante ASP

DataBase Type: PostgreSQL

3. Crear la página **index.asp**. Esta página desplegará los datos(excepto las contraseñas) ingresados en la tabla **usuarios**. La página debe ser creada dentro del directorio de publicación del Internet Information Server.

```

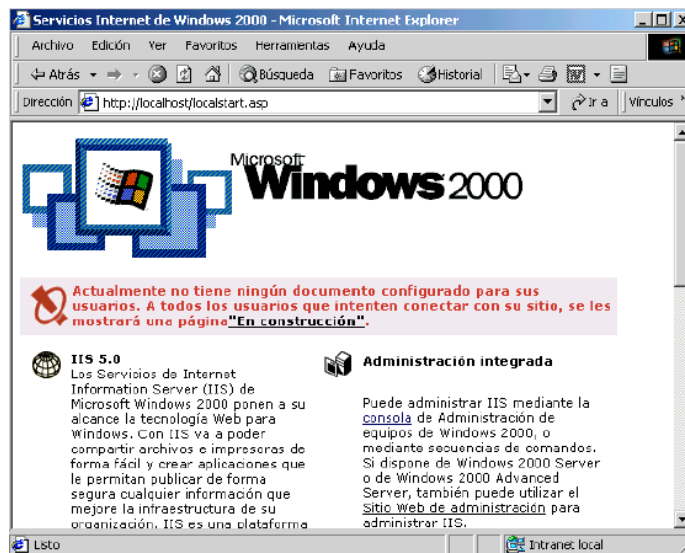
<HTML>
  <%@ LANGUAGE="VBSCRIPT" %>
  <HEAD>
    <TITLE>Página de prueba</TITLE>
  </HEAD>
  <BODY>
    /****** Conexión con PostgreSQL *****
    <%
    Dim conexion, resultado, cadena_conexion
    cadena_conexion= "DATABASE=prueba; UID=adminbdd; PWD=; DSN=postgresql_asp"
    Set conexion = Server.CreateObject ("ADODB.Connection")
    Set resultado = Server.CreateObject ("ADODB.RecordSet")
    conexion.Open cadena_conexion
    sql = "SELECT usuario, tipo_usuario FROM usuarios"
    resultado.Open sql, conexion
    Do While not resultado.EOF
      Response.Write "NOMBRE" & "          " & "TIPO DE USUARIO"
      Response.Write resultado ("usuario") & "          " & resultado ("tipo_usuario ")
      Response.Write "<BR>"
      resultado.MoveNext
    Loop
    resultado.Close
    conexion.Close
    Set resultado = Nothing
    Set conexion = Nothing
    %>
  </BODY>
</HTML>

```

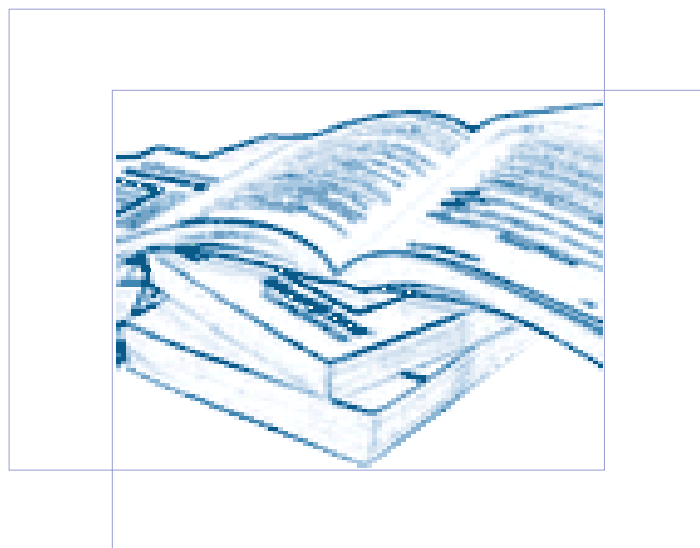
4. Desplegar la página **index.asp** en el navegador. Para desplegar la página **index.asp**, ir al navegador y escribir la siguiente dirección: **http://localhost/index.asp**. Donde **localhost** es el nombre de la máquina, dirección **IP** o dominio. Si todo ha salido bien, el resultado puede verse en la figura siguiente.



Para verificar que el servidor Internet Information Server y ASP fueron instalados correctamente, se debe escribir en un navegador la dirección: `http://localhost/localstart.asp`, si aparece la siguiente página, Internet Information Server estará listo para trabajar.



ANEXO 3



MANUALES DE INSTALACIÓN, CONFIGURACIÓN Y USO DE JSP EN LINUX Y EN WINDOWS

En este Anexo se detallan las instalaciones de los paquetes de dos formas: vía RPM y por medio de binarios

🐼 **Nota:** Debido a que el servidor Tomcat es lento para servir páginas estáticas, se instalará Apache para que haya comunicación entre los dos; Apache se encargará de servir páginas **.html** y Tomcat páginas **.jsp**.

INSTALACIÓN, CONFIGURACIÓN Y USO DE JSP EN LINUX

INSTALACIÓN VÍA RPM

Para instalar vía RPM se deben seleccionar los siguientes paquetes:

- postgresql-server.rpm y postgresql-7.2.2-1.1.rpm
- apache-2.0.40.8.rpm

Adicionalmente, se instalarán de forma automática los siguientes paquetes: postgresql-devel.rpm, postgresql-docs.rpm, postgresql-jdbc.rpm, postgresql-odbc.rpm, postgresql-libs.rpm, postgresql-perl.rpm y postgresql-python.rpm.

La instalación se realiza como **root**.

Luego de realizar la instalación es importante verificar los directorios donde se han instalado cada uno de los paquetes; de esta forma podemos conocer donde encontrar los archivos de configuración.

En este tipo de instalación, necesitamos dos paquetes adicionales: JDK y Tomcat para Linux; ambos paquetes podemos instalarlos manualmente por medio de los instaladores binarios **.tar**.

🐼 **Nota:** Para instalar JDK, ver la sección [INSTALACIÓN Y CONFIGURACIÓN DEL JDK \(ENTORNO JAVA PARA JSP\)](#). Se deben seguir todos los pasos. Para instalar Tomcat, ver la sección [INSTALACIÓN Y CONFIGURACIÓN DE TOMCAT](#).

INSTALACIÓN POR MEDIO DE BINARIOS

Al instalar por medio de binarios, se debe seguir un orden específico, el cual se detalla a continuación:

1. Instalación y configuración de la base de datos: PostgreSQL
2. Instalación y configuración de JDK(entorno JAVA para la programación con JSP)
3. Configuración de JDBC(permite la conexión de PostgreSQL con JSP)
4. Instalación y configuración de Apache Web Server
5. Creación e instalación de **mod_jk** para integrar Apache y Tomcat.
6. Instalación y configuración de Tomcat

INSTALACIÓN Y CONFIGURACIÓN DE POSTGRESQL

Paquete: Servidor de base de datos PostgreSQL para Linux

Sitio Web: <ftp://ftp4.es.postgresql.org/Mirror/www.postgresql.org/v7.1.3/postgresql-7.1.3.tar.gz>

Nombre del archivo instalador: postgresql-7.1.3.tar.gz

Directorio de descarga del instalador: /usr/src

Ver la sección [INSTALACIÓN Y CONFIGURACIÓN DE POSTGRESQL](#). La instalación es la misma, excepto que en el paso **3** se debe aumentar el módulo **--with-java**. Este módulo sirve para trabajar con PostgreSQL mediante **JDBC**. Este paso se indica a continuación:

3. Empezar el proceso de compilación/instalación de PostgreSQL. Para ello ingresamos con el usuario **postgres**.

```
# su postgres
# cd /usr/src/pgsql
# gunzip postgresql-7.1.3.tar.gz
# tar -xvf postgresql-7.1.3.tar
# cd /usr/src/pgsql/postgresql-7.1.3
# ./configure --prefix=/usr/local/pgsql --with-odbc --with-java --enable-locale
```

En el script **configure** se indica el directorio donde se instalará PostgreSQL en el parámetro **prefix**, además se indica que será compilado con soporte **ODBC** y **JDBC(--with-java)**.

👉 **Nota:** Luego de configurar PostgreSQL con soporte a **JDBC**, se siguen los demás pasos descritos en la sección [INSTALACIÓN Y CONFIGURACIÓN DE POSTGRESQL](#).

INSTALACIÓN Y CONFIGURACIÓN DEL JDK (ENTORNO JAVA PARA JSP)

Paquete: JDK(Java Development Kit) para Linux.

Sitio Web: <http://java.sun.com/j2se>

Nombre del archivo instalador: `jdk-1_2_2_011-linux.i386.tar.gz`

Directorio de descarga del instalador: `/usr/local/`

1. Desempaquetar el instalador `jdk-1_2_2_011-linux.i386.tar.gz` en el directorio `/usr/local`

```
# cd /usr/local tar -xvf jdk-1_2_2_011-linux.i386.tar.gz
```

Este paso crea el directorio `/usr/local/jdk 1_2_2_011`, que puede ser cambiado a `/usr/local/jdk` para trabajar con mayor facilidad.

2. Dar valores a las variables de entorno. Las variables de entorno deben ser añadidas en el archivo `.bash_profile` del directorio **home** del usuario **postgres**.

```
# vi ~/.bash_profile
JAVA_HOME=/usr/local/jdk
CLASS_PATH=$JAVA_HOME/jre/lib/tools.jar
PATH=$JAVA_HOME/bin
export JAVA_HOME CLASS_PATH PATH
```

El archivo empaquetado `tools.jar` contiene la herramienta **javac** que es necesaria para la compilación de los archivos `.jsp` correspondientes a las páginas JSP.

Con la instalación del **JDK**(entorno Java para programación de JSP), se instala también el **API JDBC**(paquete `java.sql`) para acceso a bases de datos.

CONFIGURACIÓN DE JDBC

Paquete: Controlador JDBC para PostgreSQL en Linux

Sitio Web: <http://jdbc.postgresql.org/download.html>

Nombre del archivo instalador: jdbc7.1-1.2.jar

Directorio de descarga del instalador: /usr/local/jdk

Luego de instalar PostgreSQL con soporte para JDBC e instalar el JDK, se debe configurar el controlador JDBC. Dicho controlador debe ser adecuado para la versión de PostgreSQL y del JDK. Se puede descargar del siguiente sitio: <http://jdbc.postgresql.org/download.html>.

Las versiones más nuevas de los controladores tienen extensión **.jar**, mientras que las versiones antiguas tienen extensión **.tar** o **.tar.gz**. Si se cuenta con las dos últimas extensiones, primero se debe descomprimirlas usando el siguiente comando:

```
# cd /usr/local/jdk/  
tar -xvf jdbc7.1-1.2.tar  
o con:  
tar -zxvf jdbc7.1-1.2.tar.gz
```

En cualquiera de los casos se obtendrá un archivo **.jar**, ya que al descomprimir un controlador **.tar** o **.tar.gz** también se obtiene un archivo **.jar**. El archivo **jdbc7.1-1.2.jar** deberá estar en el directorio: **/usr/local/jdk/**.

La ruta donde se encuentra el archivo **jdbc7.1-1.2.jar** deberá ser añadida a la variable de entorno **CLASS_PATH**. La variable **CLASS_PATH** es la que se configuró al instalar el **JDK** en la sección [INSTALACIÓN Y CONFIGURACIÓN DEL JDK \(ENTORNO JAVA PARA JSP\)](#). Esta variable se encarga de determinar dónde buscar: clases, librerías y controladores de Java para que el **JDK** sepa dónde encontrarlos.

Las variables de entorno se encuentran en el archivo **.bash_profile** del directorio **home** del usuario **postgres**. La variable **CLASS_PATH** debe quedar como se indica a continuación:

```
# vi ~/.bash_profile  
JAVA_HOME=/usr/local/jdk  
CLASS_PATH=$JAVA_HOME/jre/lib/tools.jar  
CLASS_PATH=.%JAVA_HOME%/jdbc7.1-1.2.jar; %CLASS_PATH%  
PATH=$JAVA_HOME/bin  
export JAVA_HOME CLASS_PATH PATH
```

Para que los cambios surtan efecto hay que reiniciar la máquina.

En este punto la base de datos PostgreSQL se encuentra lista para aceptar conexiones de scripts JSP por medio de JDBC.

INSTALACIÓN Y CONFIGURACIÓN DE APACHE WEB SERVER

Paquete: Apache Web Server para Linux

Sitio Web: http://www.apache.org/dist/httpd/apache_1.3.20.tar.gz

Nombre del archivo instalador: apache_1.3.20.tar.gz

Directorio de descarga del instalador: /usr/src

1. Descomprimir y desempaquetar el código fuente en el directorio **/usr/src**. Se debe ingresar al sistema como el superusuario **root**.

```
# su
# cd /usr/src
# gunzip apache_1.3.20.tar.gz
# tar -xvf apache_1.3.20.tar
# cd apache_1.3.20
# ./configure --prefix=/usr/local/apache --enable-module=so --enable-rule=SHARED_CORE
```

Con **enable-module=so** se añade soporte para módulos **.so**, especialmente para el módulo **mod_jk.so** de Tomcat, dicho módulo sirve para la integración Apache-Tomcat.

2. Instalar Apache.

```
#make
#make install
```

3. Iniciar Apache. Este paso se realiza para comprobar si la instalación fue correcta.

```
#cd /usr/local/apache/bin/apachectl star
```

Si todo está correcto, pasamos a la instalación de Tomcat. Más adelante se procederá a la configuración de Apache, luego de instalar Tomcat y **mod_jk.so**(módulo para integrar Apache con Tomcat).

Para que la instalación de Tomcat y la integración del mismo con Apache tengan éxito, es necesario parar Apache.

```
#cd /usr/local/apache/bin/apachectl stop
```

INSTALACIÓN Y CONFIGURACIÓN DE TOMCAT

Paquete: Servidor Tomcat para Linux.

Sitio Web: <http://www.apache.org/dist/jakarta/jakarta-tomcat-4.0/release/v4.1.18/bin/jakarta-tomcat-4.1.18.tar.gz>

Nombre del archivo instalador: jakarta-tomcat-4.1.18.tar.gz

Directorio de descarga del instalador: /usr/local

1. Descomprimir y desempaquetar el archivo **jakarta-tomcat-4.1.18.tar.gz** en el directorio **/usr/local/**. Este paso crea el directorio **/usr/local/jakarta-tomcat-4.1.18**, que puede ser cambiado a **/usr/local/tomcat** para mayor facilidad.

```
# cd /usr/local/  
# gunzip jakarta-tomcat-4.1.8.tar.gz  
# tar -xvf jakarta-tomcat-4.1.8.tar  
# rename /usr/local/tomcat
```

2. Definir variable de entorno en el archivo **.bash_profile** localizado dentro del directorio **home** del usuario. A partir de la versión 4 de Tomcat, la variable de entorno se llama **CATALINA_HOME**.

```
# vi ~/.bash_profile  
    CATALINA_HOME=/usr/local/tomcat  
    export CATALINA_HOME
```

En este punto Tomcat puede ser iniciado, pero se recomienda primero instalar **mod_jk.so**, configurar Apache y posteriormente iniciar Tomcat.

CREACIÓN E INSTALACIÓN DE MOD_JK

Paquete: Módulo para integrar Apache con Tomcat en Linux.

Nombre del archivo instalador: mod_jk.so

Directorio del instalador: /usr/local/tomcat/src/native/apache

Si la instalación de PostgreSQL y Apache se han realizado vía **RPM**, para la creación e instalación de **mod_jk.so** se debe tomar en cuenta que el directorio de instalación de Apache no es **/usr/local/apache**, dicho directorio debe ser cambiado por el directorio en donde el Administrador de Paquetes **RPM** instaló Apache.

1. Crear el módulo **mod_jk.so**. Este módulo sirve para la integración de Apache con Tomcat, se encuentra en el directorio **/usr/local/tomcat/src/native/apache**, y debe ser creado con el comando **apxs** que viene con Apache.

```
# cd /usr/local/tomcat/src/native/apache
# apxs -o mod_jk.so -I ../jk -I/usr/local/jdk/include/ -I/usr/local/jdk/include/linux -c *.c ../jk/*.c
```

El comando **apxs** está incluido en el directorio **/usr/local/apache/bin/apxs** y es utilizado para compilar módulos empleados por productos que operan en conjunción con Apache. El parámetro **-I** indica los directorios que serán incluidos en la compilación del módulo. Con este paso **mod_jk.so** está creado.

2. Instalar **mod_jk.so**. Para la instalación de **mod_jk.so** basta copiar el módulo al directorio **/usr/local/apache/libexec/**.

```
# cd /usr/local/tomcat/src/native/apache/
# cp mod_jk.so /usr/local/apache/libexec/
```

Una vez que se ha copiado el módulo, es necesario configurar el archivo **httpd.conf** que se encuentra en **/usr/local/apache/conf/**.

3. Modificar el archivo **httpd.conf** editando las líneas que se indican a continuación:

```
# cd /usr/local/apache/conf/
# vi httpd.conf
    LoadModule jk_module libexec/mod_jk.so
    AddModule mod_jk.c
    JkWorkersFile /usr/local/tomcat/conf/workers.properties
    JkLogFile /usr/local/tomcat/log/mod_jk.log
    JkLogLevel warn
```

Básicamente, los parámetros que se modifican son: **LoadModule** para indicar donde se encuentra **mod_jk.so**, **JkWorkersFile** para indicar como se comunica Apache con Tomcat, **JkLogFile** que establece el archivo donde se almacenarán los errores generados por la comunicación de Apache-Tomcat y **JkLogLevel** indica el nivel de errores soportados.

4. Editar el archivo **workers.properties** del directorio **/usr/local/tomcat/conf**

```
# cd /usr/local/tomcat/conf/
# vi workers.properties
workers.tomcat_home=/usr/local/tomcat
workers.java_home=/usr/local/jdk
ps=/
worker.list=ajp12, ajp13
worker.ajp13.port=8009
worker.ajp13.host=localhost
worker.ajp13.type=ajp13
```

El valor **ajp13** es el protocolo para comunicar Apache con Tomcat. El puerto **8009** es el que será utilizado por Tomcat.

5. Indicar a Apache que solicitudes debe enviar a Tomcat. Para esto se edita el parámetro **JkMount** dentro del **httpd.conf**.

```
# cd /usr/local/apache/conf/
# vi httpd.conf
JkMount /*.jsp ajp13
JkMount /servidordatos/Insertar ajp13
```

En **JkMount** se indica que todos los archivos con extensión **.jsp** serán pasados a Tomcat por medio del protocolo **ajp13**.

6. Iniciar los servidores. Para que la comunicación entre Tomcat y Apache funcione, se debe iniciar primero el servidor Tomcat y luego Apache.

```
# cd /usr/local/tomcat/bin
# ./startup.sh
# cd /usr/local/apache/bin/httpd star
```

Si todo va bien, al iniciar Tomcat, podemos ver las siguientes líneas:

```
[root@servidor1 bin]# ./startup.sh
Using CLASSPATH:/usr/local/tomcat/bin/bootstrap.jar:/usr/local/jdk/jre/lib/tools.jar
Using CATALINA_BASE: /usr/local/tomcat/
Using CATALINA_HOME: /usr/local/tomcat/
Using JAVA_HOME: /usr/local/jdk
```


Si se desea realizar alguna configuración en cualquiera de los servidores, primero se debe detener Apache y luego Tomcat.

```
# cd /usr/local/apache/bin/httpd stop
# cd /usr/local/tomcat/bin
# ./shutdown.sh
```

VERIFICACIÓN DE LAS INSTALACIONES

Para verificar que las instalaciones de PostgreSQL, Apache-Tomcat y JSP funcionan, creamos una página web con el nombre **index.jsp** en el directorio **/usr/local/tomcat/webapps**. Esta página contendrá un script de conexión a una base de datos de nombre **prueba** con una tabla **usuarios**.

Cabe destacar que la página **index.jsp** se conectará a la base de datos PostgreSQL por medio del controlador JDBC: **jdbc7.1-1.2.jar** y utilizará el paquete **java.sql**.

 **Nota:** Tomar en cuenta que debe estar iniciado el servidor web Tomcat, el servidor web Apache y la base de datos PostgreSQL.

1. Crear la base de datos y tablas en PostgreSQL. Una vez creada la base de datos, se debe crear el usuario **apache** y darle privilegios para trabajar sobre las tablas.

```
# su postgres
# postmaster -i -D /usr/local/pgsql/data
# su postgres (en otra consola)
# CREATEDB prueba;
# psql prueba
prueba => CREATE TABLE usuarios {
           nombre varchar(10),
           clave varchar(10),
```

```

        tipo_usuario varchar(10)
    };
prueba => \q
#createuser apache;
# psql prueba
prueba => GRANT SELECT ON usuarios TO apache;
prueba => \q

```

2. Crear la página **index.php**. Esta página desplegará los datos(excepto las contraseñas) ingresados en la tabla **usuarios**. La página debe ser creada dentro del directorio de publicación de Tomcat: **/usr/local/tomcat/webapps**.

```

<HTML>
  <%@page import="java.sql.*"%>
  <HEAD>
    <TITLE>Página de prueba</TITLE>
  </HEAD>
  <BODY>
  <%
    try
    {
      String url="jdbc:postgresql://localhost/";
      String usuario_bdd="usuario";
      String clave_bdd="";
      Driver driver_bdd= (Driver) Class.forName("org.postgresql.Driver");
      Connection conexion = DriverManager.getConnection(url, "usuario_bdd", "clave_bdd");
      ResultSet resultado = stmt.executeQuery("SELECT nombre, tipo_usuario FROM usuarios ");
      System.out.println("NOMBRE" + " " + "TIPO DE USUARIO");
      while (resultado.next())
      {
        String usuario = resultado.getString("usuario");
        String tipo = resultado.getString("tipo_usuario");
        System.out.println(usuario+" " + tipo);
        System.out.println("<BR>");
      }
      resultado.close();
      conexion.close();
    }
    catch(SQLException error_conexion)

```

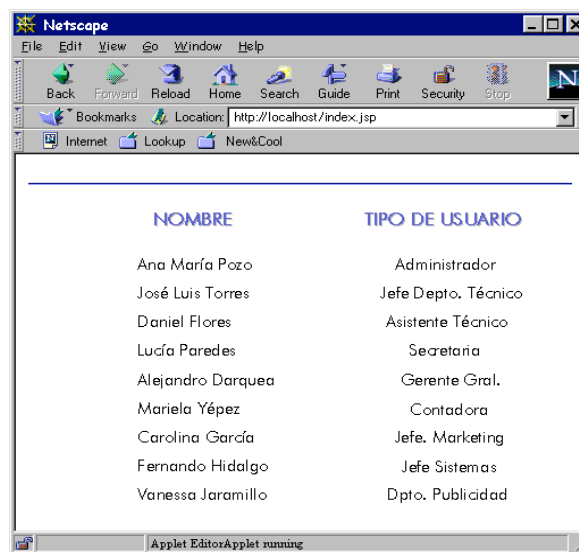


```

{
    System.out.println("Durante la conexión se produjo el error:" + error_conexion);
}
%>
</BODY>
</HTML>

```

- Desplegar la página **index.jsp** en el explorador. Para desplegar la página **index.jsp**, ir al navegador y escribir la siguiente dirección: **http://localhost/index.jsp**. Donde localhost es el nombre de la máquina, dirección IP o dominio. Si todo ha salido bien, el resultado puede verse en la figura siguiente.



También se puede verificar que Tomcat está instalado y configurado correctamente, escribiendo en un navegador la siguiente dirección: **http://localhost:8009/index.html**, si aparece la siguiente página Tomcat está listo para trabajar.



INSTALACIÓN, CONFIGURACIÓN Y USO DE JSP EN WINDOWS

Para la instalación en Windows, se debe instalar los paquetes en un orden específico:

1. Instalación y configuración de la base de datos: PostgreSQL
2. Instalación y configuración de JDK(entorno JAVA para la programación con JSP)
3. Configuración de JDBC(permite la conexión de PostgreSQL con JSP)
4. Instalación y configuración de Apache Web Server
5. Creación e instalación de mod_jk para integrar Apache y Tomcat.
6. Instalación y configuración de Tomcat

INSTALACIÓN Y CONFIGURACIÓN DE POSTGRESQL

Para la instalación de PostgreSQL en Windows se deben instalar las utilidades: **Cygwin** y **CyglPC**. La instalación de dichas utilidades se encuentra detallada en las secciones: [INSTALACIÓN DE CYGWIN](#) e [INSTALACIÓN DE CYGIPC](#) respectivamente. Todos los pasos deben ser seguidos. Más adelante, se configurará JDBC para conectar PostgreSQL con JSP.

INSTALACIÓN Y CONFIGURACIÓN DEL JDK (ENTORNO JAVA PARA JSP)

Paquete: JDK(Java Development Kit) para Windows.

Sitio Web: java.sun.com/j2se

Nombre del archivo instalador: `jdk-1_2_2_011-linux.i386.tar.gz`

Directorio de descarga del instalador: `C:\`

1. Desempaquetar el instalador `jdk-1_2_2_011-linux.i386.tar.gz` en el directorio `C:\`. No importa que el instalador sea `.tar.gz`, el **WinZip** de Windows si descomprime este tipo de archivo. Debido a que el entorno Java es multiplataforma, el instalador binario sirve igualmente en Linux y en Windows.

Este paso crea el directorio `C:\jdk-1_2_2_011`, que puede ser cambiado a `C:\jdk` para trabajar con mayor facilidad.

2. Dar valores a las variables de entorno. Las variables de entorno deben ser añadidas por medio del icono **Mi PC** en **Propiedades del Sistema**, pestaña **Avanzado** y botón **Variables de entorno**. En **Variables de entorno** elegir **Variables de usuario** y crear las nuevas variables o editarlas en caso de que existan. Los valores que deben contener son los que se indican, dependiendo del directorio de instalación del JDK, en este caso son:

```
JAVA_HOME=C:\jdk  
CLASS_PATH=%JAVA_HOME\jre\lib\tools.jar  
PATH=%JAVA_HOME\bin
```

El archivo empaquetado **tools.jar** contiene la herramienta **javac** que es necesaria para la compilación de los archivos **.jsp** correspondientes a las páginas JSP.

Con la instalación del **JDK**(entorno Java para programación de JSP), se instala también el **API JDBC**. Este API JDBC se encuentra encapsulado en el paquete **java.sql** y sirve para el acceso a bases de datos desde entornos de programación Java, por lo tanto, en este caso desde JSP.

CONFIGURACIÓN DE JDBC

Paquete: Controlador JDBC para PostgreSQL en Windows

Nombre del archivo instalador: postgresql.jar

Directorio del instalador: C:\Cygwin\usr\share\postgresql\java\

Además de instalar Cygwin y CygIPC, se debe configurar **JDBC**. Al instalar Cygwin y CygIPC se instala PostgreSQL con soporte para **JDBC**(Conectividad a Base de Datos Java), ya que los archivos instaladores vienen compilados con Java. Pero esto no es suficiente, hay que configurar el controlador **JDBC**, dicho controlador se encuentra localizado dentro del directorio donde se instaló Cygwin, la ruta es la siguiente:

```
C:\Cygwin\usr\share\postgresql\java\postgresql.jar
```

La configuración del controlador **postgresql.jar** consiste en dos pasos: primero, copiar el archivo **postgresql.jar** al directorio **C:\jdk**. Segundo, una vez copiado el archivo, se debe añadir la ruta **C:\jdk\postgresql.jar** a la variable de entorno **CLASS_PATH**. La variable **CLASS_PATH** es la que se configuró al instalar el **JDK** en la sección [INSTALACIÓN Y CONFIGURACIÓN DEL JDK \(ENTORNO](#)

[JAVA PARA JSP](#)). Esta variable se encarga de determinar dónde buscar: clases, librerías y controladores de Java para que el **JDK** sepa dónde encontrarlos.

La configuración debe quedar como se indica a continuación:

```
JAVA_HOME=C:\jdk
CLASS_PATH=.\%JAVA_HOME%\postgresql.jar; %CLASS_PATH%
PATH=%JAVA_HOME%\bin
```

Para que los cambios surtan efecto hay que reiniciar la máquina.

En este punto la base de datos PostgreSQL se encuentra lista para aceptar conexiones de scripts JSP por medio de JDBC.

INSTALACIÓN Y CONFIGURACIÓN DE APACHE WEB SERVER

Paquete: Apache Web Server para Windows

Sitio Web: <http://www.apache.org/dist/httpd/apache-1.3.12-Win32-src.zip>

Nombre del archivo instalador: apache-1.3.12-Win32-src.zip

Directorio de descarga del instalador: C:\Downloads\Apache\

El proceso de instalación de Apache en Windows es el mismo que se encuentra detallado en la sección [INSTALACIÓN Y CONFIGURACIÓN DE APACHE](#) (los 5 primeros pasos deben ser seguidos y luego se debe continuar en esta sección).

Debido a que Apache cooperará con Tomcat, la configuración de Apache se debe realizar luego de instalar Tomcat y el módulo **mod_jk** que hace que ambos se comuniquen. El servidor Apache debe ser iniciado luego de que se haya instalado Tomcat, el módulo **mod_jk**, configurado el archivo **httpd.conf** e iniciado Tomcat.

INSTALACIÓN Y CONFIGURACIÓN DE TOMCAT

Paquete: Tomcat para Windows.

Sitio Web: <http://www.apache.org/dist/jakarta/jakarta-tomcat-4.0/release/v4.1.18/bin/jakarta-tomcat-4.1.18.exe>

Nombre del archivo instalador: jakarta-tomcat-4.1.18.exe

Directorio de descarga del instalador: C:\

1. Ejecutar el instalador. Este paso creará la carpeta C:\tomcat.
2. Crear la variable de entorno para Tomcat. Ir a **Mi PC, Propiedades del Sistema**, pestaña **Avanzado** y botón **Variables de entorno**. En **Variables de entorno** elegir **Variables de usuario** y crear la nueva variable. El valor que deben contener se indica a continuación:

```
TOMCAT=C:\tomcat
```

Adicionalmente a la variable TOMCAT_HOME, se debe verificar la existencia de las variables de entorno del JDK(ver el siguiente recuadro), ya que estas variables son útiles para el funcionamiento de Tomcat y deben estar establecidas antes de iniciar Tomcat.

```
JAVA_HOME=C:\jdk  
CLASS_PATH=%JAVA_HOME\jre\lib\tools.jar  
PATH=%JAVA_HOME\bin
```

En este punto Tomcat puede ser iniciado, pero se recomienda primero instalar **mod_jk**, configurar Apache y posteriormente iniciar Tomcat.

CREACIÓN E INSTALACIÓN DE MOD_JK

Paquete: Módulo para integrar Apache con Tomcat en Windows.

Sitio Web: http://jakarta.apache.org/builds/jakarta-tomcat/release/v3.3/bin/win32/i386/mod_jk.dll

Nombre del archivo instalador: mod_jk.dll

Directorio de descarga del instalador: C:\

1. Instalar **mod_jk**. Simplemente se debe copiar el archivo **mod_jk.dll** en el subdirectorio **/modules** que se encuentra dentro del directorio de instalación de Apache.
2. Modificar el archivo **httpd.conf** de Apache. Este archivo se modifica para que Apache utilice a Tomcat para servir las páginas .jsp y los Servlets, mientras que Apache se encarga de servir las páginas estáticas. Para ello añadimos en **httpd.conf** la siguiente línea al final del archivo:

```
Include C:\tomcat\conf\jk\mod_jk.conf
```

3. Modificar el archivo **mod_jk.conf** del directorio **C:\tomcat\conf\jk**. En este archivo editar las directivas que se indican a continuación:

```
LoadModule jk_module libexec/mod_jk.dll
AddModule mod_jk.c
JkWorkersFile C:\tomcat\conf\jk\workers.properties
JkLogFile C:\Archivos de Programa\Apache Group\Apache\logs\mod_jk.log
JkLogLevel warn
JkMount /*.jsp ajp13
JkMount /servlet/* ajp13
JkMount /otherworker/*.jsp remoteworker
```

El valor **ajp13** es el protocolo para comunicar Apache con Tomcat

4. Iniciar el servidor Tomcat. Para iniciar el servidor lanzamos en una consola MS-DOS el script **startup.bat** que se encuentra en el directorio **C:\tomcat\bin**:

```
C:\tomcat\bin\startup.bat
```

Si todo va bien, al iniciar Tomcat, podemos ver las siguientes líneas:

```
Using CLASSPATH: C:\tomcat\bin\bootstrap.jar;%jdk%\jre\lib\tools.jar
Using CATALINA_BASE: C:\tomcat\
Using CATALINA_HOME: C:\tomcat\
Using JAVA_HOME: C:\jdk
```

Una vez iniciado Tomcat, iniciamos Apache desde: **Inicio – Programas – Apache Web Server – Apache Start**.

Para detener Tomcat, primero se debe detener Apache(desde una consola MS-DOS), los pasos se indican a continuación:

```
apache -k shutdown
C:\tomcat\bin\shutdown.bat
```

Cada vez que se solicite una página .jsp, el servidor web Apache redirigirá la solicitud a Tomcat.

VERIFICACIÓN DE LAS INSTALACIONES

Para verificar que las instalaciones de PostgreSQL, Apache-Tomcat y JSP funcionan, creamos una página web con el nombre **index.jsp** en el directorio **C:\tomcat\webapps**. Esta página contendrá un script de conexión a una base de datos de nombre **prueba** con una tabla **usuarios**.

Cabe destacar que la página **index.jsp** se conectará a la base de datos PostgreSQL por medio del controlador JDBC: **postgresql.jar** y utilizará el paquete **java.sql**.

Nota: Tomar en cuenta que debe estar iniciado el servidor web Tomcat, el servidor web Apache y la base de datos PostgreSQL.

1. Crear la base de datos y tablas en PostgreSQL. Una vez creada la base de datos, se debe crear el usuario **apache** y darle privilegios para trabajar sobre las tablas.

```
postgres@nombre_pc$ postmaster -i -D /usr/share/postgresql/data
postgres@nombre_pc $ CREATEDB prueba; (en otra consola)
postgres@nombre_pc $ psql prueba
prueba => CREATE TABLE usuarios {
           nombre varchar(10),
           clave varchar(10),
           tipo_usuario varchar(10)
        };
prueba => \q
#createuser apache;
# psql prueba
prueba => GRANT SELECT ON usuarios TO apache;
prueba => \q
```

2. Crear la página **index.php**. Esta página desplegará los datos(excepto las contraseñas) ingresados en la tabla **usuarios**. La página debe ser creada dentro del directorio de publicación de Tomcat: **C:\tomcat\webapps**.

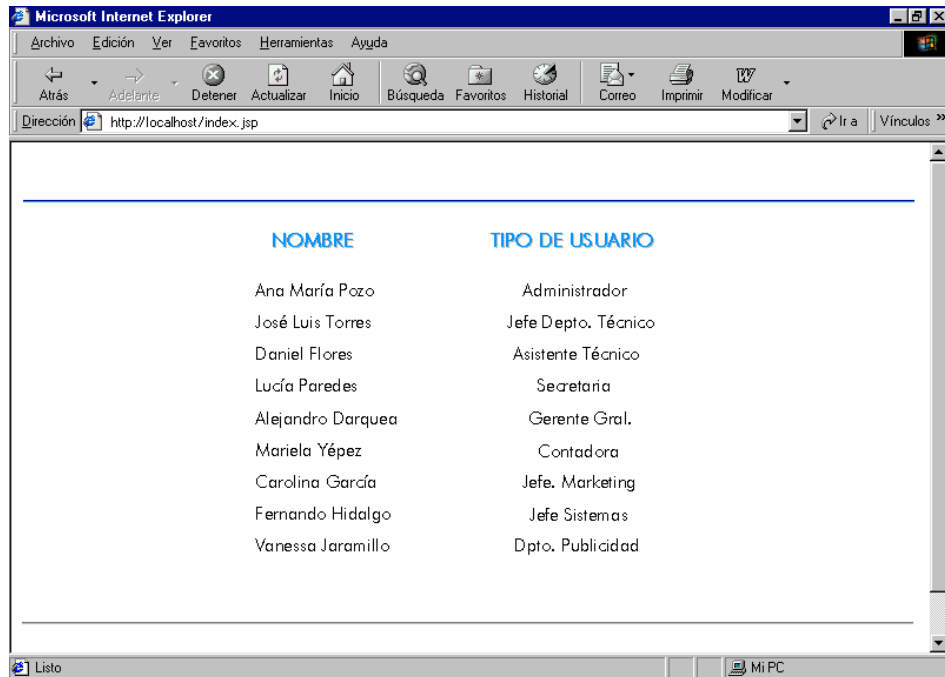
```
<HTML>
  <%@page import="java.sql.*"%>
  <HEAD>
    <TITLE>Página de prueba</TITLE>
```

```

</HEAD>
<BODY>
<%
try
{
String url="jdbc:odbc:postgresql://localhost/";
String usuario_bdd="usuario";
String clave_bdd="";
Driver driver_bdd= (Driver) Class.forName("org.postgresql.Driver");
Connection conexion = DriverManager.getConnection(url, "usuario_bdd", "clave_bdd");
ResultSet resultado = stmt.executeQuery("SELECT nombre, tipo_usuario FROM usuarios ");
System.out.println("NOMBRE" + " " + "TIPO DE USUARIO");
while (resultado.next())
{
String usuario = resultado.getString("usuario");
String tipo = resultado.getString("tipo_usuario");
System.out.println(usuario+" "+tipo);
System.out.println("<BR>");
}
resultado.close();
conexion.close();
}
catch(SQLException error_conexion)
{
System.out.println("Durante la conexión se produjo el error:" + error_conexion);
}
%>
</BODY>
</HTML>

```

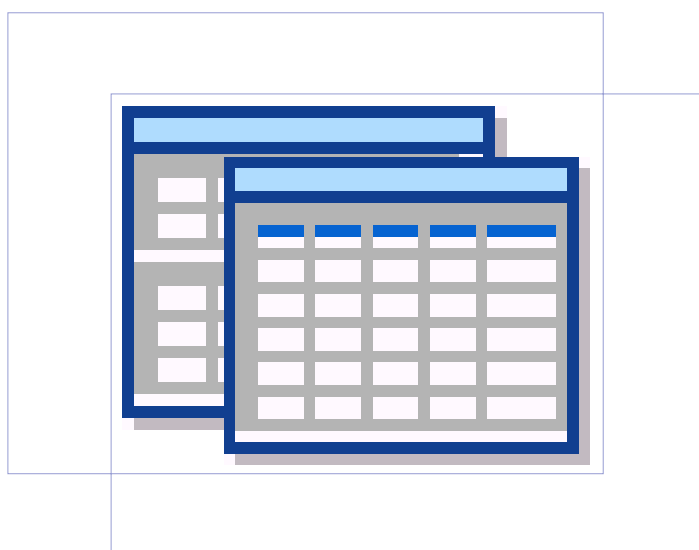
3. Desplegar la página **index.jsp** en el explorador. Para desplegar la página **index.jsp**, ir al navegador y escribir la siguiente dirección: **http://localhost/index.jsp**. Donde localhost es el nombre de la máquina, dirección IP o dominio. Si todo ha salido bien, el resultado puede verse en la figura siguiente.



También se puede verificar la instalación y configuración de Tomcat escribiendo en un navegador la dirección: <http://localhost:8009/index.html>, si aparece la siguiente página, Tomcat estará listo para trabajar.



ANEXO 4



TABLAS

1. Comparación de los Sistemas Operativos Windows 2000 Server y Linux Red Hat(6.0 -7.x).
2. Combinación de Herramientas para programar Aplicaciones Web.
3. Servicios proporcionados en Internet y sus respectivos puertos.
4. Evaluación de Servidores Web gratuitos y comerciales que trabajan sobre WINDOWS/LINUX y soportan: PHP/ASP/JSP.
5. Directivas de configuración del archivo httpd.conf de Apache Web Server(ejemplos en LINUX).

CUALIDADES TOMADAS EN CUENTA PARA LA TABLA 1

Las cualidades deseables de los servidores de web son:

- a) Disponibilidad: Prestación de servicios sin interrupción.
- b) Seguridad: Impedimento del acceso no autorizado a datos.
- c) Escalabilidad: Mantenimiento de tiempo de respuesta ante mayores requerimientos.
- d) Versatilidad: Variedad de tecnologías, servicios e interfases soportados.

Estas cualidades se relacionan de la siguiente manera:

- La versatilidad de servicios puede influir o no en cada cualidad según el servicio.
- La mayor seguridad aumenta la disponibilidad, ya que disminuye el riesgo de ataques de hackers.
- La mayor escalabilidad aumenta la disponibilidad, ya que disminuye el riesgo de que el sistema colapse por un exceso de requerimientos.
- A su vez, la disponibilidad implementada como duplicación con balance de carga, permite mejorar el rendimiento, y por lo tanto la escalabilidad.

Tabla 1: Comparación de los Sistemas Operativos Windows 2000 Server y Linux Red Hat(6.0 -7.x)♦

PARÁMETROS DE COMPARACIÓN		WINDOWS 2000 SERVER	LINUX	
DISPONIBILIDAD Es el porcentaje de tiempo en que el sistema brinda los servicios.	Tolerancia a fallo de zona (cluster remoto o multisitio)	Posee servicios de replicación por los cuales se podrían replicar estructuras de directorio de un servidor a otro a través de una red, incluyendo internet Las aplicaciones pueden implementarse a través de sitios conectados por MSMQ	No tiene soporte en particular de replicación a distancia. El soporte a las aplicaciones teóricamente puede implementarse a través de IPCs.	
	Tolerancia a fallo de Servidor	Replicación de almacenamiento.	Incluye "Replication Services"	La replicación debe implementarse manualmente.
		Recursos compartidos (cluster)	Windows 2000 Advanced Server es la versión de Windows 2000 para clusters, y la de Data Center es potente.	No tiene límite en cuanto a la cantidad de nodos en un cluster.
	Tolerancia a fallos de disco	RAID por hardware	Windows tiene como siempre un soporte casi absoluto de hardware de RAID.	Soporta los principales controladores para dispositivos RAID como AMI MegaRaid, DAC 960 y Compaq SmartArray.
		RAID por software	Windows 2000 tiene un soporte de RAID muy rico heredado del NT.	Desde Redhat 6.0 incluye soporte RAID completo, de niveles 0,1,4 y 5, y "Hot Swap". Es manejado y configurado por el kernel. Tiene soporte lineal y proceso de reconstrucción por "threads"
	Tolerancia a fallos de conexión de red		Permite el uso de dos placas de red con la misma IP, con soporte Hot Swap.	Permite el uso de dos placas de red con la misma IP.
	Fallas de software	Estabilidad	NT ejecuta las aplicaciones en modo protegido, o sea impidiendo que un programa acceda a memoria de otro programa o del mismo SO. NT adquirió fama de inestable e inmaduro, pero con el tiempo y los parches que fue realizando Microsoft, se fue consolidando como servidor. Windows 2000, según Microsoft, es 3 veces más confiable que NT.	Linux, al igual que Unix en general, goza de reputación como sistema estable incluso en hardware viejo como 386. Las aplicaciones pueden fallar y el sistema se puede hacer lento pero continúa prestando servicios.
		Monitoreo	El sistema Event Logging de NT y Windows 2000 permite interacción con SNMP.	Linux, al igual que Unix, utiliza el servicio syslogd para registrar eventos. Este es configurable y puede generar varios logs (además del /etc/syslog) según el nivel de criticidad y de la aplicación.

♦ Fuente: http://www.geocities.com/negro_bros/eduguia/dmassaldi/tesis.htm

	Reasignación de prioridades	Windows 2000 tiene un administrador de procesos que permite asignar prioridades. Además dispone de "Quality of Services", según Microsoft una solución para garantizar ancho de banda y disponibilidad a determinados servicios.	Los comandos de shell top (para ver los procesos que más consumen procesador) y nice (para cambiar prioridad de procesos) permiten ver y asignar prioridades, pero no en forma automática. Esto se podría programar en scripts o lenguaje C según los casos. Linux soporta SNMP que combinado con traffic shaping, verifica el ancho de banda para cada dispositivo de red y puerto, y permite su reasignación.
	Migración de datos	Si, lo incluye.	No. Debe implementarse manualmente.
Interrupciones Programadas	Interrupción por actualización	NT debía ser rebooteado por cada modificación o instalación de software del sistema, o de aplicación. Windows 2000 ha mejorado hasta casi no requerir rebooteo.	Linux permite la configuración mediante la recompilación de su kernel, lo cual hasta el kernel 2.0 implicaba el rebooteo. El kernel 2.2 permite configurar al kernel a través del virtual file system sin detener el funcionamiento, en la mayoría de los casos.
	Por reconfiguración	NT es famoso por la necesidad de rebooteo por cualquier instalación de hardware, además de la baja de servicio por apertura de la máquina. O por cambio de IP por ejemplo. Windows 2000 no requiere rebotear para configurar hardware casi en ningún caso.	Linux no requiere rebooteos por cambios de configuración de software o hardware.
Ataques externos		Microsoft mantiene en internet información sobre seguridad y parches. Terceros ofrecen detección de vulnerabilidades: Webtrends, Network associates	Redhat y la comunidad Linux y de seguridad mantienen información y parches de seguridad disponibles en internet. Terceros que ofrecen detección de vulnerabilidades: Internet Security Systems Inc: Internet Scanner Network Associates.

ESCALABILIDAD

Es la capacidad de mejorar el tiempo de respuesta del SO en la actualidad, y en el futuro inmediato mediante nuevas tecnologías

Plataformas de 64 bits		Windows 2000 soporta los procesadores Intel de 32 bits y Alpha y el próximo procesador Itanium de Intel, de 64 bits.	Linux se puede instalar sobre las Ultra Sparc de 64 bits. La comunidad Linux está trabajando para soportar los procesadores Intel.
SMP		Windows 2000, en sus diversas versiones, incluye soporte hasta 64 procesadores, según Microsoft.	Permite hasta 16 procesadores, aunque después de los 4 su eficacia va disminuyendo. El kernel 2.4 incluye varias mejoras para SMP.
RAM	Máximo direccionable	Windows 2000 Server: 2 GB , Advanced Server : 8 Gb, DataCenter Server: 64 Gb	INTEL o Alpha: 1 GB, hasta 2 reconfigurando el kernel. SPARC32, 3.5 GB. SPARC64 64Gb Además, permite partición SWAP de 2 GB
Gigaethernet		Windows lo soporta	Linux lo soporta
Canal de fibra (Fiber Chanel)		Windows 2000 lo soporta.	Linux no parece soportarlo
Tarjeta de SSL		Windows 2000 lo Soporta	No parece soportarlo aún.
I2O (Intelligent Input/Output)		Windows 2000 lo incluye.	Linux no lo soporta, pero lo hará con el kernel 2.4. A la comunidad linux le interesa esta tecnología por su independencia de plataforma.
Filesystems	Soporte para 64 bits	Posee soporte de 64 bits en su filesystem	Posee soporte de 64 bits en su filesystem
Archivos	Nombre de archivo	Permite archivos de hasta 256 caracteres de cualquier tipo	Permite archivos de hasta 256 caracteres de cualquier tipo
	Tamaño máximo de archivo	Windows 2000 no tiene limitación de tamaño de archivo.	Máximo 2 Gb. Se eliminará la restricción en el kernel 2.4
Reverse Proxy		Windows 2000 no incluye Reverse Proxy, pero se puede incorporar con el MS Proxy Server	Esta capacidad viene incluida.

SEGURIDAD

Seguridad es el control de Acceso a datos de lectura o escritura.

A nivel hardware, existen tarjetas SSL para encriptación.

A nivel de paquetes de red lo da el firewall.

A nivel de objetos de red (archivos, servidores) lo da la red a través de Kerberos o PKI.

A nivel aplicación se da en certificados digitales y a través de Java.

Autenticación por SO	Control de password de usuario	Incluye todos los controles	Incluye todos los controles
	Base de datos de passwords	Tiene opción de encriptar el archivo de passwords completo.	Tiene opción de ocultar el archivo de passwords. Redhat pregunta al instalar si se utilizará "hidden passwords". Esta opción es altamente deseable ya que si no es relativamente fácil copiarse el archivo de passwords en unix a un disco local y así desencriptarlo con tiempo.
	Encriptación de password	Los password viajan encriptados	Los password viajan sin encriptar por default al ejecutar Telnet o FTP, pero se puede implementar Kerberos.
	Acceso de superusuario	El acceso de superusuario puede controlarse según desde donde es la conexión.	Se puede configurar para que el ingreso de root sea solamente desde consola mediante el archivo securetty. Permite el ingreso al sistema como usuario común y la posterior ejecución del comando "su", el cual pide el password de root. Para acceder a root, un hacker debe acceder primero a otro password.
Autenticación por red		Windows 2000 usa Kerberos 5.0 como su principal protocolo de autenticación. Esto le permitiría compatibilidad con otros servidores que utilicen Kerberos e integración absoluta a PKI	Se puede implementar PAM (Pluggable Authentication Module). También se puede implementar Kerberos. Además existen paquetes de terceros.
Encriptación de Filesystem		Windows 2000 provee granularidad en sus servicios de encriptación, ya que puede incluir desde archivos y directorios hasta filesystem	No tiene el método específico de Windows, pero se puede simular por ejemplo con los comando tar y crypt.
Autorización		El manejo de ACL(Lista de Control de Acceso) es muy completo En total tiene 27 derechos de usuario para asignar.	Implementa el mecanismo que funciona en Unix desde hace décadas. Consiste en asignar derechos de lectura, escritura y ejecución a cada archivo según usuario, grupo y otros. Mediante el comando suid, es posible permitir a un usuario que ejecuta una aplicación adquirir la personalidad del dueño de la aplicación. Este comando es muy poderoso, pero es responsable de muchos de los problemas de seguridad de los sistemas unix y linux. Existen implementaciones de ACL de terceros.
Confidencialidad	SSL 3.0 (Secure Socket Layer)	Incluido, con wizard	Incluido, con wizard

	Certificados digitales	<p>NT 4.0 y Windows 2000 incluyen un servidor de certificados para certificados digitales X.509, mapeables a cuentas de usuarios, y permite encriptación de 128 bits.</p> <p>Terceros que ofrecen también servidores de encriptación y claves de acceso : CheckPoint Software : VPN-1 Certificate Manager Entrust Technologies (Entrust PKI) Sun-Netscape Alliance : iPlanet Certificate Management System</p>	<p>Se puede implementar mediante productos gratuitos.</p> <p>Redhat ofrece CCVS, un producto para verificación de tarjetas de crédito, que incluye el código fuente en C, TCL, PERL, Java, PHP y otros.</p>
	VPN	Incluye un soporte excelente a VPN, a través de IPSec o PPTP	Incluye IPSec y SSL a través del cual se puede implementar VPN, en combinación con FreeS/WAN.
DetECCIÓN DE INTRUSOS		<p>Terceros que ofrecen sistemas de detección de intrusos:</p> <p>Intrusion.com : Kane Secure Enterprise Network Flight Recorder IDA Suite Network ICEpac Suite Tripwire Security Systems: Tripwire</p>	<p>Entre los productos gratuitos : Snort, el cual es popular y gratuito. PortSentry monitorea actividad en los puertos de TCP</p>
AUDITORÍA	De accesos de usuario	Incluido en el event monitor	Incluido en los archivos btmp, wtmp, sulog y syslog, y visibles con comandos de shell como last y lastb, ya que algunos son binarios.
	De archivos accedidos / modificados	Tiene esta capacidad.	No la tiene pero se puede implementar manualmente con el comando dic. Por otra parte existe software gratuito, como : LogCheck que compara los archivos de logs.
	Configuración de nivel de auditoria	Posee un completo conjunto de posibilidades de auditoria.	Sus posibilidades de configuración son limitadas.
VIRUS		<p>Es vulnerable a ataques de virus si el operador usa el sistema desde consola. Especialmente si usa el Outlook. Pero existe gran cantidad de software antivirus de terceros.</p> <p>Un virus en Windows 2000 puede ejecutarse por booteo de diskette o por ejecución de un programa infectado.</p>	No existen virus propiamente dichos en Linux, porque éstos están diseñados particularmente para ambientes Windows.

VERSATILIDAD

Evaluación de la variedad de servicios y tecnologías que el sistema le ofrece al administrador del servidor, así como al conjunto de los usuarios de la organización (LAN) y de internet.

Código Abierto vs. Propietario		Windows NT tiene 16 millones de líneas de código. Windows 2000 tiene 40 millones de líneas de código. MS reveló 1 bug por semana. A los 6 meses de salir el Windows 2000 salió el Service Pack 1.	Sus problemas de seguridad son rápidamente resueltos por la comunidad Linux, pero suelen encontrarlos primero los hackers.
Plataformas	Requerimientos mínimos	Windows 2000 requiere como mínimo 128 MB de RAM y una Pentium II	Linux puede ejecutarse en una máquina X386 con 32 MB de RAM.
	Variedad	NT soporta SGI Virtual Workstation, PC compatibles Pentium y Alpha Servers. Windows 2000, está orientado a Intel Microsoft decidió no soportar más el procesador Alpha	La diversidad de plataformas es uno de los puntos sobresalientes de Linux con respecto a cualquier sistema operativo. Además de X86 (incluyendo 386), Linux se puede instalar sobre las Sparc y Ultra Sparc de 64 bits también. También se puede ejecutar en SGI Virtual Workstation, Apple 68, Alpha Servers y Power PC Linux se está desarrollando para todo tipo de hardware, desde palmtop (ya hay versiones disponibles) hasta AS400.
Kernel	Configuración	Microsoft implementa un modelo de micro kernel, con distintas capas asociadas, que son las que se modifican con interfaces estándar. No requiere la modificación de un kernel en sí ya que no posee un kernel monolítico sino que está dividido según funciones.	Tiene la capacidad de agregarle o quitarle componentes, sin rebootear. Esto es muy apreciado por la comunidad Linux, aunque requiere un alto grado de conocimientos.
Actualizaciones y parches del SO		Windows 2000 tiene un registro detallado de los parches y versiones del SO.	Para conocer los parches instalados del sistema no existe una herramienta genérica. Los subsistemas deben ser verificados uno por uno.

Administración de Software	Instalación de software	Windows 2000 tiene la interfase estándar de Windows	Redhat es el creador del formato de paquetes de software RPM (Redhat Packet Manager) que se extendió a otras distribuciones de Linux.
	Presentación de software instalado	Windows 2000 lleva un registro de los archivos que se instalaron en cada aplicación	RPM guarda información completa de las instalaciones realizadas.
Interfase de administrador en consola.	CLI	Permite acceso a un shell similar a DOS, con varios agregados. El shell de NT era pobre pero mejoró mucho. Está orientado a ejecutar comandos pero no a visualizar estados del sistema. Aunque Microsoft continúa criticando las CLI, Windows 2000 incluye un lenguaje de script muy rico, el Windows Scripting Host, versión 2.	La interfase CLI en unix se conoce como shell. Este es el modo estándar de Unix en general desde hace décadas. Por esto el shell de unix es muy rico. Permite múltiples accesos por consola con distintos usuarios, y acceder al login de otro usuario con el comando "su" desde una sesión de shell. Requiere amplio conocimiento de shell ya que la ayuda en línea asume que el administrador tiene experiencia.
	GUI	Es el modo estándar de Microsoft por excelencia, desde que se impuso el Windows 3.0 sobre el DOS. Su interfase además es unificada para todas las tareas, y similar en los distintos productos Microsoft y del mercado en general. Toda tarea administrativa de Windows 2000 puede hacerse por GUI.	El protocolo X Windows para unix acepta distintos tipos de interfases gráficas. Se puede usar idénticamente para acceso local o remoto. Redhat incluye Gnome y KDE entre otros X-Windows. Las interfases de administración son independientes entre sí, y su aspecto varía según el programador que la diseñó ya que las aplicaciones de linux son desarrolladas independientemente.
Interfase de administrador remoto	CLI	Windows 2000 si tiene acceso en modo carácter a través de telnet.	Su acceso remoto por Telnet tiene las mismas características que el acceso por consola. Además que se pueden realizar múltiples conexiones.

	GUI	Windows 2000 incluye el Terminal Server que ofrece una interfaz gráfica en el cliente de Terminal Server como si fuera una consola, integrado con el MMC, Microsoft Management Center.	Linux implementa el protocolo X-Windows, en el que se ejecutan comandos en el server con display local, o sea que lo que viaja no es la imagen de pantalla sino comandos para el cliente local de X-Windows. Permite múltiples sesiones.
Web browser / editor incluido		Incluye el MS Explorer integrado a la interfaz de usuario. Permite la instalación de Netscape	Incluye Netscape. No existe MS Explorer para otra plataforma que no sea Microsoft.
Ayuda en línea		La ayuda en línea de NT es pobre pero en Windows 2000 es mucho más detallada.	Existen varias formas de acceder a ayuda, como el comando "man" de Unix, o los famosos mini-howto que no vienen incluidos pero están ampliamente disponibles en internet y por lo tanto no toma en cuenta el contexto. Algunas cosas están muy documentadas y otras nada.
Soporte de WAN	ISDN	Incluye ISDN	Incluye ISDN
	ATM	Tiene soporte	Lo incluye desde el kernel 2.4
	Frame Relay	No incluye.	No incluye.
TCP/IP	Tunneling	NT soporta PPTP y también IPSec a través de Cisco. NT permite usar una conexión IP para transportar protocolos como IPX y NetBEUI además de IP dentro de IP	Linux permite solo IP dentro de IP, aunque se está desarrollando para otros protocolos que Linux si soporta como servidor de archivos. Mediante el paquete FreeS/WAN pueden implementarse múltiples VPN estables.
	IPv6	Microsoft dice Windows 2000 soporta el estándar IPv6, pero aún está en desarrollo.	Linux soporta el estándar IPv6 desde el kernel 2.0

SERVIDOR DE WWW
 Su finalidad es enviar páginas HTML, aunque también hay nuevos formatos como DHTML, XML, etc. Las páginas incluyen imágenes multimedia. En este caso se dice que un servidor de web provee de "contenido" en vez de páginas. El browser recibe páginas del servidor de web, y determina como mostrarlas, según el encabezamiento MIME.

Servidor de Acceso Remoto		Su RAS es completo. Soporta PPTP, L2TP e IPsec y conexiones multienlace	Su RAS es completo. El kernel 2.2 ofrece balance de carga para líneas seriales de una conexión multienlace.
Servicios de red	NNTP (Network News Transfer Protocol)	Incluido servidor y cliente	Incluido servidor y cliente
	DHCP	En Windows 2000, incluye alertas por límites de IP disponibles y estadísticas.	Si
	DNS (Domain Name Service)	El DNS de NT era poco confiable e inestable. Se supone que el Windows 2000 mejoró su DNS.	El DNS de linux es confiable y veloz.
	SNMP Simple Network Management Protocol	Si	Si
	FTP	Incluido	Incluye WU-FTP SERVER
Disponibilidad		IIS : No soporta Fault Tolerance ni clustering NT tiene Load Balancing al nivel Servidor de web Layer Windows 2000 soporta balance de carga en Application Layer.	Posee balance de carga.
Recuperación de sesión		Está incluido y es configurable	No es configurable
Escalabilidad.	Páginas estáticas	Tanto NT como Windows 2000 superan a Linux y Novell para servicio de páginas estáticas.	Linux/Apache es muy lento según benchmarks realizados por PC Magazine. Pero usando kHTTPd, la velocidad era 3 veces mayor para páginas estáticas con un solo procesador.

	Páginas dinámicas	Tiene una performance superior al promedio	Tiene una performance inferior al promedio
Virtual hosting		IIS tiene capacidad de miles de dominios.	Apache tiene capacidad de cientos de miles de dominios
Seguridad	Integración con SSL	IIS lo tiene integrado.	El secure Apache incluye ahora SSL como módulo interno.
	Certificados criptográficos	Incluye Servidor de certificados digitales criptográficos en el mismo SO Además puede generar certificados propios para autenticación de Intranet/VPN.	No incluye soporte de certificados en forma integrada pero sí a través de plug-ins. No incluido en Apache, pero sí en otros como Zeus, Stronghold y Roxen El SO de Redhat si tiene un servidor
Variedad de terceros		Incluido : Microsoft IIS Apache O'Reilly Software : Website Pro Sun-Netscape Alliance: iPlanet Web Server, ex Netscape Enterprise Server. Lotus Domino	Incluido : Secure Apache Gratis: Stronghold, basado en Apache, incorpora mayor seguridad Roxen Challenger Web Server : Particularmente adaptado para ecommerce, www.roxen.com Comercial: Zeus Technology :Zeus Web Server
Integración		ISS esta integrado a Win NT Directory Services, Active X para IIS : Soporte Transaccional Active X para NT : Active Server Pages (ASP) FrontPage, MS Transaction Server, Message Queue, Visual InterDev, Site Server.	Secure Apache incluye mod_perl incorporado, lo cual lo hace más veloz
HTTP 1.2		No	No

APLICACIONES

Productos de software incluidos en el SO o disponibles en el mercado que funcionan complementando el servidor web, para diseño, desarrollo y producción de páginas HTML, DHTML, XML, etc. en forma estática o dinámica.

XML		IIS 5 lo soporta	Apache lo soporta mediante plug-ins.
Análisis del Sitio		<p>NT incluye análisis de sitio en su MS Site Server</p> <p>Windows 2000 incluye también análisis de sitio</p> <p>Además , existe software de terceros como:</p> <p>Avesta: eWatcher HP: Firehunter Lucent: VitalSuite Network Associates : Distributed Sniffer WebTrends : ERS</p>	<p>Apache lleva log de hits y páginas accedidas</p> <p>Linux no tiene un monitoreo específico para Servidor de web incluido.</p> <p>Terceros: ERS (de WebTrends)</p>
Publicación de contenido		<p>Existe gran cantidad de soporte:</p> <p>Allaire: Spectra</p> <p>INSO: Dynabase</p> <p>Interwoven : Teamsite</p> <p>Netobjects : Authoring Server Suite</p> <p>Vignette : Storyserver</p> <p>MicroSoft : FrontPage, utiliza FPSE, FrontPage Sever Extensions</p> <p>También soporta WebDav</p>	<p>Gratis :</p> <p>Frontpage :</p> <p>Frontpage FPSE permite la ejecución de CGI en unix/linux.</p> <p>Existe una implementación gratuita para FrontPage Server Extensions en http://www.nimh.org/fpse.shtml.</p> <p>Existe software de terceros para Frontpage como por ejemplo CiliSoft que permite interactuar con la base de datos MySQL.</p> <p>Además se puede implementar el protocolo WebDav, que es estándar en Internet , que define las extensiones de http para herramientas distribuidas de colaboración en desarrollo de páginas, en www.webdav.org</p>

RECURSOS HUMANOS
Análisis de los costos de recursos humanos requeridos, separados como disponibles dentro de la empresa y en el mercado, y de las licencias requeridas.

Bases de Datos		<p>Windows 2000 es compatible con todas las principales bases de datos, como Oracle, Sybase e Informix</p> <p>También está integrado a MS SQL Server y MTS</p>	<p>Hay una variedad de bases de datos para Linux, incluyendo algunas de código abierto.</p> <p>Redhat incluye con PostgreSQL</p> <p>Redhat también ofrece una versión "Enterprise Edition" optimizada para Oracle8i, y certificada por Oracle, con algunas capacidades extra tales como archivos mayores a 2 Gb, acceso a 4 Gb de RAM, y refinamiento de controladores de hardware.</p> <p>Open Source : PostgreSQL 6.5, MySQL, db.Linux, Interbase</p> <p>Gratis : Informix SE</p> <p>DBMS comerciales: Oracle 8 (\$5000 / licencia), Informix Dynamic Server, IBM DB2, Sybase, Texpress.</p>
Licencias por usuario		<p>Las licencias de uso de NT pueden ser por usuario o por servidor.</p> <p>Según los casos conviene uno u otro.</p>	<p>Es ilimitado ya que es gratis.</p>
RRHH Externos	Soporte Técnico	<p>El soporte de SO de Microsoft se hace a través de agentes, con distintos niveles de soporte, desde el envío de un técnico al lugar físico del servidor hasta la consulta por Email o telefónica.</p> <p>La información puede encontrarse en sitios relacionados con http://www.microsoft.com/, como por ejemplo MSDN.</p> <p>Un buen lugar para configurar la seguridad de NT es: www.trustedsystems.com/downloads.htm</p> <p>También existen numerosos sitios de comunidades de usuarios.</p>	<p>Para Linux, existe soporte desde 1999 de las principales empresas de software (excepto Microsoft por supuesto). Pero gran parte de la comunidad Linux utiliza sitios abiertos a todos para intercambiar soluciones, como:</p> <p>www.slashdot.org,</p> <p>www.freshmeat.net,</p> <p>www.osnews.com,</p> <p>www.questionexchange.com, etc</p>

Tabla 2: Combinación de Herramientas para programar Aplicaciones Web.♦

HERRAMIENTAS DEL CLIENTE	HERRAMIENTAS DEL LADO DEL SERVIDOR						
	CGI	PERL	PHP	ASP	JSP	CFML	SERVLETS
HTML/DHTML/WML	Si	Si	Si	Si	Si	Si	Si
XML	Opcional	Opcional	Opcional	Si	Opcional	Opcional	Opcional
JAVASCRIPT	Opcional	Opcional	Opcional	No	Opcional	Opcional	Opcional
VBSCRIPT/JSCRIPT				Si			
ACTIVEX				Si		Si	
FLASH						Si	
APPLETS					Si		Si
HERRAMIENTAS DEL SERVIDOR	CGI	PERL	PHP	ASP	JSP	CFML	SERVLETS
CGI	Si	Si				Opcional	
PERL	Si	Si	Opcional	PerlScript en Linux			
PHP			Si				
ASP				Si		Componentes ASP	
JSP					Si	Etiquetas de JSP	
CFML						Si	
JAVA			Componentes Java		Si	Si, por medio de TAGS	Si
SERVLETS					Si	Si, por medio de TAGS	Si

♦ Fuente: www.jesusolivares.com/tabla_tecnologias.htm

Tabla 3: Servicios proporcionados en Internet y sus respectivos puertos.

PROTOCOLO	SERVICIO	PUERTO	EJEMPLO	DESCRIPCIÓN
FTP	Transferencia de Archivos	21	ftp://ftp.dccia.ua.es/pub/soft/winzip.exe	FTP es el protocolo que permite la transferencia de archivos en Internet, es posible enviar y recibir archivos desde o hacia un servidor FTP. Es parte del protocolo TCP
SMTP	Distribución de Correo	25	carlos@dccia.ua.es	El protocolo SMTP es el estándar para la distribución de correo electrónico en Internet. Es un protocolo orientado a texto que utiliza los servicios de TCP/IP para recibir correo desde un cliente y para transferir mensajes desde un servidor a otro de forma fiable.
HTTP	WWW	80	http://www.ua.es/index.htm	Es el protocolo de transferencia de Hipertexto que permite la existencia del servicio WWW.
POP	Correo	110	mailto:carlos@dccia.ua.es	El protocolo POP define el diálogo entre un servidor de correo y la aplicación de correo electrónico. Al recibir los mensajes el servidor de correo los almacena en buzones privados para cada usuario, para que puedan accederlos, descargarlos y luego borrarlos.
Telnet	Acceso Remoto	23	telnet://locator.ntm.nih.gov telnet host/IP	Telnet es un protocolo de terminal remoto sencillo, el cual permite a un usuario establecer una conexión TCP con un servidor de acceso a otro. La terminal se comporta como si el usuario estuviera sentado frente a la máquina remota. Es parte del protocolo TCP.
Gopher	Acceso a Directorios y Documentos	70	gopher://sunsite.uchile.cl/cec/readme.txt	Este protocolo permite acceder a un servidor Gopher que contiene información de manera jerárquica similar a un árbol de directorios con subdirectorios y archivos desplegados como enlaces. Gopher es anterior al servicio WWW(http://) y tiene la misma funcionalidad que los enlaces de WWW, mediante los que se pueden hacer búsquedas.
Finger	Búsqueda de usuarios, grupos y servidores	79	finger johnc@idsoftware.com	Muestra información de usuarios o grupos de usuarios de un sitio de la Red. Se usa para saber si una persona tiene cuenta en un sitio de Internet. Muestra información acerca de un usuario, o de todos los usuarios, que han iniciado una sesión en el sistema local o en un sistema remoto. Normalmente muestra el nombre completo, la última vez que se conectó el usuario, el tiempo de inactividad, y la ubicación del terminal (cuando sea aplicable).
NNTP	Acceso a artículos de grupos de noticias.	119	news:alt.dev.null	Se utiliza para intercambiar mensajes de grupos de noticias entre servidores de noticias. Permite ingresar a un servidor de noticias para acceder a un grupo de noticias especificado.

Tabla 4: Evaluación de Servidores Web gratuitos y comerciales que trabajan sobre WINDOWS / LINUX y soportan: PHP/ASP/JSP.

	APACHE WEB SERVER	ZEUS WEB SERVER	INTERNET INFORMATION SERVER	PERSONAL WEB SERVER	TOMCAT WEB SERVER	iPLANET WEB SERVER
EMPRESA	Apache Software Foundation	Zeus Technology	Microsoft Corporation	Microsoft Corporation	Apache Software Foundation (Proyecto Jakarta)	Sun Microsystems
SITIO WEB	http://www.apache.org	http://www.zeus.com	http://www.microsoft.com	http://www.microsoft.com/iis	http://jakarta.apache.org	http://www.iplanet.com
TIPO DE LICENCIA	Gratuita bajo GPL [^]	Gratuita por 30 días Comercial por \$1.700	Comercial Incluida: Windows 2000, XP Option Pack 4	Gratuita con Option Pack 4 y con Windows 98	Gratuita bajo GPL	Gratuita por 60 días Comercial por \$1.495
CÓDIGO FUENTE	Open Source	No	No	No	Open Source	No
VERSIONES	1.x.x y 2.x.x	1.x, 2.x, 3.x y 4.x	1.0, 2.0, 3.0, 4.0 ,5.0,6.0	1.0, 2.0, 3.0, 4.0	3.x.x, 4.x.x, 5.x.x	4.x, 5.x, 6.x
PLATAFORMAS SOPORTADAS	Alpha Intel	Alpha Intel SPARC	Alpha Intel	Alpha Intel	Alpha Intel	Alpha Intel SPARC
SISTEMAS OPERATIVOS	Unix Linux Windows 9.x, NT, 2000	Linux Digital Unix Solaris	Windows NT Win 2000 Prof. y Server	Windows 95,98 Windows NT,	Linux Windows NT, 2000	Windows NT Solaris

[^]GPL(General Public Licence – Licencia Pública General). Licencia para productos de código abierto.

	OS/2, AS/400 Netware 5.x	IBM AIX Mac OS	Win XP Profesional y Server Windows Me Profesional AS/400, OS/2	Win 2000, XP, ME (ediciones Home)	Solaris	IBM AIX Linux HP-UX
INTERFAZ DE INSTALACIÓN	Línea de comandos Vía RPM en Red Hat Linux Gráfica en Windows	Gráfica	Gráfica: • Instalación Automática • Mediante Panel de control	Gráfica, mediante wizards.	Línea de comandos	Gráfica
INTERFAZ DE CONFIGURACIÓN	Línea de comandos (Archivos de configuración) Instalando APACI*	Línea de comandos	Gráfica, mediante wizards	Gráfica, mediante wizards	Línea de comandos (Archivos de configuración)	Gráfica Mediante páginas web basadas en Java
FACILIDAD DE USO	Algo complicado (Administrable por medio de páginas web) (Configurable por medio de archivos)	Complicado	Fácil	Muy Fácil	Algo complicado al configurar	Difícil de configurar
RENDIMIENTO*	Muy bueno: 4,5	Muy bueno: 4,5	Muy bueno: 5	Bueno: 4	Moderado: 3,8	Muy bueno: 4,5
SOPORTE TÉCNICO	<ul style="list-style-type: none"> • En línea vía grupo de noticias • FAQs 	En línea	<ul style="list-style-type: none"> • En línea. • Bajo documentación. • Personal calificado. 	<ul style="list-style-type: none"> • En línea. • Bajo documentación. • Personal calificado. • Ayuda en navegador 	En línea	En línea
VENTAJAS	<ul style="list-style-type: none"> • Extensible mediante módulos API. • Servidor web más usado en Internet. • Apoyo de la comunidad Internet. • Trabaja con bitácoras. • Es menos pesado que 	<ul style="list-style-type: none"> • Maneja eficientemente un alto volumen de tráfico. • Soporta clusters. • Cuarto servidor más usado. • Trabaja con balanceo de carga. 	<ul style="list-style-type: none"> • Certificados de seguridad integrados. • Mantenimiento remoto. • Segundo servidor web más usado. • Permite publicación de páginas mediante 	<ul style="list-style-type: none"> • Fácil de configurar y usar. • Puede integrarse con FrontPage para diseño y publicación de sitios. • Cuenta con algunas características de IIS. • Soporta ODBC y 	<ul style="list-style-type: none"> • Es extensible mediante módulos. • Puede instalarse como plug-in para trabajar junto con otros servidores. • Implementación oficial para JSP y Servlets. 	<ul style="list-style-type: none"> • Tercer servidor más usado. • Soporta las herramientas Chili!Soft y Forte for Java. • Soporte para manejo de algunas bases de datos.

* Es una interfaz para configuración, esta característica fue añadida desde la versión 1.3 de Apache Web Server. Se instala por separado.

* Fuente: www.serverwatch.com, sitio que se encarga de evaluar el rendimiento de servidores web.

	otros servidores	<ul style="list-style-type: none"> • Cuenta con versión de prueba gratis. • Puede integrarse con módulos propios o de terceros mediante APIs. 	<ul style="list-style-type: none"> • FrontPage. • Permite indexación. • Soporta controladores ODBC • Se integra con herramientas adicionales de Microsoft • Puede ser servidor de: FTP, NNTP, SMTP 	ADO.	<ul style="list-style-type: none"> • Consume pocos recursos. • Puede trabajar con otros servidores como: Apache e IIS. 	<ul style="list-style-type: none"> • Cuenta con una versión gratuita de prueba: FastTrack • Admite J2EE, SOAP, XML, WAP, SNMP, LDAP y CORBA. • Permite restaurar configuraciones anteriores
DESVENTAJAS	<ul style="list-style-type: none"> • No hay versión para Mac OS. 	<ul style="list-style-type: none"> • Alto costo. • Difícil de configurar. • No hay versión para Windows. 	<ul style="list-style-type: none"> • Sólo existen versiones para Windows 2000, XP en ediciones Profesional y Server. • Soporta en su mayoría productos Microsoft. • Vulnerable a ataques de virus. • Soporta menos herramientas para generación de páginas dinámicas. 	<ul style="list-style-type: none"> • Sólo se instala en Win 95, 98, ME, 2000, XP (en ediciones Home) y en Win NT. • Sólo para sitios web pequeños/personales. • Características reducidas. • Más lento que IIS. • Versiones menores a 4.0 requieren asp.exe para ASP. 	<ul style="list-style-type: none"> • No es configurable en algunos aspectos. • Es lento para servir páginas estáticas. 	<ul style="list-style-type: none"> • A partir de la versión 4.1 no es soportada en Linux. • Sólo soporta ambientes Java
SITIOS WEB QUE USAN EL SERVIDOR	<ul style="list-style-type: none"> • Javasoft • The FBI • Financial Times • The Movies Database • W3 Consortium • The Royal Family • M.I.T. • Oxford University • Harvard University • The University of Texas 	<ul style="list-style-type: none"> • eBay • Lycos • Infoseek • Blue Mountain • TD Waterhouse 	<ul style="list-style-type: none"> • Microsoft Corporation • Corel • Web Master • Allaire Corporation • SQL Soft • Action Technologies Inc. • Corporate Computer • FACT Software 	Sólo sitios personales	Incluido en el uso de Apache, Netscape e IIS	Portal Terra de España
REQUERIMIENTOS DE	<ul style="list-style-type: none"> • S.O: Linux o Win • Compilador ANSI C (Instalar mediante código fuente) 	<ul style="list-style-type: none"> • S.O: Linux 	<ul style="list-style-type: none"> • S. O.: Win NT/ ME/ XP/ 2000. • Option Pack 4 (Win NT) 	S.O: Win 95/ 98 /NT	<ul style="list-style-type: none"> • S.O: Win o Linux • JDK 1.2 o superior 	<ul style="list-style-type: none"> • Linux Red Hat 6.x o + <ul style="list-style-type: none"> • kernel 2.2.5 • glibc 2.1.2 • Nestcape

SOFTWARE (Para plataforma más utilizada)			<ul style="list-style-type: none"> • Service Pack 3 o superior • Internet Explorer 4 o superior 			<ul style="list-style-type: none"> • Comunicator y Navigator • JDK y SDK
REQUERIMIENTOS DE HARDWARE RECOMENDADO (Para plataforma más utilizada)	<ul style="list-style-type: none"> • Procesador: 586 o superior • Espacio en disco: 15MB • Memoria: 8MB. 	<ul style="list-style-type: none"> • Procesador: 586 o superior 	<ul style="list-style-type: none"> • Procesador: Pentium a 200 MHz. • Memoria: 128MB. • Espacio en disco: 200MB • Monitor: SVGA • CD-ROM: 6X 	<ul style="list-style-type: none"> • 31 MB libres • Procesador: 586 o superior 	<ul style="list-style-type: none"> • Procesador: Pentium • Memoria: 64MB. • Espacio en disco: 50MB 	<ul style="list-style-type: none"> • 34.2 MB libres para instalar • 128 MB RAM • 110 MB en disco • 256 MB swap
TRABAJA COMO SERVIDOR DE APLICACIONES	No	No	Si, Usando MTS*	Si, Usando MTS	Si	Si, usando: <ul style="list-style-type: none"> • iPlanet Trustbase Transaction Manager. • ChiliSoft ASP 3.4 • iPlanet Web Server Web Application Interface.
TRABAJA COMO SERVIDOR PROXY DIRECTAMENTE	Si	Si	No, pero cuenta con soporte para proxy	No	Si	No
PUEDE REQUIR AUTENTICACIÓN DE USUARIOS	Si	Si	Si También permite acceso anónimo	No	No	Si, grupos y usuarios
PUEDE ESCRIBIR EN MULTIPLES LOGS	Si	Si	Si	Si	Si	Si

* MTS (Microsoft Transaction Server). Servidor para el desarrollo y gestión de aplicaciones web de Microsoft que hagan uso transaccional de componentes.

INCLUYE AGENTES SNMP	Instalando mod_snmp	Si	Si	No	Con JBOSS	Si
MANTENIMIENTO BASADO EN INTERFAZ GRÁFICA	Instalando APACI	Si	Si Mediante la Consola de Administración Microsoft	Si Mediante Internet Service Manager	No	Si, mediante páginas web
CONFIGURABLE MEDIANTE:	Archivos .conf	Archivos .conf	Mediante Administrador de Servicios Internet	<ul style="list-style-type: none"> Administrador de Servicios Internet Personal Web Manager 	Archivos conf Archivos .xml	Páginas web, usando: <ul style="list-style-type: none"> Nestcape Comunicator 4.6.1 o superior Archivos .conf
PERMITE CONFIGURAR:	<ul style="list-style-type: none"> Tipo de servidor Nombre del servidor Correo de administrador Tiempo de espera de conexión Conexiones persistentes Puerto(s) Usuario y grupo Servidores virtuales Directorio del servidor Directorio de publicación WWW Tipo de documento predeterminado Carga de módulos Número máximo de conexiones Directorio para scripts del servidor Directorio CGI y SSI Trabajar como Proxy Páginas de error Directorios de logs Tipos MIME que 	<ul style="list-style-type: none"> Puerto por defecto Como trabaja SNMP Nivel de seguridad 	<ul style="list-style-type: none"> Nombre del servidor Puerto Tiempo de espera de conexión Número máximo de conexiones Inicio de sesión Directorio del servidor Directorio de publicación WWW Directorio FTP y Gopher Servidores virtuales Tipo Documento predeterminado Ficheros log Interfaz CGI e ISAPI Número de visitas por día Ancho de banda Control de autenticación 	<ul style="list-style-type: none"> Nombre del servidor Puerto Directorio de publicación Documento predeterminado Interfaz CGI e ISAPI Acceso remoto 	<ul style="list-style-type: none"> Nombre del servidor Puerto Puerto para otro servidor web Directorio de publicación WWW Directorios virtuales 	<ul style="list-style-type: none"> Nombre del servidor Usuario y grupo para ejecutar el servidor Usuario y grupo para las páginas web Puertos TCP para web y para administrar Path de instalación Nombre para identificar el servidor Directorio de publicación WWW Tipo de servidor Servidores virtuales Configurar soporte LDAP Configurar CGI Directorio de logs Páginas de error Tipos MIME Activar JSP y Servlets

	soporta					
INCLUYE SU API PROPIA	Si	Si	Si	Si	Si	Si
CUENTA CON SERVIDOR DE CERTIFICADOS	No	No	Si	No	No	Si
INTEGRA MOTOR DE BÚSQUEDA	No	Si	Index Server	No	No	Directorio de Indexación
POSIBILIDAD DE MANTENIMIENTO REMOTO	No	Si	Si, mediante: <ul style="list-style-type: none"> • Microsoft Management Console • Terminal Server • Mediante páginas web 	No	Si	Si
SOPORTA ISAPI/NSAPI	No	Si, ISAPI	Si, ISAPI	Si, ISAPI	No	Si, NSAPI
SOPORTA IPv6	Si	No	Si (Sólo IIS 6.0)	No	No	No
SOPORTA SNMP	No	Si	No	No	No	Si
SOPORTA HTTP/1.1	Si	No	Si	Si	Si	Si
SOPORTA SERVIDORES VIRTUALES	Si	Si	Si	No	Si	Si

INTEGRA LENGUAJES DE SCRIPT O MÓDULOS	mod_php mod_perl Python SSI(Server Side Include)	No	ASP.DLL ISAPI	ASP.EXE(versiones < 4.0) ASP.DLL	Moror de Servlets	Moror de Servlets
LENGUAJES DE SCRIPTS SOPORTADOS	PHP PERL JSP(versiones antiguas de Apache) CFML SSI, Python CGI, FastCGI ASP con ChiliSoft ASP	PHP JSP y J2EE ASP PERL CGI, FastCGI	ASP ASP.NET(IIS versión 5.0 y +) PHP(como CGI o DLL) JSP CFML CGI, FastCGI, ISAPI SSI (IIS versión 6.0)	ASP PHP(como CGI o DLL) JSP CFML	JSP Java Servlets SSI CGI	JSP Java Servlets PHP como NSAPI (versiones nuevas)
SOPORTA SSL [◇]	Si	Si	Si	No	Si	Si
SOPORTA WebDAV [◦]	Si, con mod_dav	Si	Si	Si	Si	Si
PERMITE PERSONALIZAR PÁGINAS DE ERROR	Si	Si	Si	Si	Si	Si
VULNERABLE A VIRUS	No	No	Si	Si	No	No

◇ SSL(Secure Socket Layer – Capa de Servidor Seguro). Protocolo de seguridad.

◦ WebDAV (Web-based Distributed Authoring and Versioning). Es un estándar para autorizar a múltiples usuarios construir documentos y publicarlos en Intranets o Internet.

Tabla 5: Directivas de configuración del archivo httpd.conf de Apache Web Server (Ejemplos en Linux)

DIRECTIVA	DESCRIPCIÓN	EJEMPLOS
ServerType	Indica como puede ser arrancado el servidor, como demonio individual(standalone) o dentro del conjunto de demonios(inetd).	ServerType standalone
ServerRoot	Directorio raíz donde residen los fuentes del servidor, la configuración del servidor, los archivos de logs y los de error.	ServerRoot "/usr/local/apache"
Port	Especifica el número de puerto por el que pueden llegar las solicitudes al servidor. Se debe tener permisos de superusuario.	Port 80
User	El usuario con el que se ejecutarán los procesos del servidor. Si es diferente del usuario que lanza el demonio, el usuario que lanza el demonio debe de ser root.	User nobody
Group	El grupo con el que se ejecutarán los procesos del servidor.	Group nobody
HostnameLookups	Traduce la IP del cliente. Puede estar en On u Off. Si está en on traducirá la IP, pero esto consume recursos.	HostnameLookups Off
ServerAdmin	E-mail del usuario administrador del servidor Apache. A este e-mail se enviarán mensajes en caso de errores.	ServerAdmin admin@www.dominio.com
ServerName	Nombre del dominio, de la máquina o dirección IP. Debe ser un nombre real registrado en el sistema DNS.	ServerName www.dominio.com
BindAddress	Puede hacer que el servidor escuche en varias direcciones IP, es decir establece dominios virtuales.	BindAddress www.dominio1.com
ErrorLog	Establece el archivo donde se almacenan los errores generados de peticiones de clientes.	ErrorLog /usr/local/apache/logs/error_log

TransferLog	Establece el archivo donde se almacenan las transferencias de datos con los clientes.	TransferLog /usr/local/apache/logs/access_log
PidFile	Indica el archivo donde se almacena el pid (identificador del proceso del servidor).	PidFile /usr/local/apache/logs/httpd.pid
ScoreBoardFile	Indica el archivo donde se almacena información del estado del servidor. Permite analizar lo que ocurre al interior del servidor.	ScoreBoardFile /usr/local/apache/logs/apache_status
TimeOut	Tiempo en segundos tras el cual se envía o recibe el fin de plazo de una petición.	TimeOut 300
KeepAlive	Permite la existencia de conexiones persistentes, es decir que por una misma conexión se puedan enviar varias peticiones HTTP.	KeepAlive On
MaxKeepAliveRequests	Especifica el número máximo de solicitudes que se permite enviar en una misma conexión persistente. El valor 0 permite un número ilimitado de solicitudes.	MaxKeepAliveRequests 100
KeepAliveTimeout	Fija el tiempo máximo en segundos que el servidor espera entre solicitudes de un mismo cliente en las conexiones persistentes antes de cerrar la conexión.	KeepAliveTimeout 15
MaxClients	Limita el número total de procesos del servidor ejecutándose y, por tanto, el número de clientes que pueden conectarse simultáneamente. Debe ser un número alto para que no se bloquee el servidor.	MaxClients 150
MaxRequestsPerChild	Determina el número máximo de solicitudes que puede atender un mismo proceso hijo antes de morir. Para eliminar esta limitación debe asignarse el valor 0.	MaxRequestsPerChild 0
ProxyRequests*	Especifica si el servidor web actuará también como proxy. Permite que se guarde una caché con las páginas más accedidas por los clientes.	ProxyRequest On
CacheRoot	Indica el directorio para el servidor proxy.	CacheRoot /usr/local/apache/proxy
ScriptAlias	Indica el directorio o directorios donde se encuentran los scripts del servidor.	ScriptAlias /php/ "/usr/local/php/"

* Si ProxyRequest es activado, diversas opciones sobre la caché deben ser indicadas: tamaño de caché, cuando expiran los documentos, de que URL no se debe hacer caché, etc.

AddType	Define las extensiones que Apache utilizará para saber cuando ejecutar un script PHP. Se puede definir una o varias extensiones.	AddType application/x-httpd-php .php .phtml AddType application/x-httpd-php-source .phps
Action	Permite definir el parser que el servidor ejecutará cada vez que se encuentre con un script PHP.	Action application/x-httpd-php "/usr/local/php"
DocumentRoot	Establece la raíz del árbol de directorios que serán accesibles a través del servidor web. En estos directorios se encontrarán las páginas web a ser publicadas.	DocumentRoot "/var/www/html"
DirectoryIndex	Indica cuales serían los archivos que abrirá Apache por defecto. Estos archivos son las páginas web que se deben desplegar por defecto.	DirectoryIndex index.html index.shtml index.php

GLOSARIO

ADO	(ActiveX Data Object – Objeto de datos ActiveX) Conjunto de interfaces de acceso a datos basadas en objetos optimizados para las aplicaciones basadas en Internet y centradas en datos. ADO se incluye con Internet Information Server y con Microsoft Visual InterDev.
APACI	(Apache AutoConf Interfaz – Interfaz Apache al estilo Autoconf) Interfaz de Apache que simplifica considerablemente el procedimiento de instalación y configuración.
API	(Application Programming Interface – Interfaz de Programación de Aplicaciones) Conjunto de rutinas, protocolos y herramientas que definen como se invoca un método o servicio desde un programa.
ASP	(Active Server Page – Página Activa de Servidor) ASP es una herramienta de generación de contenido dinámico del lado del servidor desarrollada por Microsoft, permite la creación de páginas .asp para ser ejecutadas por Internet Information Server. Los scripts de las páginas ASP pueden ser escritos en VBScript, JScript o PerlScript e incrustarse en HTML.
CGI	(Common Gateway Interface – Interfaz de Compuerta Común) Es un protocolo creado por NSCA para establecer un mecanismo estándar de comunicación entre un servidor web y un lenguaje de programación, de tal forma que desde aplicaciones programadas en dicho lenguaje, el servidor web pueda procesar peticiones de usuario.
ChiliSoft ASP	Es el software que permite integrar Apache y ASP en Windows y en Linux. Cuenta con licencias: gratuita y comercial.
COM	(Component Object Model - Modelo de Componentes de Objetos) Modelo de componentes de objetos desarrollado por Microsoft.
Cookie	Es una pieza de información que el servidor web envía al cliente para que se guarde en su disco duro. Cuando el cliente vuelve al sitio web reenvía el cookie al servidor y de esa forma éste puede distinguir a los usuarios y hacerles un seguimiento individual.
CORBA	(Common Object Request Broker Architecture – Arquitectura de Corredor Común para la Solicitud de Objetos) Es una arquitectura y especificación estándar para crear, distribuir y administrar objetos de programas distribuidos en una red.
CSS	(Cascade Style Sheet – Hoja de Estilo en Cascada) Es un conjunto de especificaciones que definen o redefinen el comportamiento de una etiqueta en un documento HTML.
Cyg/PC	Es una utilidad requerida para correr PostgreSQL en Windows.
Cygwin	Es una "capa de emulación que expone suficiente funcionalidad del API de Unix", en otras palabras es una API compuesta de una DLL(cygwin1.dll) y varias herramientas que traen la mayoría de las funciones de Unix para poder correr programas de Unix en Windows mediante una consola MS-DOS.

DHTML	(Dynamic HTML – HTML Dinámico) Es en realidad la versión 4.0 de HTML que incorpora nuevas herramientas o tecnologías web como: JavaScript y Hojas de Estilo en Cascada para mejorar las limitaciones de HTML.
DLL	(Data Link Library – Librería de Enlace Dinámico)
DNS	(Data Name System – Sistema de Nombres de Dominio) Sistema para convertir los nombres de los servidores en las direcciones IP respectivas y viceversa.
DSO	(Dynamic Shared Object – Objeto Dinámico Compartido) Son módulos para ser soportados como librerías de objetos compartidos. Se instalan en Linux.
EJB	(Enterprise Java Beans – Componentes Java Empresariales) Es un modelo de componentes específico de una arquitectura basada en Java para construir componentes transaccionales del lado del servidor. Los EJB encapsulan la lógica del negocio, tal como el acceso a bases de datos, seguridad e integridad de las transacciones.
FastCGI	Es la alternativa mejorada a CGI.
Flash	Es un lenguaje estándar e independiente del navegador para crear animaciones de gran calidad para ser usadas en las páginas web. Para la visualización de animaciones Flash es necesario tener instalado el plug-in correspondiente.
FTP	(File Transfer Protocol – Protocolo de Transferencia de Archivos) Protocolo que permite a un usuario transferir archivos de una ubicación a otra a través de Internet.
GET	Método de envío del protocolo HTTP. Las variables con sus respectivos valores son pasados al servidor web a través de la URL.
HEAD	Método de envío del protocolo HTTP.
GOPHER	Es un protocolo que permite a los clientes acceder a un servidor Gopher que contiene información almacenada de forma jerárquica similar a un árbol de directorios con subdirectorios y archivos desplegados como hipervínculos.
HTML	(HyperText Markup Language – Lenguaje de Marcas de HiperTexto) Es el lenguaje estándar para crear documentos estructurados con formato de páginas web. Está basado en marcas o etiquetas que definen la presentación de las páginas.
HTTP	(HyperText Transfer Protocol – Protocolo de Transferencia de HiperTexto) Protocolo que entrega información a través de WWW. Por medio de una URL permite recuperar hipertexto (texto, gráficos, sonido, etc), desde un servidor web.
ICAP	(Internet Content Adaption Protocol – Protocolo de Adaptación de Contenido de Internet)
IMAP	(Internet Message Access Protocol – Protocolo de Acceso a Mensajes en Internet) Protocolo diseñado para el manejo del correo electrónico. Es un poco más sofisticado que POP, ya que permite que el usuario vea los encabezados de los e-mails directamente en el servidor remoto (sin "bajarlos").
InstantASP	Software para soportar páginas de tipo ASP en Novell.

IP	(Internet Protocol – Protocolo Internet)
IPV6	Protocolo IP versión 6.
ISAPI	(Internet Server API – API del Servidor Internet Information Server) Interfaz de programación de aplicaciones del Internet Information Server.
JavaBeans	Componentes desarrollados en Java que pueden ser insertados en páginas JSP.
JavaScript	Lenguaje de programación del lado del cliente desarrollado por Netscape. Pertenece a la familia del lenguaje Java, pero a diferencia de éste último se incrusta en HTML para ser interpretado por los navegadores.
JDK	(Java Development Kit – Kit de Desarrollo Java) Conjunto de herramientas que permiten el desarrollo de: aplicaciones, applets, servlets, páginas JSP, JavaBeans, etc, por medio del lenguaje de programación Java.
JDBC	(Java Data Base Connectivity – Conectividad de Bases de Datos con Java) Es una API Java incluida en el JDK que permite la conexión con bases de datos mediante Java.
JScript	Lenguaje similar a JavaScript, desarrollado por Microsoft.
JSP	(Java Server Pages – Páginas Java de Servidor) JSP es una herramienta de generación de contenido dinámico del lado del servidor desarrollada por Sun, permite la creación de páginas .jsp para ser ejecutadas por Servidores web compatibles. Los scripts de las páginas JSP están escritos en Java y pueden ser incrustados en HTML o almacenarse en archivos independientes.
JVM	(Java Virtual Machine – Máquina Virtual de Java) Es el componente que se encarga de verificar los bytecodes generados al compilar programas desarrollados en Java. Por lo general la JVM viene con los navegadores.
J2EE	(Java 2 Enterprise Edition – Java 2 Edición Empresarial) Kit de desarrollo de Java edición empresarial. Cuenta con herramientas como JDK, EJB, etc.
LDAP	(Lightweight Directory Access Protocol – Protocolo Ligero de Acceso a Directorios) Protocolo que permite ubicar organizaciones, individuos y otros recursos tales como archivos y dispositivos en una red, ya sea en Internet o en una Intranet.
MIME	(Multipurpose Internet Mail Extensions – Extensiones Multipropósito para el Correo de Internet) Es un estándar oficial de Internet que especifica cómo se deben formatear los mensajes para que puedan ser intercambiados entre diferentes sistemas de correo electrónico. MIME es un formato muy flexible, que permite incluir cualquier tipo de dato (texto, audio, video, imágenes, datos de aplicaciones, etc.) en los mensajes de correo.
MMC	(Microsoft Management Console – Consola de Administración Microsoft). La Consola de Administración de Microsoft es donde se puede realizar algunas tareas de administración del Sistema Operativo y de otros programas como el Internet Information Server.
mod_jk	Módulo para integrar el servidor web Apache con Tomcat.

MTS	(Microsoft Transaction Server – Servidor de Transacciones de Microsoft)
NIC	(Net Information Center – Centro de Información de Red) Es el organismo encargado de asignar las direcciones IP en Internet.
NNTP	(Network News Transport Protocol – Protocolo de Transporte de Noticias en Red) Los servidores de noticias hacen uso de este protocolo.
NSAPI	(Netscape API – API de Netscape) Interfaz de programación de aplicaciones del Netscape Server.
ODBC	(Open Data Base Connectivity – Conectividad Abierta a Bases de Datos) Es una interfaz que permite a las aplicaciones acceder a los datos de un Sistema de Base de Datos usando SQL.
OLE-DB	(Object Linking And Embedding Data Base – Objetos Vinculados e Incrustados a Base de Datos) Es un API escrita en C++ que utiliza interfaces de manera eficiente para acceder, no sólo a datos, sino a funciones y objetos, se trate o no de una fuente de datos tradicional.
PERL	(Practical Extraction and Report Language – Lenguaje Práctico de Extracción y Reportes) Lenguaje de programación (interpretado para Unix, compilado para Windows) utilizado principalmente para programar aplicaciones de tipo CGI.
PHP	(PHP Hypertext Preprocessor – Preprocesador de Hipertexto PHP) Lenguaje de script interpretado e incrustado en páginas HTML, se ejecuta del lado del servidor. Está diseñado para facilitar el rápido desarrollo de sitios y aplicaciones web dinámicas. Su funcionamiento es similar a ASP, JSP y ColdFusion.
POP	(Postal Office Protocol – Protocolo de Oficina Postal) Protocolo de correo electrónico.
POST	Método de envío del protocolo HTTP. Las variables con sus respectivos valores son pasados al servidor web por el método Submit de los formularios.
Python	Lenguaje de scripts interpretado, generalmente usado para programar aplicaciones CGI.
RAID	(Redundant Array Disk – Array Redundante de Discos)
RDO	(Remote Data Object – Objeto de datos Remotos) Conjunto de interfaces de acceso a datos similar a ADO.
RPM	(Red Hat Package Manager – Administrador de Paquetes Red Hat) Es el administrador de paquetes del sistema Red Hat Linux, viene junto al sistema operativo y es análogo a Agregar o quitar programas en Windows.
Servlets	Los Servlets son la respuesta de la tecnología Java a la programación CGI. Son programas Java que se ejecutan en un servidor web y construyen páginas web dinámicamente. Son los antecesores de JSP.
SMTP	(Server Mail Transfer Protocol – Protocolo de Transferencia de Correo)

Principalmente permite establecer conexiones para la transmisión de correo , facilita la inclusión de correo en formularios de páginas web, dichos formularios son procesados por medio del servidor de SMTP en lugar de ser procesados directamente con un CGI o script del servidor web.

SSI	(Server Side Includes – Inclusiones del Lado del Servidor) Puede considerarse como un lenguaje sencillo de directivas para inserción de datos en páginas web. El propósito principal es incrustar directivas dentro del HTML, que permitan desplegar el contenido de archivos, información dinámica o la salida resultante de la ejecución de ciertos comandos del sistema.
SSL	(Secure Socket Layer – Capa de Sockets Seguros) Protocolo posterior al TCP/IP, creado por Netscape, posibilita la transmisión cifrada y segura de información a través de la red.
TCP/IP	(Transmission Control Protocol/Internet Protocol – Protocolo de Control de Transmisión/Protocolo Internet)
Telnet	Protocolo estándar de Internet para iniciar sesiones en ubicaciones remotas.
TLS	(Transport Layer Security – Capa de Transporte Seguro) Este protocolo se basa en SSL, y es empleado para encriptaciones a nivel más bajo que el de aplicación, es decir, por debajo del HTTP utilizado para la visualización de las páginas Web.
VBScript	Lenguaje similar a JavaScript usado sólo por plataformas Microsoft. Se deriva de Visual Basic.
WebDAV	(Web-based Distributed Authoring and Versioning) Es un estándar para autorizar a múltiples usuarios construir documentos y publicarlos en Intranets o Internet.
WML	(Wireless Markup Language – Lenguaje de Marcas Inalámbrico) Lenguaje de marcas similar a HTML, utilizado para generar páginas web para dispositivos inalámbricos.
XML	(eXtended Markup Language – Lenguaje de Marcas Extendido)

BIBLIOGRAFÍA

INTERNET

PHP

http://www.mexicoextremo.com.mx/ayuda/manuales/Como_instalar_PHP_en_RedHat.html

<http://www.planetalinux.com.ar/article.php?aid=43>

<https://www.mexicoextremo.com.mx/tienda/cursos.php#php>

<http://www.ciberteca.net/webmaster/php/index3.asp>

<http://www.openresources.com/es/index.html> Instalacion de PHP

ASP

<http://www.eitig.com/lomasweb/cursillo/ASP.htm>

<http://www.soloasp.com.ar>

http://www.asp-hosting.netregistry.com.au/asp_adding_postgres.html

JSP

<http://www.javahispano.org/tutoriales/jsp/parte4.htm>

<http://www.javahispano.com/tutoriales/apachetomcat4.pdf>

http://java.programacion.net/servlets_jsp/servlets_jsp_instalacion.htm

CHILISOFT ASP

<http://www.sunone.com/software/ChiliSoft-Linux.html>

<http://www.sunone.com/software/ChiliSoft-Windows.html>

http://developer.chilisoft.com/caspdoc/chilisoft_asp_docs.pdf

<http://www.chilisoft.com/forum/>

POSTGRESQL

http://www.linux-es.com/Montar_una_web_con_apache+php+postgreSQL_en_Linux.html

http://www.asp-hosting.netregistry.com.au/asp_adding_postgres.html

<http://www.postgresql.org/users-lounge/docs/faq.html>

<http://tlali.iztacala.unam.mx/listas/pgsql-ayuda/2002-March/003410.html>

APACHE WEB SERVER

http://www.principiantes.org/Añadir_soporte_PHP_a_APACHE.htm

http://www.principiantes.org/Instalacion_de_Apache_Web_Server_en_Windows.htm

http://www.principiantes.org/Configurar_Servidor_Apache.htm

http://www.internautas.org/curso_servidores/apache.php

INTERNET INFORMATION SERVER

http://fmc.axarnet.es/winnt4svr/soporte/tema_08.htm

<http://www.htmlpoint.com/iis/02.htm>

<http://www.htmlpoint.com/iis/01b.htm>

TOMCAT WEB SERVER

http://www.osmosislatina.com/aplicaciones/servidor_web.htm

<http://www.ultradev-es.com/Instalar Tomcat en Windows 2000.html>

<http://java.programacion.net/tomcatintro/tomcat1.htm#tomcat1>

<http://www.javahispano.com/articulos/como/apache-tomcat/windows.shtml>

<http://geneura.ugr.es/~jmerelo/JSP/Instalacion de Tomcat 4.0.html>

[http://www.osmosislatina.com/catalina/configuracion\(Tomcat 4.x\).htm](http://www.osmosislatina.com/catalina/configuracion(Tomcat 4.x).htm)

<http://www.alu.ua.es/p/psp4/Documentacion/index.html>

<http://www.osmosislatina.com/tomcat/instalacion.htm#ejecucion>

MOD_JK

http://jakarta.apache.org/builds/jakarta-tomcat/release/v3.3/bin/win32/i386/mod_jk.dll

<http://www.alu.ua.es/j/jhm/cocoon/install3.htm>

CYGWIN

<http://www.cygwin.com/>

<http://www.cygwin.com/cygwin/cygwin-ug-net.pdf>

<http://www.neuro.gatech.edu/users/cwilson/cyguutils/index.html>

<http://www.infohackers.org/sections.php?op=viewarticle&artid=126>

CYGIPC

<http://www.neuro.gatech.edu/users/cwilson/cyguutils/cygipc/index.html#README>

ODBC

<http://skyscraper.fortunecity.com/rofl/434/articulos/odbc/odbc.htm>

<http://skyscraper.fortunecity.com/rofl/434/articulos/odbc/odbc.htm>

<http://es.tldp.org/Articulos-periodisticos/jantonio/odbc/odbc2.html>

JDBC

<http://industry.java.sun.com/products/jdbc/drivers>

<http://java.programación.net/jdbc1.html>

http://www.programacion.com/java/tutorial.jdbcbas.3.html#3_ejemplocompleto

OTROS

<http://mundolinux.com> (lplanet Web Server)

<http://mundolinux.com/lplanet.pdf> (lplanet Web Server)

<http://www.zeus.com/products/zws/> (Zeus Web Server)

<http://www.tejedoresdelweb.com/307/article-1052.html> (FTP, HTTP, Finger, Gopher, SMTP, NNTP)