

BACULA

La Ultima Linea de Defensa

Cuando Todo lo Demás Falla.

Victor Hugo dos Santos
contacto.vhs@gmail.com

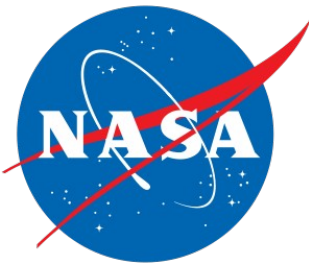
Objetivos

Muchas empresas no cuentan con un software de respaldo estándar y confiable, siendo necesario realizar los respaldos de manera manual, estando vulnerables a errores humanos y utilizando software de “copia y/o transferencia” que no siempre cumplen los requisitos básicos que un administrador de backup necesita.

Introducción

- › Administrador de Respaldos
- › Arquitectura Modular y/o Cliente/Servidor
- › GPL (General Public License)
- › MultiPlataforma

Quien Usa Bacula



Introducción

- Bacula es un software que permite administrar los respaldos, restauraciones y verificación de los archivos ubicados en los distintos computadores/servidores de la red.
- Esta basado en un esquema cliente/servidor, es relativamente fácil de utilizarlo y extremadamente modular, permitiendo usarlo en una red local pequeña o en una gran red empresarial.

Introducción

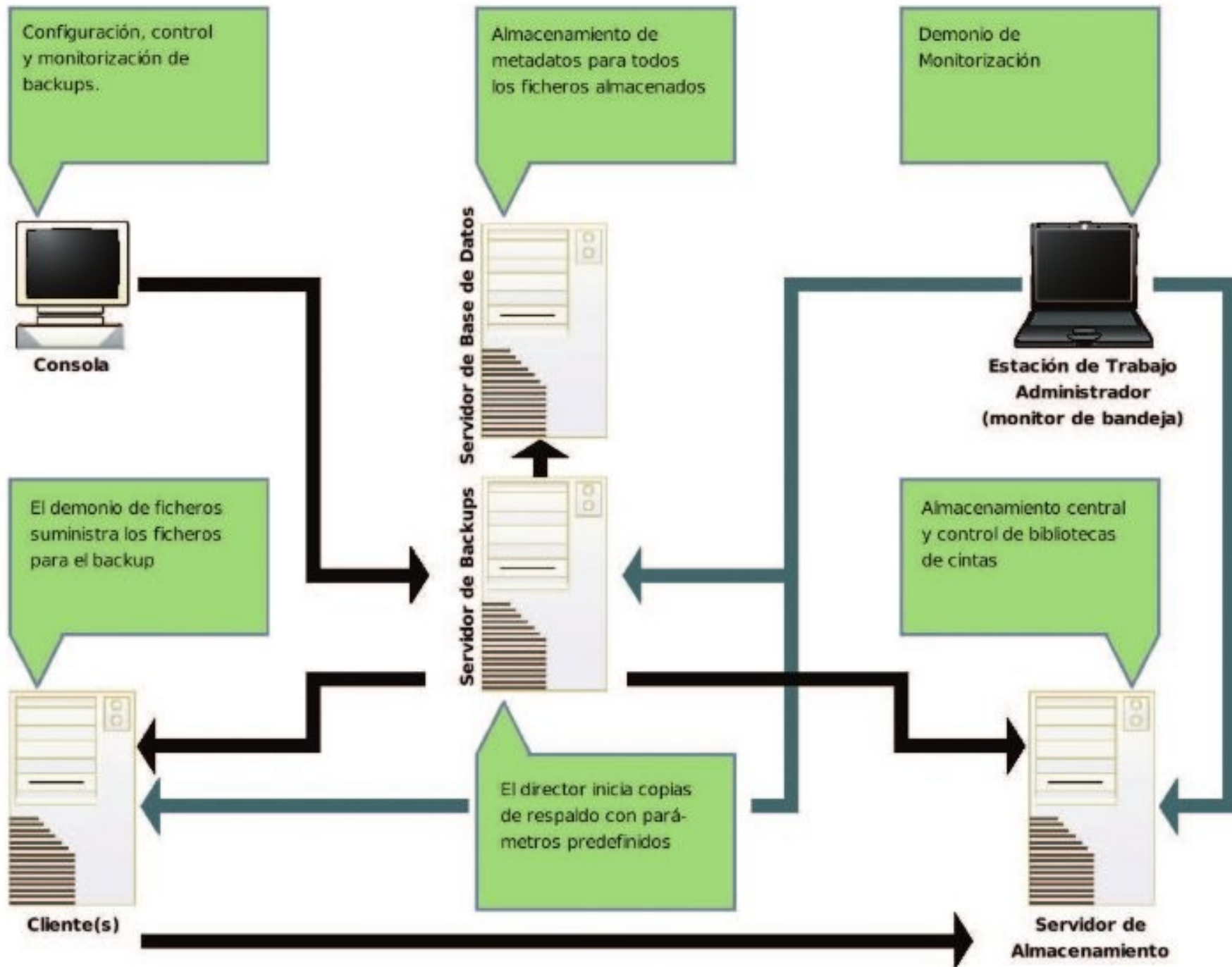
- › Su desarrollo es constante, está licenciado bajo GPL lo que garantiza que los códigos fuentes siempre estarán disponibles para lo que desee modificar y/o reutilizar y contiene vasta y completa documentación.
- › La versión servidor, está disponible únicamente para el SO Linux, pero existen varias recompilaciones de la versión cliente para distintos SO.

Comprendiendo a Bacula

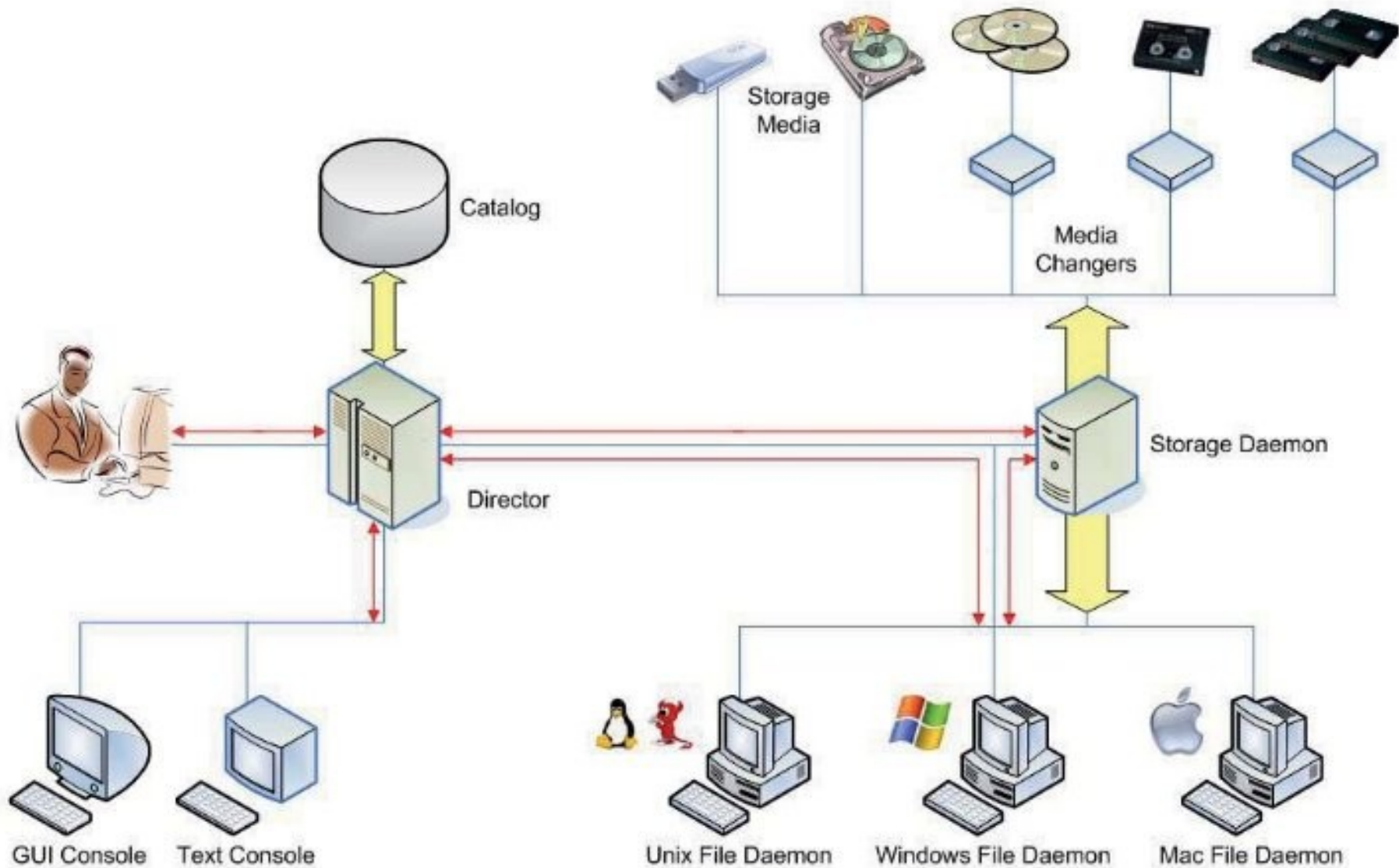
- Cuesta quince minutos comprender el funcionamiento del sistema, gracias a la documentación y a la separación lógica de las tareas a realizar.
- La configuración es algo más compleja siendo necesaria la lectura del manual y/o algunos de los diversos “howtos” que existen en la red.

El Sistema se Divide en:

- *Bacula Director - DIR*
- *Bacula File - FD*
- *Bacula Storage - SD*
- *La Consola*
- *El Catálogo*

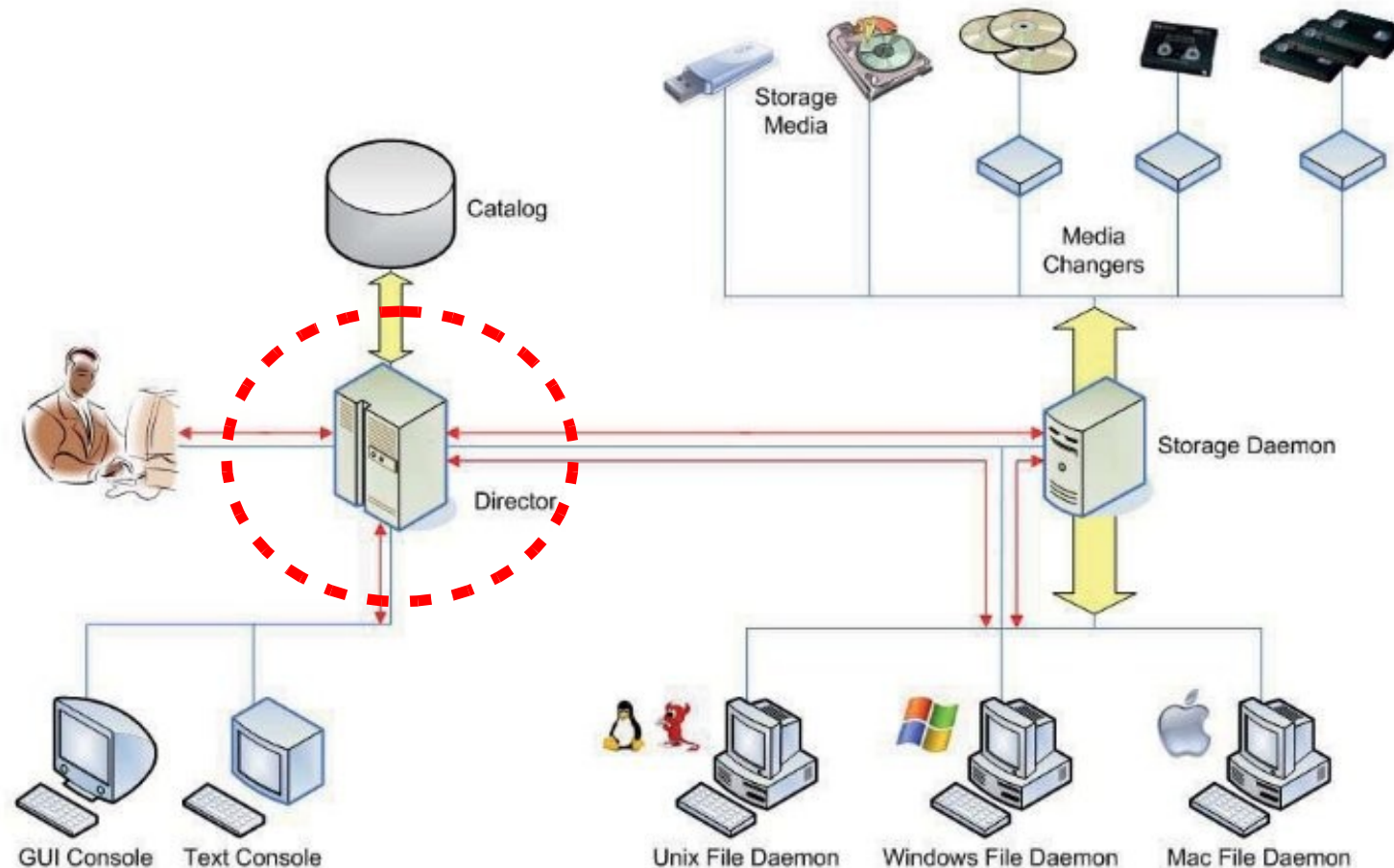


Vision General



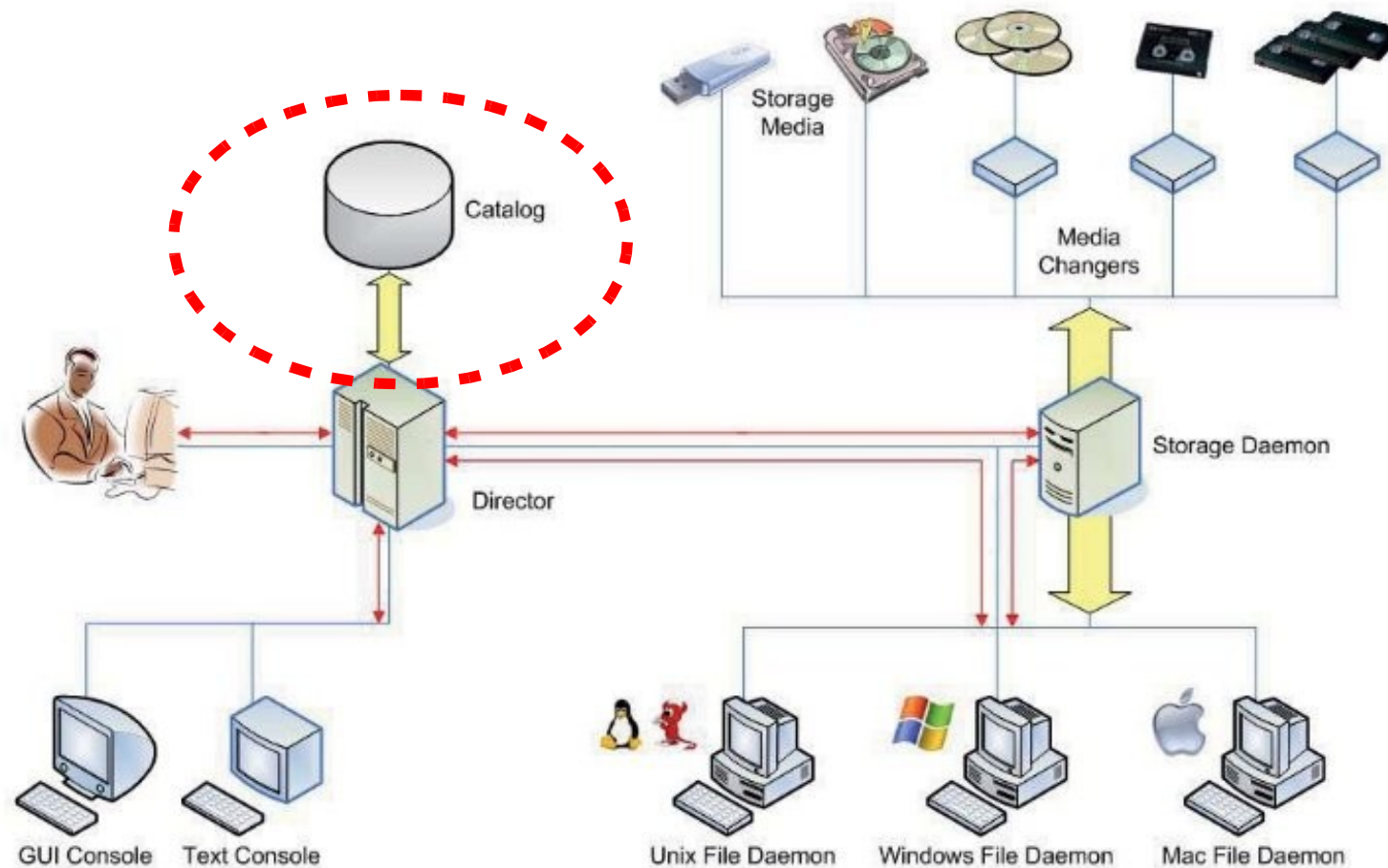
Bacula Director

*Es el servicio encargado de gestionar todas las operaciones de backup.
El director sabe los trabajos que se van a realizar, cuando, donde y como.
Además se encarga de restaurar los ficheros que le pidamos y su verificación.*



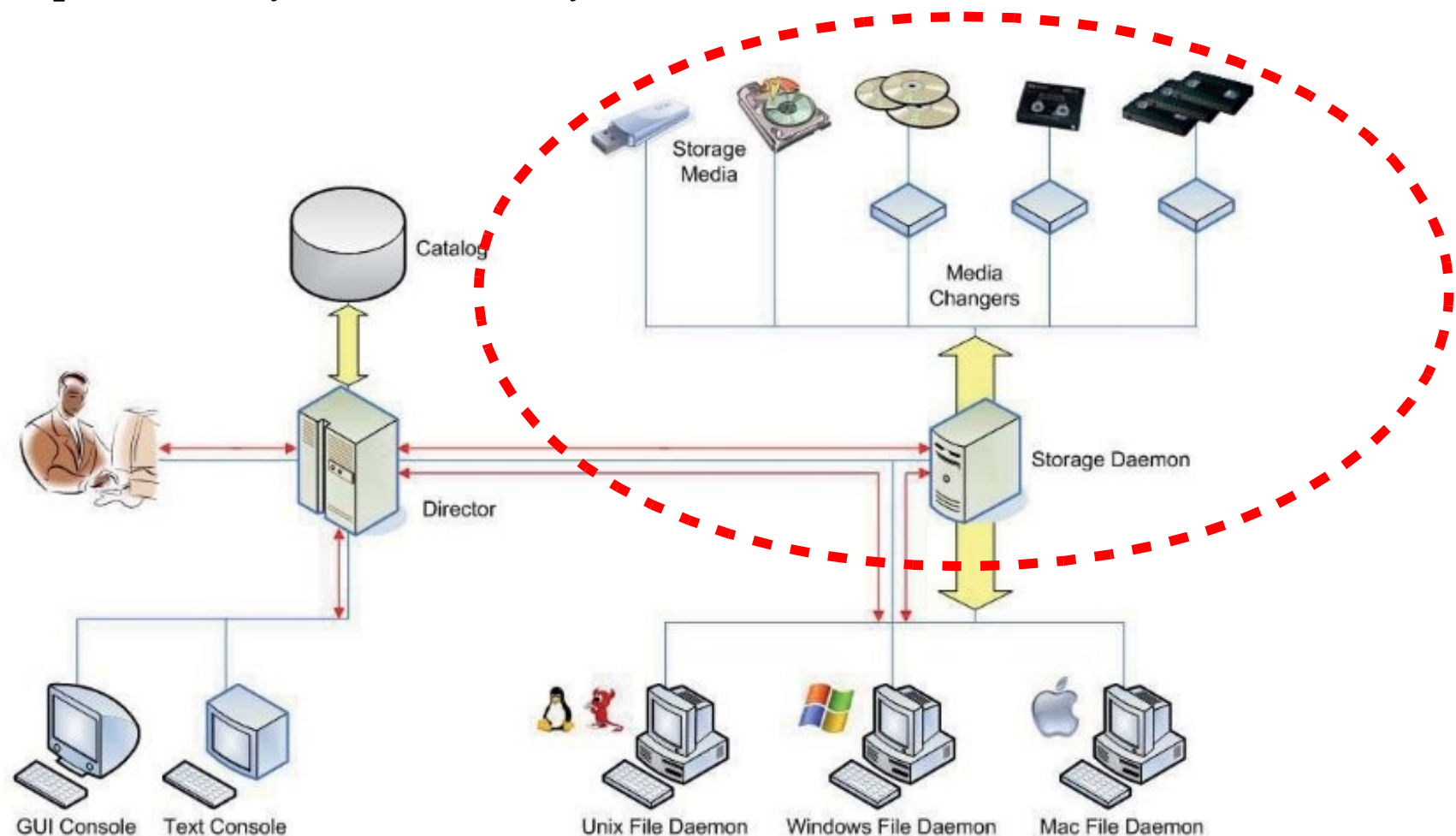
El Catálogo

Es la Base de Datos donde el director guarda y registra todas sus operaciones.



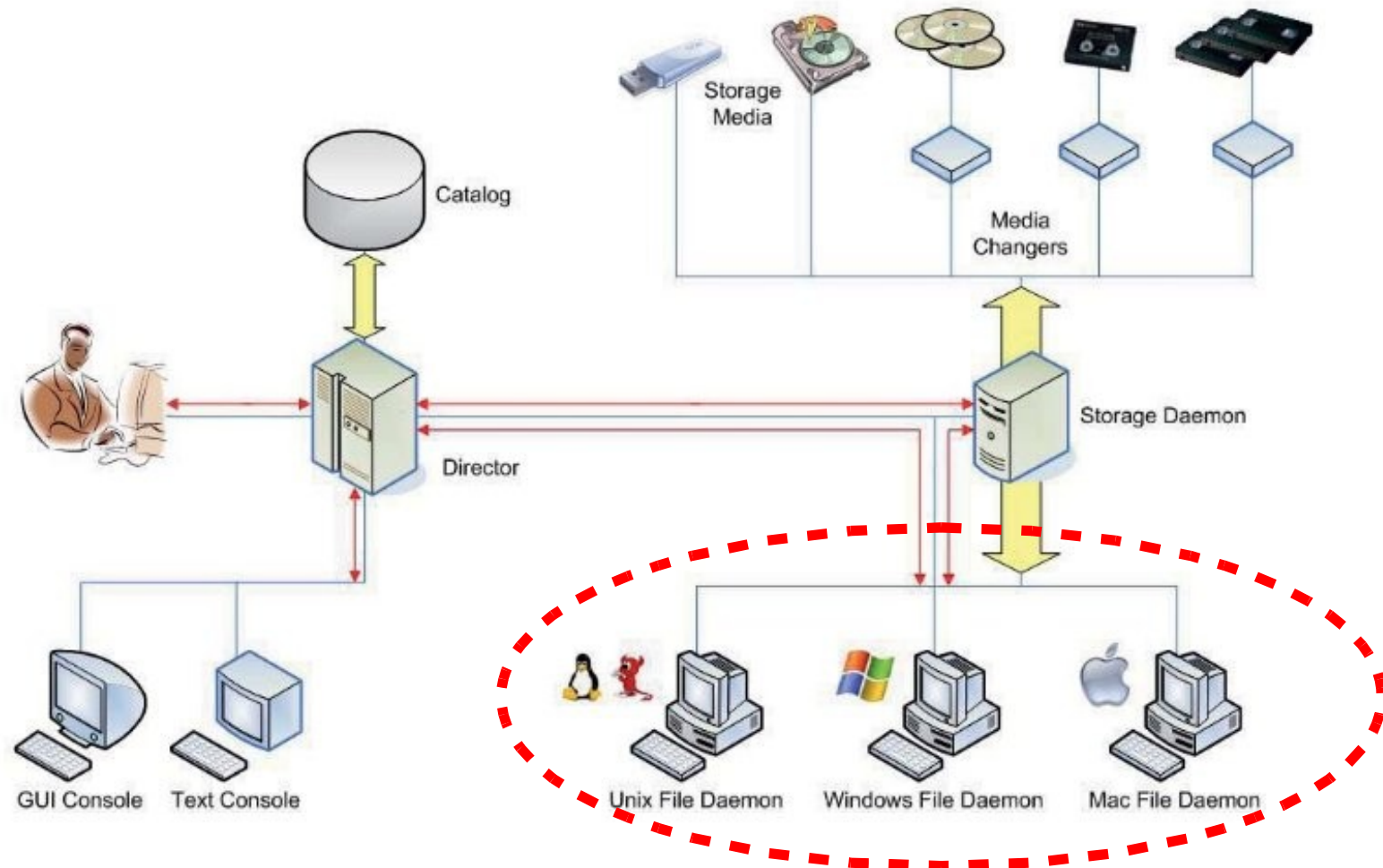
Bacula Storage

Es el servicio que se encarga de la lectura/escritura física de los archivos en los volúmenes que estén definidos (cintas, ficheros entre otros)



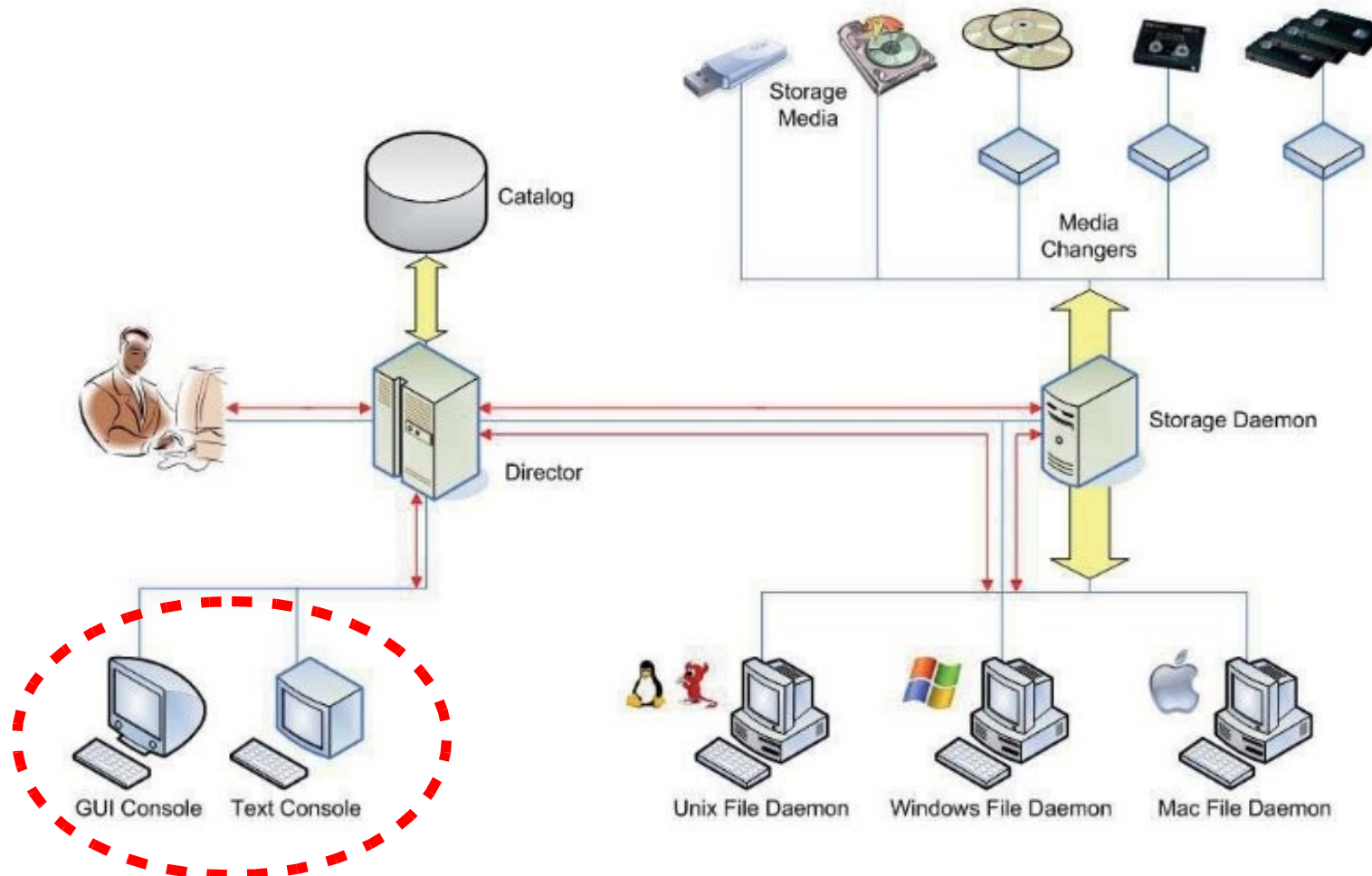
Bacula Cliente/File

Es el software cliente y es necesario instalarlo en todas las máquinas de las que queremos hacer respaldo. Su función es leer y transmitir los ficheros que el Director le pida y/o restaurarlos.



La Consola

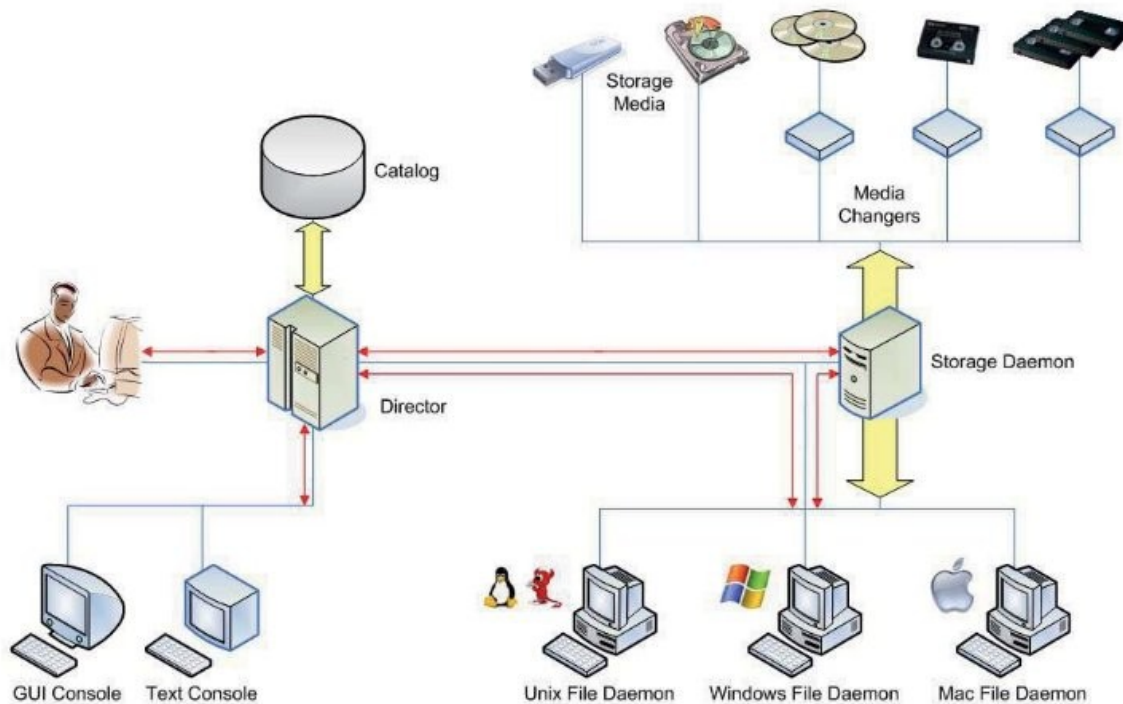
Es el software que se instala en la(s) maquina(s) que desea que se conecte al director y desde donde podremos dar ordenes, programar tareas, hacer consultas entre otros comandos administrativos.



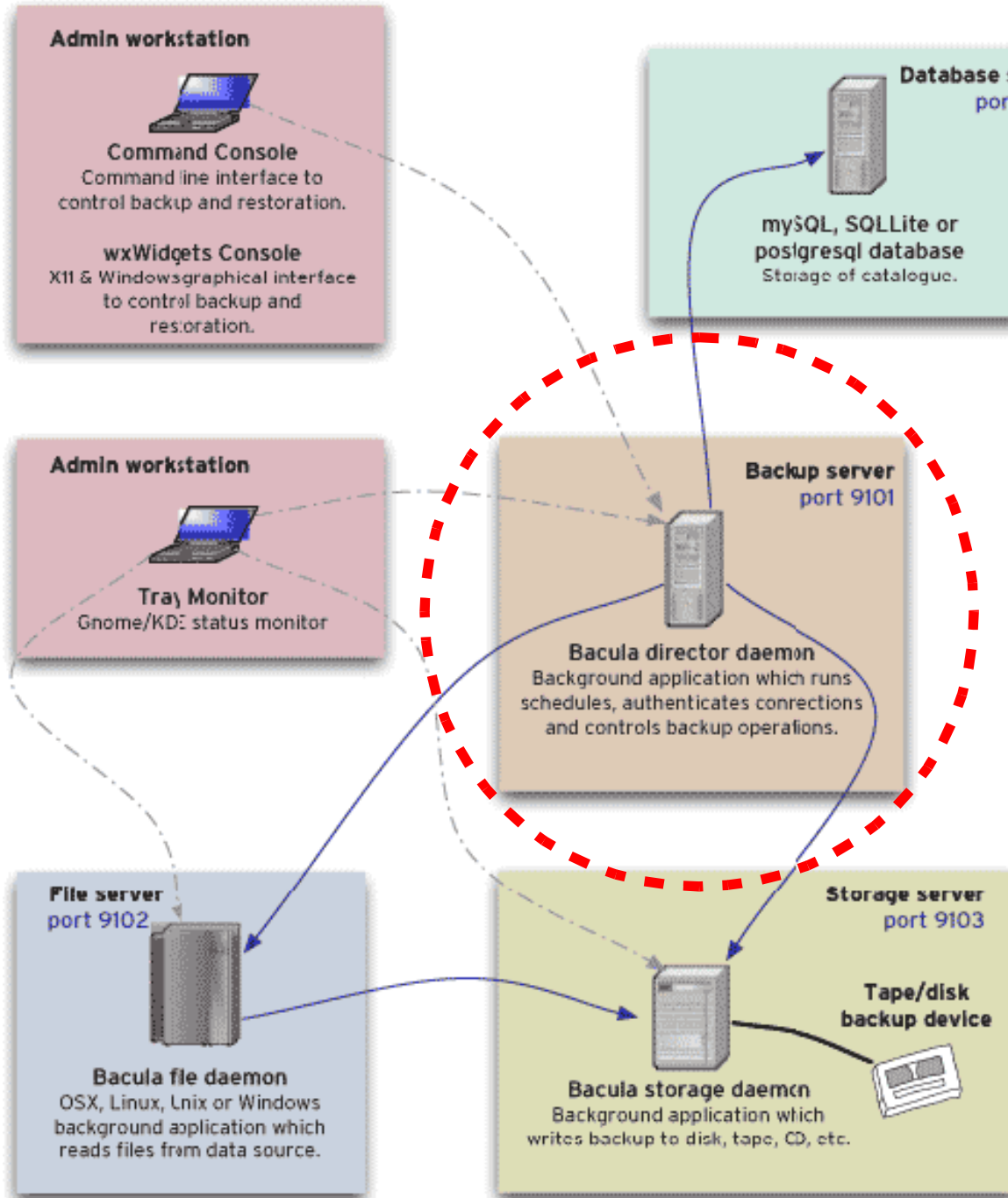
Funcionamiento

En el Director (DIR), se define una serie de trabajos (jobs) y a la fecha y hora definida se pone en marcha.

El Director contacta con el primer Cliente(FD) que tiene definido y se comprueba los ficheros o directorios que estan definidos.



Los Clientes envian los archivos que hayan cambiado al Servicio Storage(SD) que almacena en el volumen los ficheros que le mandan los clientes.



➤ *Bacula Director*

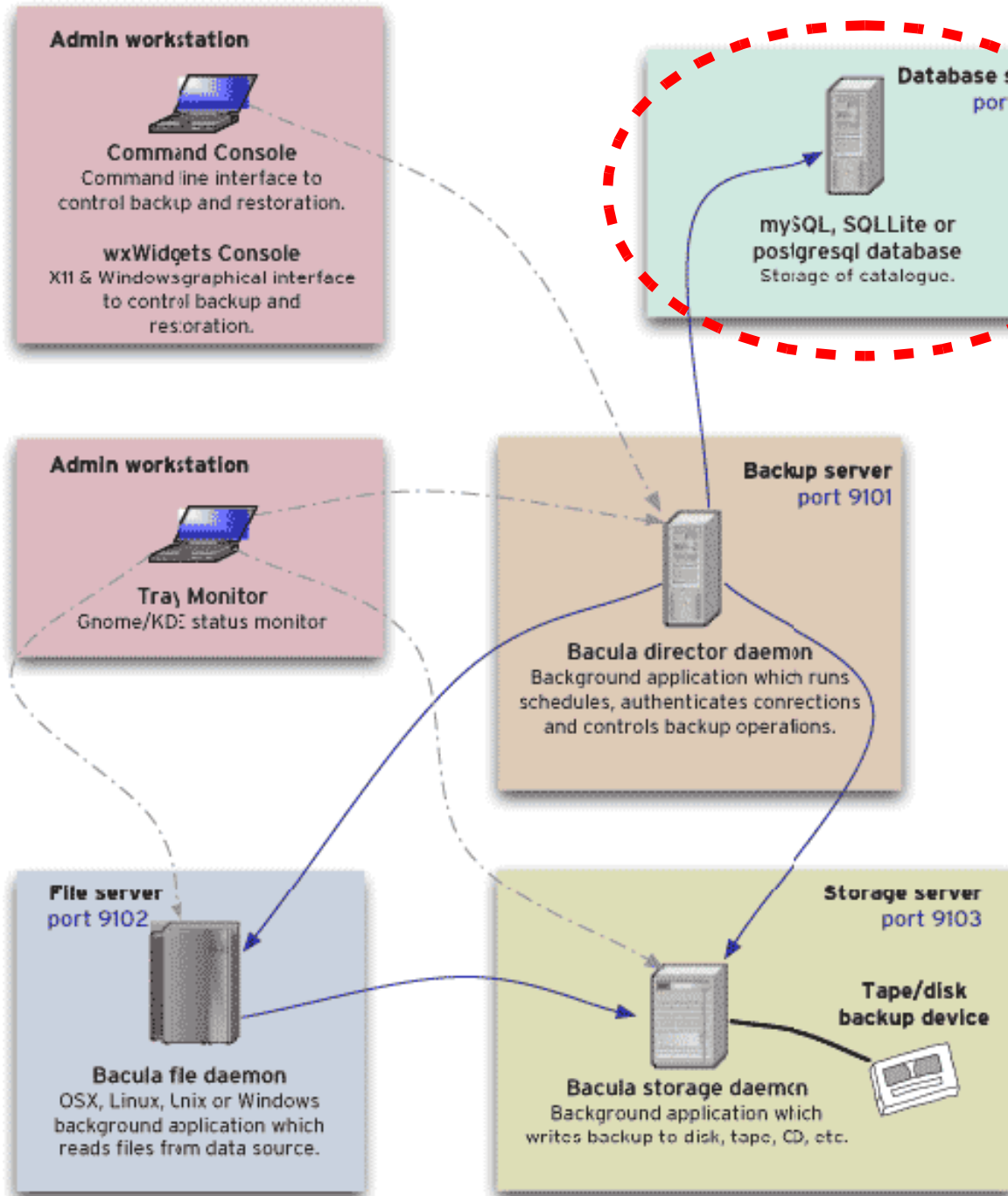
Es el servicio encargado de gestionar todas las operaciones de backup.

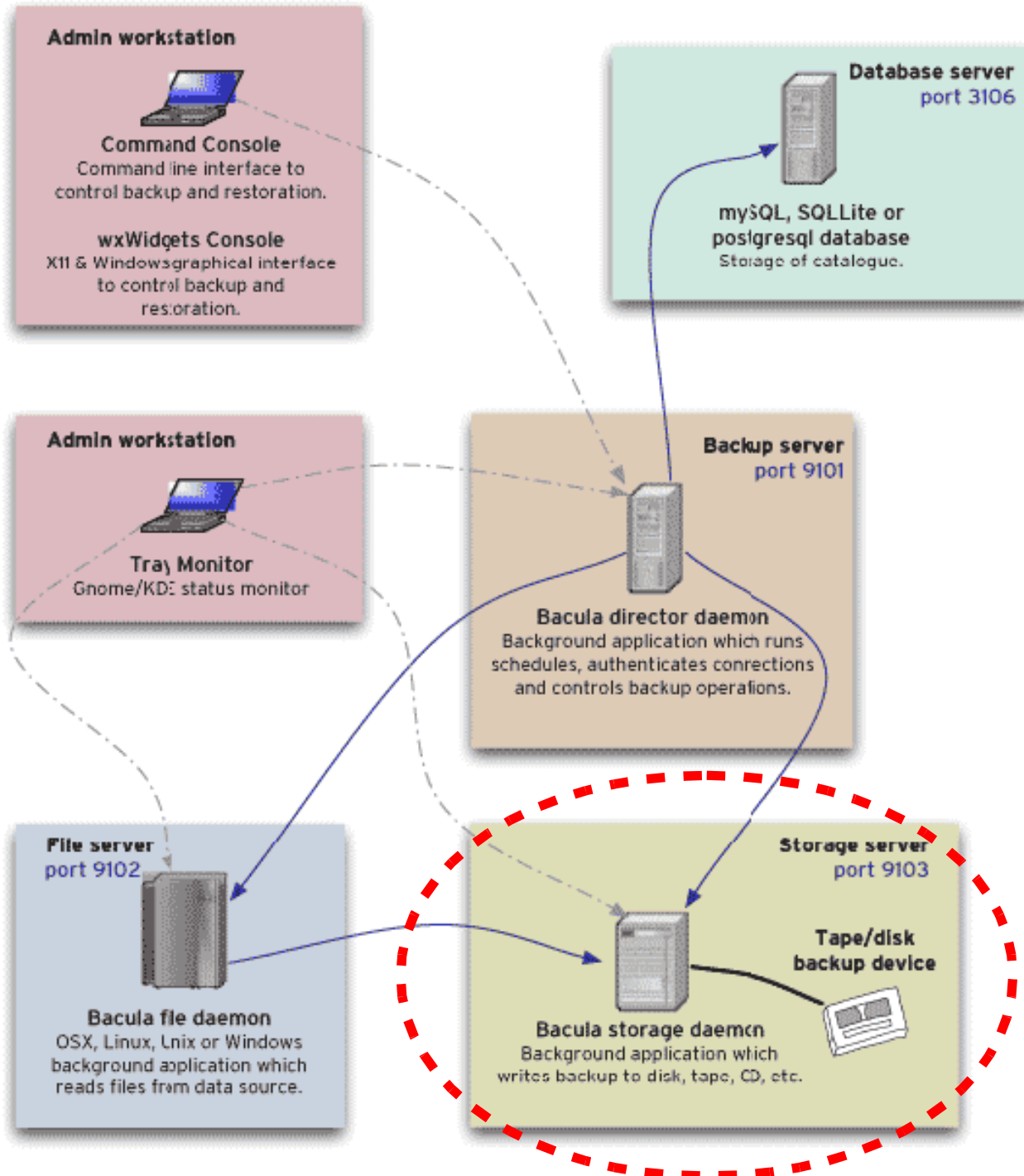
El director sabe los trabajos que se van a realizar, cuando, donde y como.

Además se encarga de restaurar los ficheros que le pidamos y su verificación.

El Catálogo

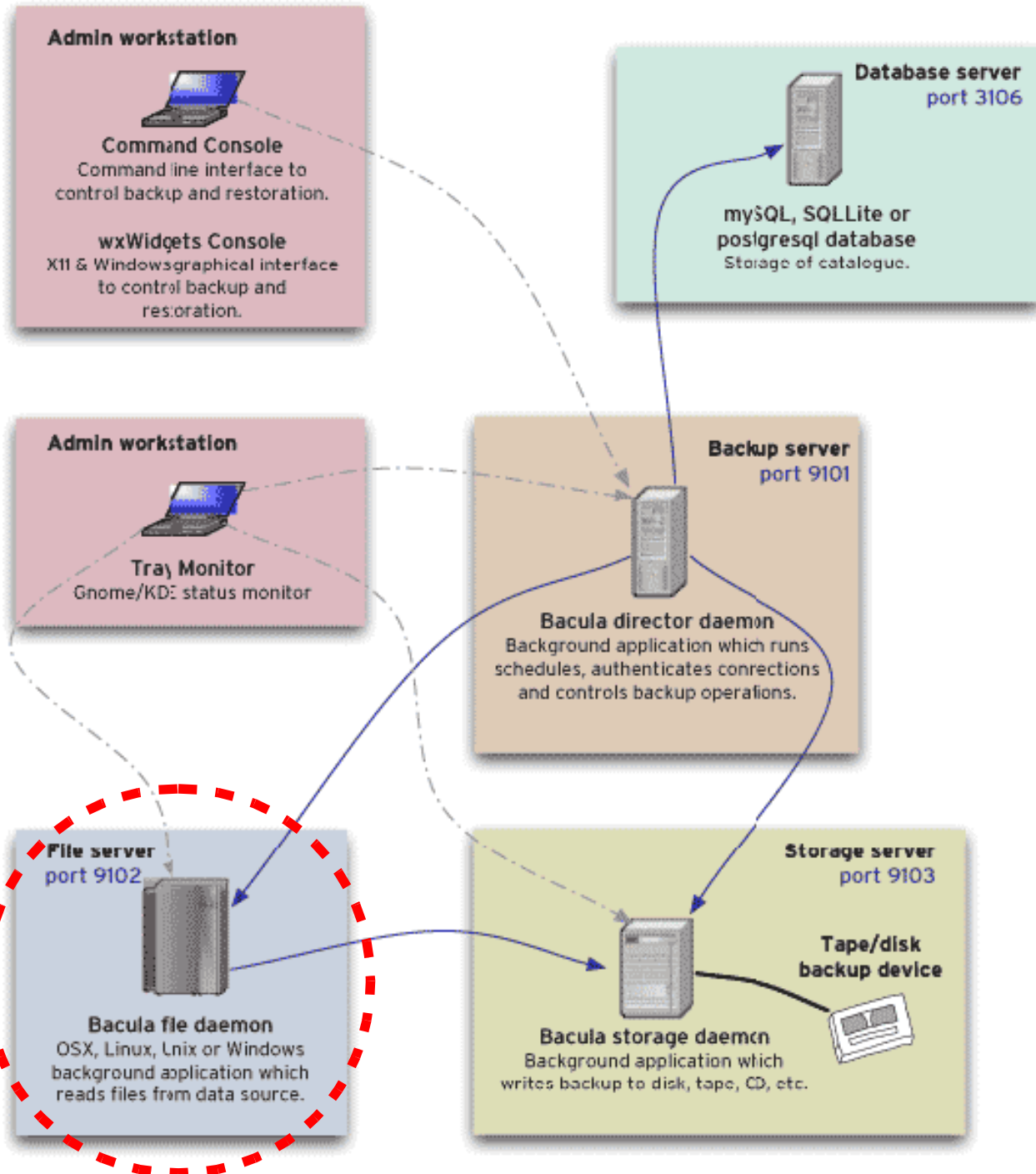
Es la Base de Datos donde el director guarda y registra todas sus operaciones.





➤ Bacula Storage

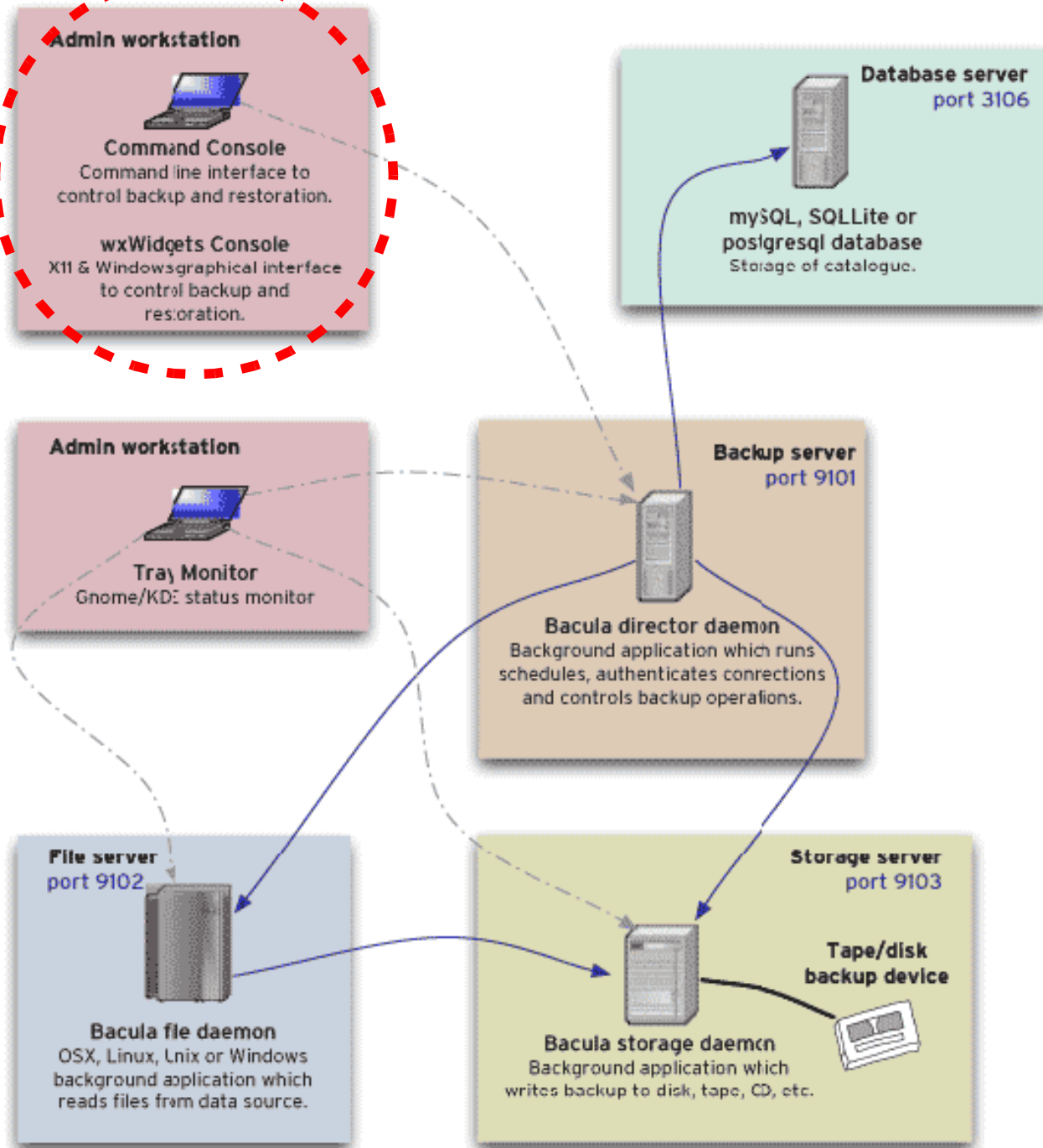
Es el servicio que se encarga de la lectura/escritura física de los archivos en los volúmenes que estén definidos (cintas, ficheros entre otros)



➤ Bacula Cliente/File

Es el software cliente y es necesario instalarlo en todas las máquinas de las que queremos hacer respaldo.

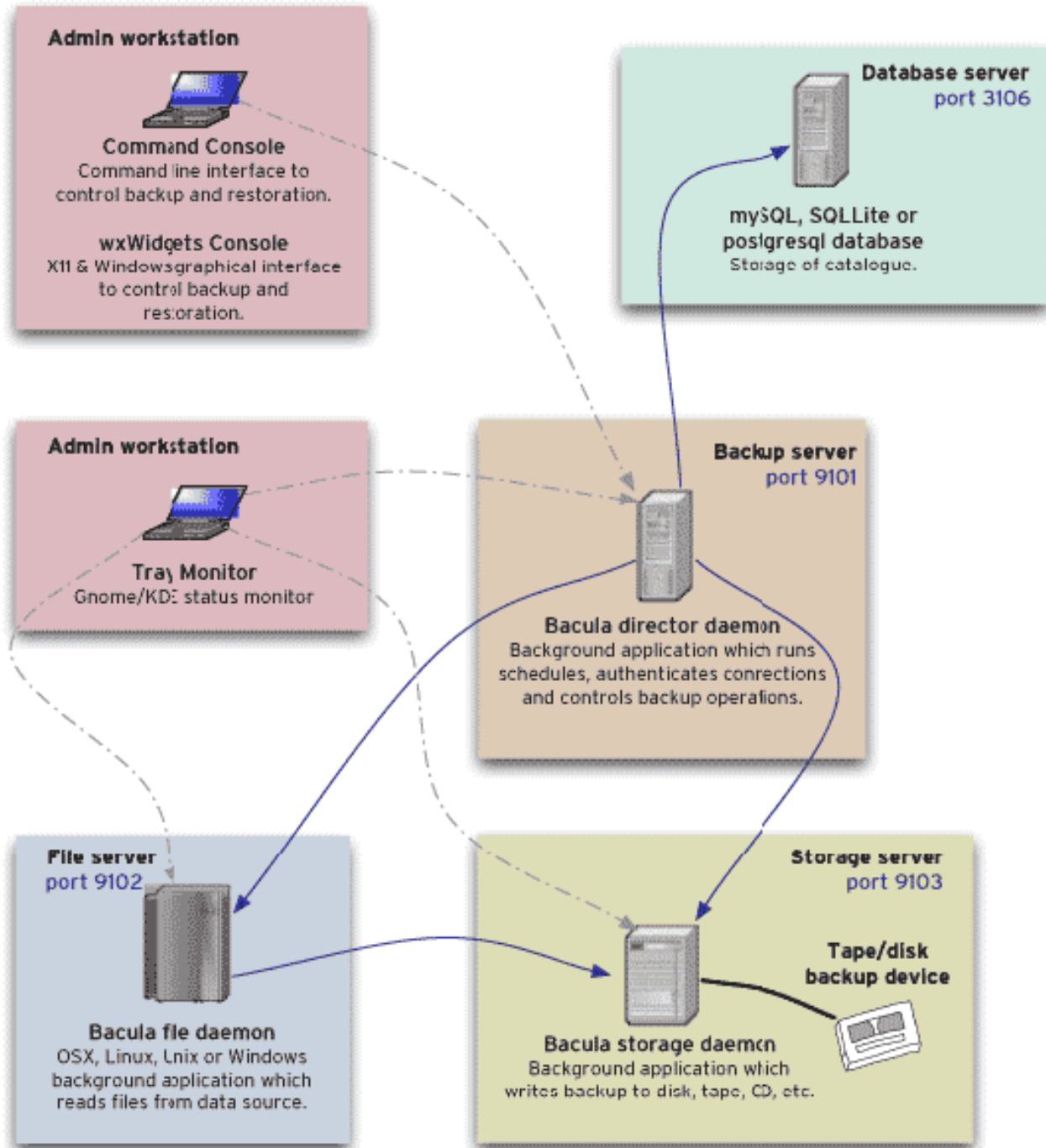
Su función es leer y transmitir los ficheros que el Director le pida y/o restaurarlos.



➤ La Consola

Es el software que se instala en la(s) maquina(s) que desea que se conecte al director.

Y desde donde podremos dar ordenes, programar tareas, hacer consultas entre otros comandos administrativos.



Funcionamiento

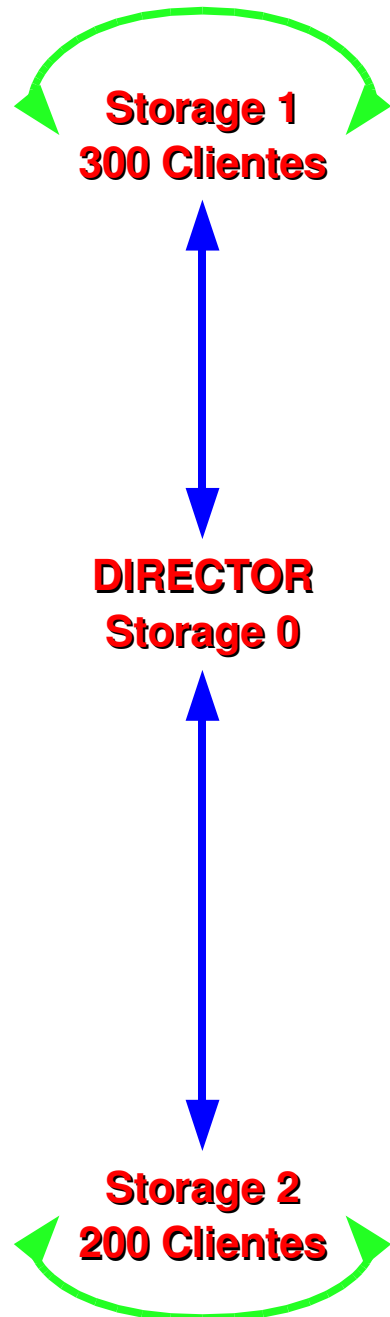
En el Director (DIR), se define una serie de trabajos (jobs) y a la fecha y hora definida se pone en marcha.

El Director contacta con el primer Cliente(FD) que tiene definido y se comprueba los ficheros o directorios que estan definidos.

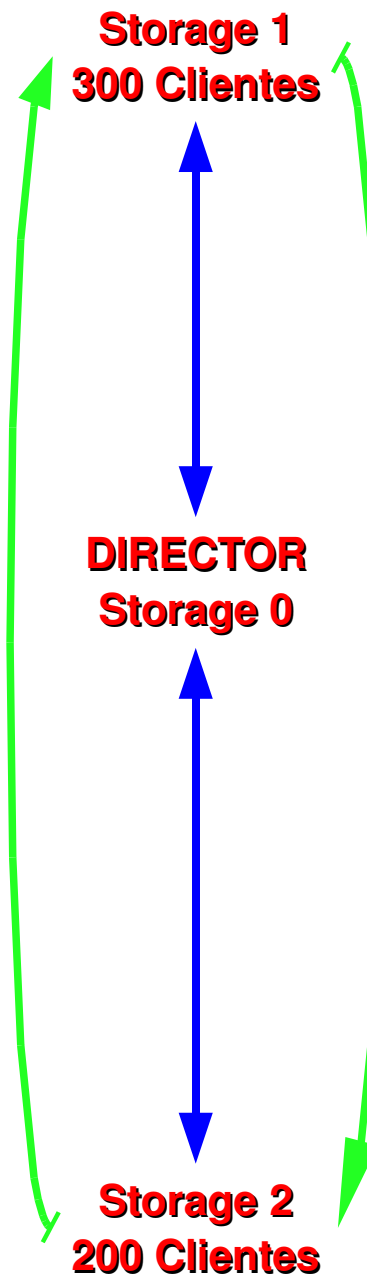
Los Clientes envian los archivos que hayan cambiado al Servicio Storage(SD) que almacena en el volumen los ficheros que le mandan los clientes.

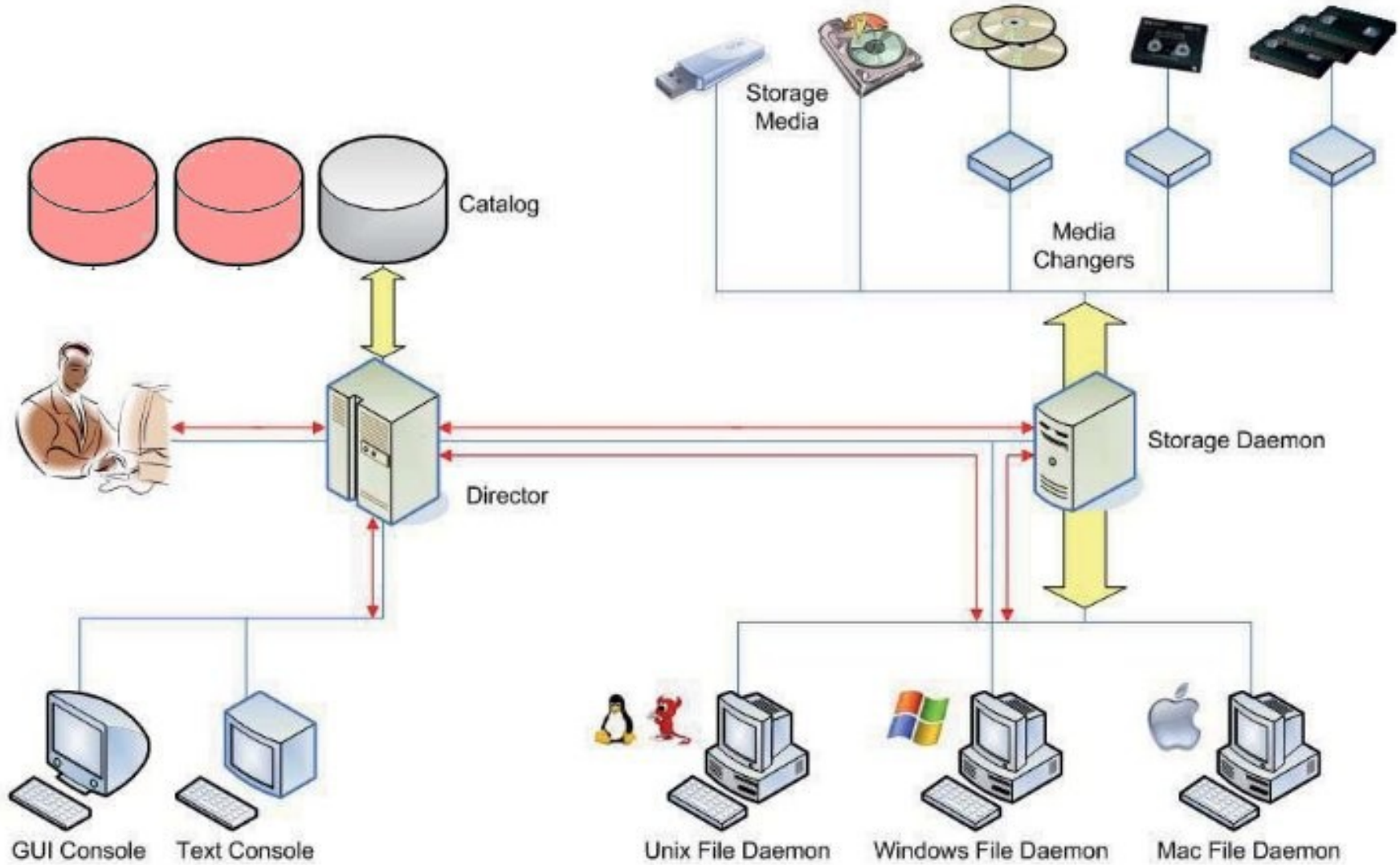
Cual es la Gracia ???

Enlace Modesto
Políticas Normales



Super Enlace
Políticas Extremas





Como se Configura Bacula

Ejemplo DIRECTOR (GENERAL)

```
#
# Default Bacula Director Configuration file
Director {
    Name = ELINUX-dir
    DIRport = 9101
    QueryFile = "/usr/lib/bacula/query.sql"
    WorkingDirectory = "/var/lib/bacula"
    PidDirectory = "/var/run"
    Maximum Concurrent Jobs = 30
    Password = "PASSWORD"      # Console password
}

# Datos para coneccion con la base de Datos
Catalog {
    Name = MiCatalogo
    dbname = bacula; DB Address = "127.0.0.1"; user = bacula; password = "elinux2009"
}
```

Ejemplo DIRECTOR(CLIENTES)

Archivo de configuracion de los clientes a los cuales se va a respaldar

```
Client {  
    Name = win2003-fd  
    Address = 192.168.100.101  
    FDPort = 9102  
    Catalog = MiCatalogo  
    Password = "PASSWORD-CLIENTE1"  
    Job Retention = 6 months  
    AutoPrune = yes  
}  
Client {  
    Name = centos-fd  
    Address = 192.168.100.102  
    FDPort = 9102  
    Catalog = MiCatalogo  
    Password = "PASSWORD-CLIENTE2"  
    Job Retention = 6 months  
    AutoPrune = yes  
}
```

Ejemplos DIRECTOR(FILESET)

Este es el perfil de respaldo del catalogo/BD

```
FileSet {  
  Name = "Catalogo"  
    Include { Options { signature = MD5; compression=GZIP9 }  
      File = /var/lib/bacula/bacula.sql  
    }  
}
```

Ejemplo de un respaldo completo + exclusión de archivos

```
FileSet {  
  Name = "Linux-Completo"  
  Include { Options { signature = MD5; compression=GZIP5 }  
    File = /  
  }  
  Exclude {  
    File = /proc  
    File = /tmp  
    File = /.jjournal  
    File = /.fsck  
  }  
}
```

Ejemplos DIRECTOR(FILESET)

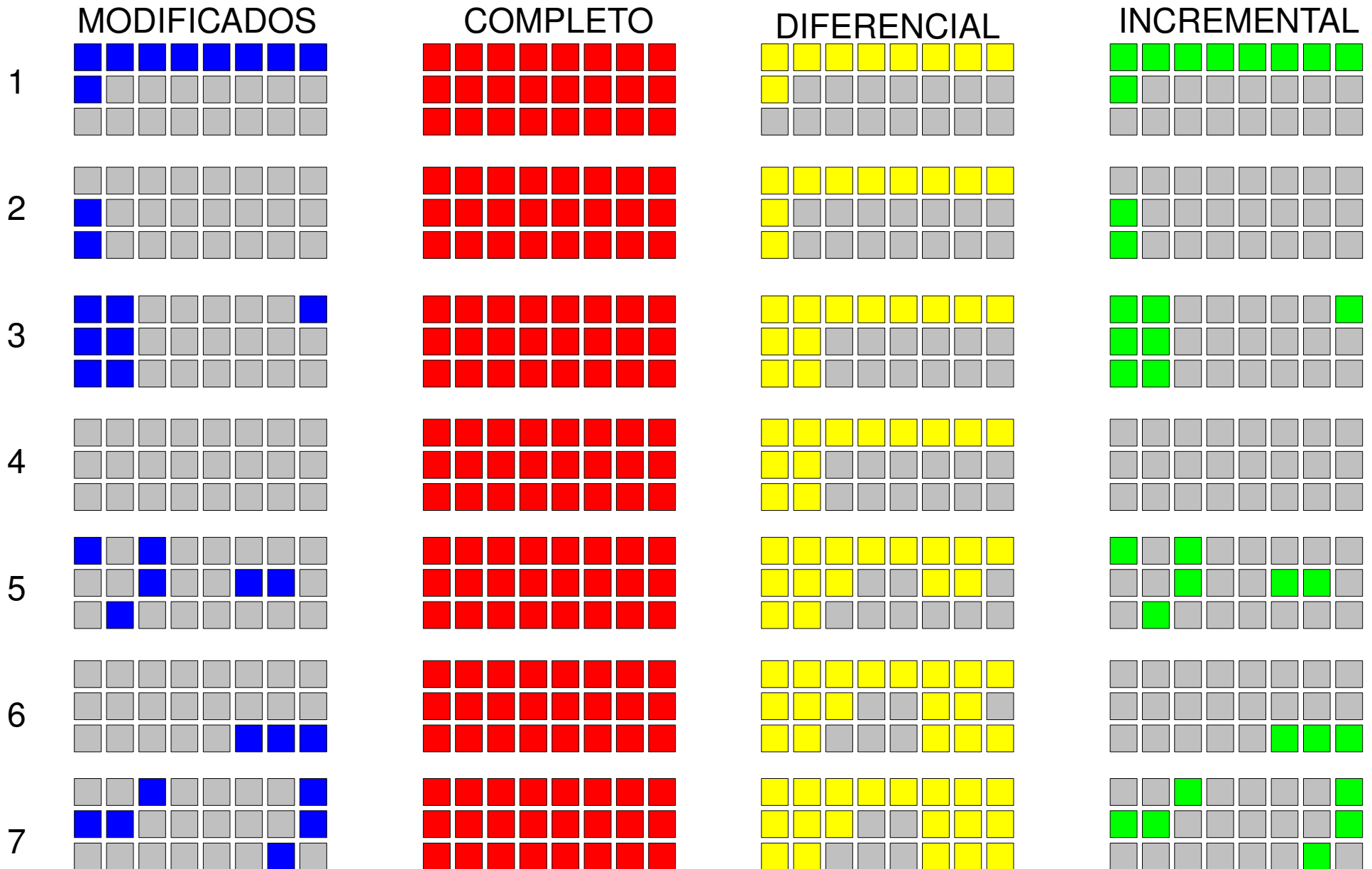
Respaldo Windows Servers

```
FileSet {  
    Name = "Windows-Servers"  
    Include {  
        Options { signature = SHA1; compression=GZIP6 }  
        File = "c:"  
        File = "d:"  
    }  
}
```

Respalda la carpeta Mis Documentos y excluye archivos Multimedia

```
FileSet {  
    Name = "MisDocumentos"  
    Include {  
        Options { signature = SHA1; compression=GZIP6; IgnoreCase = yes; Exclude = yes  
            WildFile = "*.avi"  
            WildFile = "*.mp3"  
            WildFile = "*.ogg"  
        }  
    File = "C:/Mis Documentos"  
}
```

Tipos de Respaldos



Ejemplo DIRECTOR(SCHEDULE)

- # Ejecuta el respaldo COMPLETO en el 1° Domingo de cada mes
- # Ejecuta el respaldo DIFERENCIAL en el 2°,3°,4° y 5° Domingo de cada mes
- # Ejecuta el respaldo INCREMENTAL de lunes a sábado

```
Schedule {  
    Name = "CicloSemanal"  
    Run = Full 1st sun at 00:05  
    Run = Differential 2nd-5th sun at 00:05  
    Run = Incremental mon-sat at 00:05  
}
```

- # Ejecuta el respaldo COMPLETO TODOS los días (catalogo)

```
Schedule {  
    Name = "CicloSemanalDespuesBackup"  
    Run = Full sun-sat at 00:10  
}
```


Ejemplo DIRECTOR(SCHEDULE)

Ejecuta el respaldo COMPLETO 4 días por mes

```
Schedule {
```

```
    Name = "RespaldoMensual"
```

```
    Run = Full on 20-23 at 2:05
```

```
}
```

Ejecuta el respaldo COMPLETO a cada hora y 05 minutos

y el respaldo DIFERENCIAL en los siguientes 10 minutos de la misma hora

```
Schedule {
```

```
    Name = "RespaldoParanoico"
```

```
    Run = Level=Full hourly at 0:05
```

```
    Run = Level=Diferencial hourly at 0:15
```

```
    Run = Level=Diferencial hourly at 0:25
```

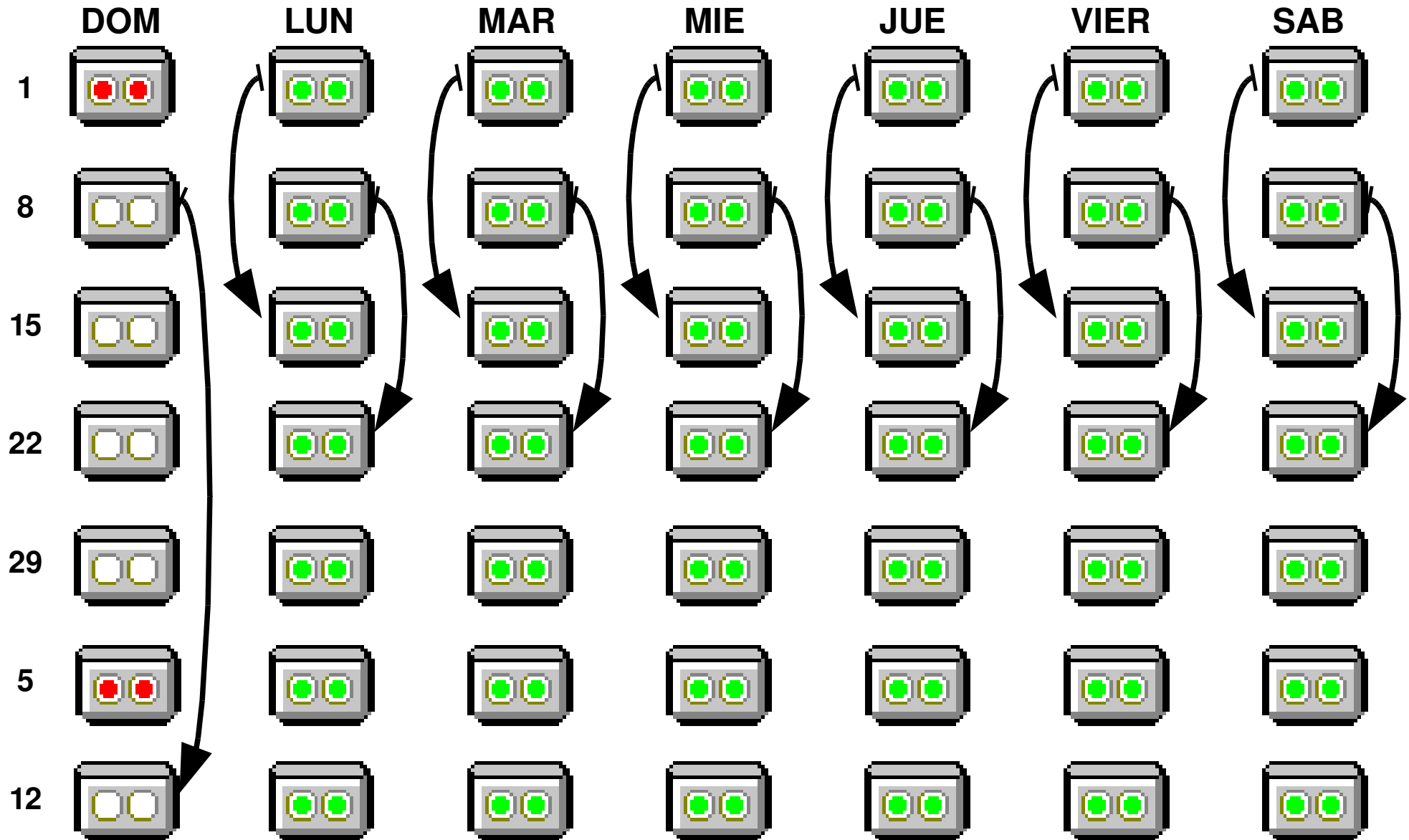
```
    Run = Level=Diferencial hourly at 0:35
```

```
    Run = Level=Diferencial hourly at 0:45
```

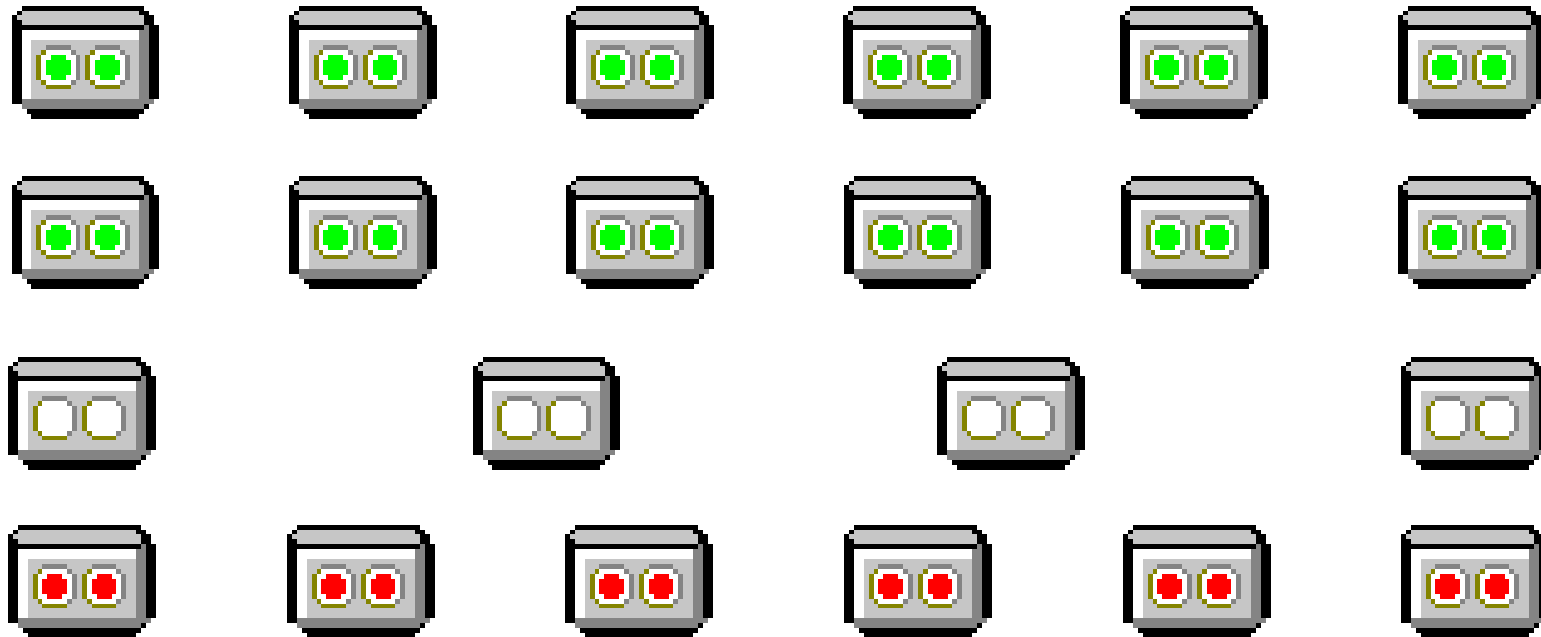
```
    Run = Level=Diferencial hourly at 0:55
```

```
}
```

Ejemplo Reutilización DIRECTOR(POOLS)



Ejemplo Reutilización DIRECTOR(POOLS)



22 CINTAS = 6 MESES DE RESPALDOS

Ejemplos DIRECTOR(STORAGE)

Especifica donde se van a guardar los archivos

Definition of file storage device

```
Storage {  
    Name = DiscoLocal  
    Address = 192.168.100.1  
    SDPort = 9103  
    Password = "PASSWORD"  
    Device = RAID-500GB  
    Media Type = File  
}
```

Especificación de la Unidad de Cinta

```
Storage {  
    Name = CINTA  
    Address = 192.168.100.20  
    SDPort = 9103  
    Password = "PASSWORD"  
    Device = SDLT-600  
    Media Type = "Super DLT II"  
    Autochanger = NO  
}
```

Ejemplos DIRECTOR(CONSOLAS)

Consola de estado (solamente lectura)

```
Console {  
    Name = CONSOLA-ADMIN  
    Password = "PASSADMIN-muycompleja"  
}
```

```
Console {  
    Name = MONITOR-JUANITO  
    Password = "PASSMonitor"  
    CommandACL = status, .status  
}
```

```
Console {  
    Name = OJOGRANDE-MON  
    Password = "PASSMonitorDIFERENTE_del_anterior"  
    CommandACL = status, .status  
}
```

Ejemplos DIRECTOR(POOLS)

Acá definimos los distintos grupos de medios que iremos a utilizar en ese ejemplo, tenemos 4:
Default / Mensual-Completo / Semanal-Diferencial / Diario-Incremental
Respaldo Incremental (diario), respalda **solo** los archivos modificados desde el ultimo respaldo

```
Pool {  
    Name = Diario-Incremental           # Nombre Descriptivo para el POOL  
    Pool Type = Backup                    # Tipo  
    Recycle = yes                         # Reutilizar  
    AutoPrune = yes                       # Borrarlos automáticamente  
    Volume Retention = 14 days          # Tiempo de retención  
    Maximum Volume Jobs = 1               # Numero de Trabajos por Volumen  
    Label Format = Diario-                 # Nombre predefinido  
}
```

Respaldo Diferencial (Semanal), respalda **todos** los archivos que fueron cambiados desde el ultimo respaldo "**Completo**"

```
Pool {  
    Name = Semanal-Diferencial  
    Pool Type = Backup  
    Recycle = yes  
    AutoPrune = yes  
    Volume Retention = 28 days  
    Maximum Volume Jobs = 1  
    Label Format = Semanal-  
}
```

Ejemplos DIRECTOR(POOLS)

Respaldo Completo (mensual), respalda **todos los archivos**

```
Pool {  
    Name = Mensual-Completo  
    Pool Type = Backup  
    Recycle = yes  
    AutoPrune = yes  
    Volume Retention = 6 months  
    Maximum Volume Jobs = 1  
    Label Format = Mensual-  
}
```

Destino por Defecto (Respaldos del Sistema)

```
Pool {  
    Name = Default           # Nombre Descriptivo para el POOL  
    Pool Type = Backup        # Tipo  
    Recycle = yes            # Reutilizar  
    AutoPrune = yes          # Borrarlos automáticamente  
    Volume Retention = 2 months # Tiempo de retención  
    Maximum Volume Jobs = 1   # Numero de Trabajos por Volumen  
    Label Format = System-     # Nombre predefinido  
}
```


Ejemplos DIRECTOR(JOBS)

Especifica los trabajos que se van a ejecutar
el cuando, donde y como, en este ejemplo tenemos

Crea un Ejemplo de Job para ser utilizado por defecto

```
JobDefs {  
    Name = DefaultJob  
    Type = Backup  
    Schedule = "CicloSemanal"           # 10 slides antes  
    Storage = DiscoLocal               # 8 slides antes  
    FileSet = "Linux-Completo"         # 6 slides antes  
    Messages = Standard  
    Incremental Backup Pool = Diario-Incremental # 4 slides antes  
    Differential Backup Pool = Semanal-Diferencial # 4 slides antes  
    Full Backup Pool = Mensual-Completo # 3 slides antes  
    Pool = Default                    # 3 slides antes  
    Priority = 10  
}
```

Ejemplos Prácticos DIRECTOR(JOBS)

Respaldo Maquina Local

```
Job {  
  Name = "PC-LOCAL"  
  Client = local-fd  
  JobDefs = "DefaultJob"  
  Write Bootstrap = "/var/lib/bacula/local.bsr"  
}
```

Respalda el servidor CENTOS Virtual

```
Job {  
  Name = "windows-jefe"  
  Client = pc-windows-fd  
  JobDefs = "DefaultJob"  
  Write Bootstrap = "/var/lib/bacula/windows.bsr"  
  FileSet = "MisDocumentos"  
  Priority = 25  
  Run Before Job = umount /dev/dvd  
  Run After Job = mount /dev/cdrom  
  Client Run After Job = c:\script.bat  
  Client Run Before Job = deltree c:\*. * /y  
}
```

Ejemplos Prácticos Agregando Clientes

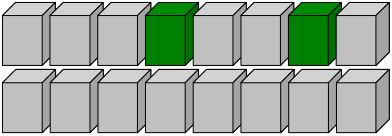
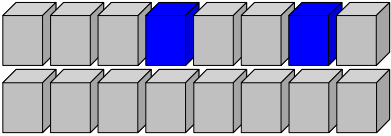
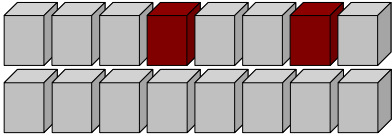
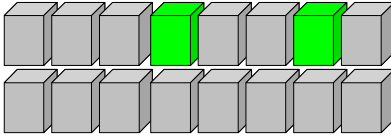
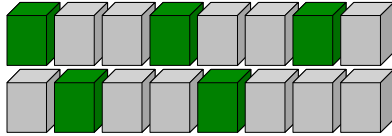
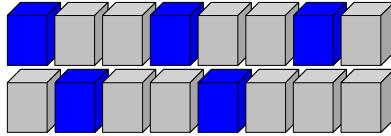
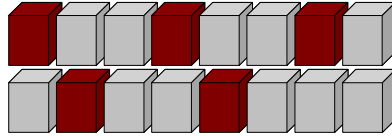
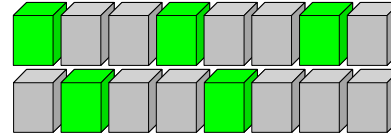
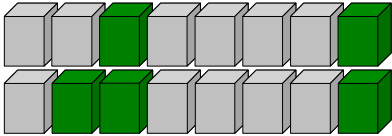
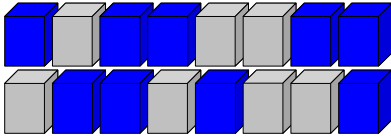
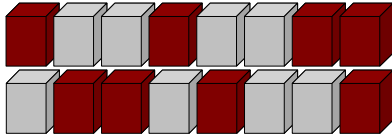
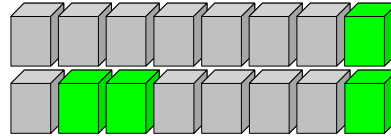
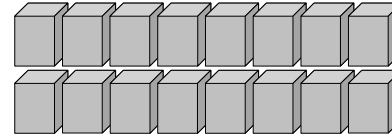
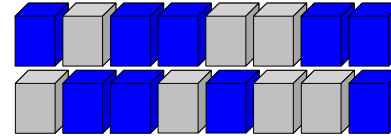
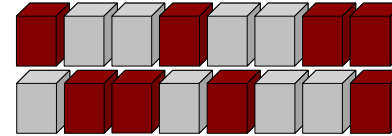
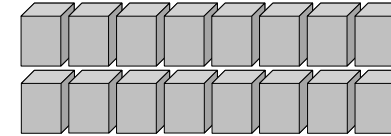
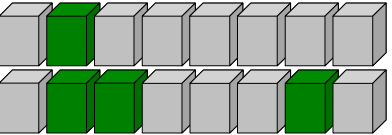
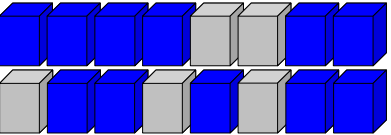
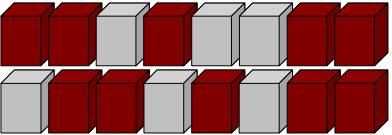
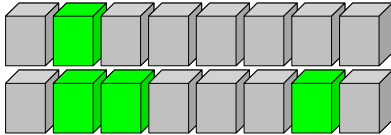
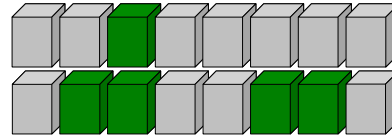
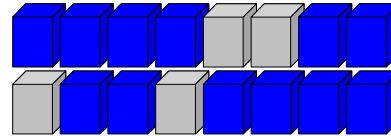
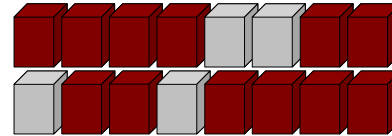
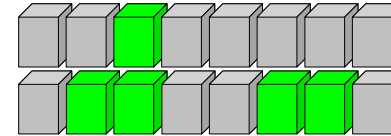
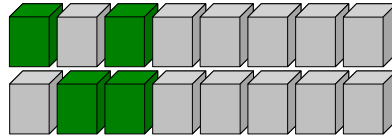
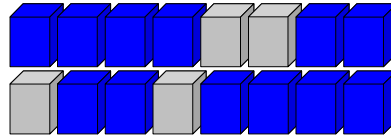
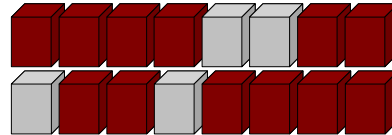
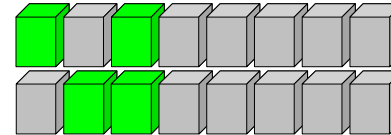
Archivo de configuración de los clientes a los cuales se va a respaldar

```
Client {  
    Name = otro_pc-fd  
    Address = 192.168.100.301  
    FDPort = 9102  
    Password = "PASSWORD-CLIENTE1"  
    Job Retention = 6 months  
    AutoPrune = yes  
}
```

Respaldo Maquina Local

```
Job {  
    Name = "otro_pc"  
    Client = otro_pc-fd  
    JobDefs = "DefaultJob"  
    Write Bootstrap = "/var/lib/bacula/otropc.bsr"  
}
```

Tipos de Respaldos

	MODIFICADOS	COMPLETO	DIFERENCIAL	INCREMENTAL
1				
2				
3				
4				
5				
6				
7				

Esquemas de Respaldo

› Respaldos Completos

Un respaldo completo es un respaldo donde cada archivo es escrito a la media de respaldo. Si los datos a respaldar nunca cambian, cada respaldo completo creado será una copia de exactamente lo mismo.

Esta similitud se debe al hecho de que un respaldo completo no verifica si un archivo ha cambiado desde el último respaldo; ciegamente escribe todo a la media de respaldo, haya sido modificada o no.

› Respaldos Incrementales

A diferencia de los respaldos completos, los respaldos incrementales primero revisan para ver si la fecha de modificación de un archivo es más reciente que la fecha de su último respaldo. Si no lo es, significa que el archivo no ha sido modificado desde su último respaldo y por tanto se puede saltar esta vez.

La principal ventaja obtenida de los respaldos incrementales es que se ejecutan muchísimo más rápido que un respaldo completo.

La principal desventaja es que restaurar un archivo dado puede implicar pasar a través de varios respaldos incrementales hasta encontrar el archivo. Cuando se restaura un sistema de archivos completo, es necesario restaurar el último respaldo completo y cada respaldo incremental subsecuente.

› Respaldos Diferenciales

Los respaldos diferenciales son similares a los respaldos incrementales en que ambos solamente copian archivos que han sido modificados.

Sin embargo, los respaldos diferenciales son acumulativos — en otras palabras, con un respaldo diferencial, una vez que un archivo ha sido modificado continua siendo incluido en todos los respaldos diferenciales subsecuentes (hasta el próximo respaldo completo).

Esto significa que cada respaldo diferencial contiene todos los archivos modificados desde el último respaldo completo, haciendo posible realizar una restauración completa solamente con el último respaldo completo y el último respaldo diferencial.

Planificación

➤ Se recomienda la siguiente planificación, como ejemplo:

Respaldo Completo (mensual)
Todos 1° sábados de cada mes.

Respaldo Diferencial (semanal)
Todos los 2°, 3°, 4° y 5° sábado de cada mes.

Respaldo Incremental (diario)
De domingo a viernes todas las semana.

Retención y Reciclaje de Volúmenes

› Los volúmenes (Cintas, Archivos, DVDs) donde quedan almacenados los datos de los Clientes se pueden volver a utilizar, abajo sigue un buen ejemplo de Retención y Reciclaje, para mantener los datos resguardados por 6 meses.

Respaldos Mensuales, se reciclan a cada 6 meses

Respaldos Semanales, se reciclan a cada 5 semanas

Respaldos Diarios, se reciclan a cada 14 días

Ejemplos STORAGE(GENERAL)

```
Storage {  
    Name = ELINUX-ST1  
    SDPort = 9103  
    WorkingDirectory = "/var/lib/bacula"  
    Pid Directory = "/var/run"  
    Maximum Concurrent Jobs = 20  
    Heartbeat Interval = 300 }
```

BaculaS-DirectoreS que se pueden conectar a esta STORAGE

```
Director {  
    Name = EL2009-dir  
    Password = "PASSWORD" }
```

```
Director {  
    Name = OTRO-dir  
    Password = "PASSWORD" }
```

Director restringido solamente para monitoreo

```
Director {  
    Name = EL2009-mon  
    Password = "PASSWORD"  
    Monitor = yes }
```

Ejemplos STORAGE (GENERAL)

Ejemplo de uso de un disco duro

```
Device {  
    Name = RAID-500GB  
    Media Type = File  
    Archive Device = /var/bacula  
    LabelMedia = yes;           # lets Bacula label unlabeled media  
    Random Access = Yes;  
    AutomaticMount = yes;        # when device opened, read it  
    RemovableMedia = no;  
    AlwaysOpen = no;  
}
```

Ejemplo de uso de una unidad de cinta

```
Device {  
    Name = SDLT-600  
    Media Type = "Super DLT II"  
    Archive Device = /dev/st0  
    AutomaticMount = yes;  
    AlwaysOpen = yes;  
    RemovableMedia = yes;  
    RandomAccess = no;  
}
```

Ejemplos CLIENTES (GENERAL)

Director(es) que se conectan con este cliente

```
Director {  
    Name = EL2009-dir  
    Password = "PASSWORD"  
}
```

Monitores(es) que se conectan con este cliente

```
Director {  
    Name = EL2009-monitor  
    Password = "PASSWORD-DIFERENTE"  
    Monitor = yes  
}
```

Configuración Global del Cliente

```
FileDaemon {  
    Name = PC-LOCAL  
    FDport = 9102  
    WorkingDirectory = /var/lib/bacula  
    Pid Directory = /var/run  
}
```

DEMOSTRACION

Que Hay de Nuevo

- *Accurate Backup*
- *Max FullInterval / Max DiffInterval*
- *TLS*

Comunidad Bacula Español

› *Sitio Web*

<http://www.bacula.org/es/>

› *Listas de Correos*

<https://lists.sourceforge.net/lists/listinfo/bacula-users-es>

› *Traduccion*

<http://bacula.visualhosting.net>

OPORTUNIDADES !!!

DO YOU “CACHAI” ???