

# ***Departamento de Informática***

## ***Planificación***

**Jordi Calopa Bosch**

Revisión 2024.01

## Sumario

1. Infraestructuras tecnológicas.....	4
1.1 Interna.....	4
1.2 Externa en la nube.....	4
2. Infraestructura interna.....	5
2.1 Servidores.....	5
Servidor de base de datos.....	5
Servidor NAS.....	6
Servidor de correo.....	6
Servidor web.....	6
Servidor proxy.....	6
2.2 Estaciones de trabajo.....	7
Sistemas operativos de código abierto.....	7
Sistemas operativos de código cerrado.....	7
2.3 Topología de red.....	8
Topologías.....	8
Ancho de banda.....	8
2.4 Inventario de hardware.....	9
2.5 Software.....	11
Aplicaciones de usuario.....	11
3. Selección del lenguaje de programación.....	12
3.1 Factores técnicos.....	12
3.1.1 Rendimiento de los programas.....	12
3.1.2 Nivel del lenguaje de programación.....	12
3.1.3 Entorno de desarrollo (IDE).....	13
3.1.4 Claridad del código.....	13
3.1.5 Conectividad con bases de datos.....	13
3.1.6 La interfaz gráfica de usuario (GUI).....	13
3.1.7 Portabilidad.....	13
3.2 Factores organizativos.....	14
3.2.1 Disponibilidad de técnicos en mercado laboral.....	14
3.2.2 Disponibilidad de información en red.....	14
3.3 Lenguajes de programación.....	15
Lenguaje Python.....	15
Lenguaje C.....	15
Lenguaje SQL.....	16
Lenguaje Bash.....	16
4. Seguridad.....	17
4.1 Seguridad física de los equipos.....	17
4.2 Copias de seguridad.....	18
4.3 Contraseñas de acceso.....	18
4.4 Antivirus y conexión de equipos externos.....	19

---

4.5 Software externo.....	19
4.6 Documentación.....	19
4.7 Sistema de gestión de documentos.....	19
5 Roles en el centro de proceso de datos.....	20
5.1 Estudios.....	20
Programador.....	20
Analista Orgánico.....	20
Analista Funcional.....	21
Administrador de Sistemas.....	21
5.2 Explotación.....	22
Grabador.....	22
Operador de consola.....	22
Técnico en microinformática.....	22
Administrador de bases de datos.....	22
Encargado de explotación.....	23
5.3 Organigrama.....	23
5.4 El centro de Atención al Usuario (CAU).....	24
Prioridades de las incidencias y/o mantenimientos.....	24
Registro de incidencias, mantenimientos y desarrollos.....	25
Indicadores de calidad del servicio técnico.....	26
6. Ciclo de vida de un proyecto y roles.....	27
6.1 Análisis de requisitos y diseño funcional.....	27
6.2 Diseño de programas y codificación.....	28
6.3 Implementación en producción.....	29
6.4 Mantenimiento.....	30
Anexo I - Ejemplos de CPD.....	31
Ordenador central y estaciones no inteligentes.....	31
Ordenador central y redes de área local.....	32
Red local con servidores de BD y Correo.....	33
Anexo II. Glosario de términos y abreviaturas.....	34
Bibliografía.....	36

---

## 1. Infraestructuras tecnológicas

Durante la fase de planificación de departamento de informática, una de las primeras opciones a analizar debería ser la dicotomía entre trabajar principalmente en las propias instalaciones informáticas internas o derivar, total o parcialmente, los recursos a la nube.

La decisión que se tome repercutirá directamente en la seguridad de la información, en la disponibilidad de acceso, en el dimensionamiento del hardware, en el software a utilizar y en los recursos humanos necesarios, para abordar los proyectos con calidad y eficiencia.

### 1.1 Interna

En opinión del autor, esta es la opción más recomendable y también es la opción técnicamente más complicada, pero la única que puede garantizar una seguridad absoluta sobre el acceso a la información de la organización y el control sobre el software utilizado, si es desarrollado por el propio departamento de informática.

En este caso los equipos informáticos internos deberán ser de más alto rendimiento y precisaremos de mayor espacio físico para nuestra instalaciones, no obstante dado la potencia actual de los ordenadores, esto no debería ser ningún problema significativo.

En el siguiente capítulo se ofrece una ampliación más detallada sobre esta opción.

### 1.2 Externa en la nube

Cuando optamos por utilizar la nube para los procesos informáticos, deberíamos tener en cuenta la dependencia que ellos significa tanto del proveedor de los aplicativos, como del proveedor del acceso a internet. Además, en este tipo de infraestructura, no existe una protección real de datos, puesto que la información de nuestra empresa está almacenada en los sistemas de una empresa externa.

Es cierto que los proveedores de servicios en la nube nos garantizan el mantenimiento de los aplicativos, su escalabilidad y la realización de backups periódicos, pero todo ello a costa de perder la seguridad absoluta sobre de nuestra información. Es de dominio público que, en los últimos años, conocidas corporaciones internacionales han comercializado con la información personal de sus clientes.

Sería factible utilizar servidores externos en aquellos casos que la información sea irrelevante para la seguridad de la empresa o bien se desee explícitamente compartir dicha información.

---

## 2. Infraestructura interna

Si optamos por infraestructuras internas o mayoritariamente de carácter interno, una de las tareas más importantes consiste en elegir las y dimensionarlas correctamente, adecuadas a la actividad de la organización para la que darán servicio.

La elección de los sistemas informáticos, tanto de software como de hardware, es una delicada labor que tendrá una importante repercusión en el futuro de la organización y deberá ser evaluada correctamente para poder dar el mejor servicio posible, dentro del presupuesto económico disponible.

En este orden de ideas, será esencial definir “a priori” donde se almacenará la información de la empresa, la capacidad de almacenamiento necesaria, cuales serán los sistemas operativos a utilizar, los sistemas de gestión de base de datos, los servidores necesarios, la potencia de los procesadores, el tipo de protocolos de comunicación y los anchos de banda de las líneas de comunicaciones, entre otros.

### 2.1 Servidores

La cantidad y tipo de los servidores depende directamente de la actividad de la empresa, del volumen de información a tratar y de la cantidad de usuarios que manejarán dicha información.



En aquellas organizaciones que sea preciso dar un servicio permanente (hospitales, hoteles ...etc), los equipos servidores deberían estar duplicados e implementar un sistema de mirroring de alta disponibilidad que permita arrancar inmediatamente el servidor alternativo en caso necesario.

### Servidor de base de datos

La mejor opción es disponer de un servidor de base de datos dedicado, interno en el CPD, y atendiendo únicamente a criterios de calidad, el autor se inclina favorablemente por servidores midrange, con sistemas operativos como VSE y/o OS400 y sistema de gestión de base de datos DB2. Entornos de alta calidad y muy difícilmente vulnerables desde el exterior.

Una opción alternativa, de menor coste, es utilizar el SGBD MySQL bajo servidores GNU/Linux. En cualquier caso no es recomendable almacenar las bases de datos de la organización en servidores externos.

## **Servidor NAS**

Como en el caso anterior, estos sistemas de almacenamiento conectados a red, es más recomendable implementarlos internamente en la propia red local.

Toda la información que no esté inscrita en la base de datos y deba ser compartida entre los usuarios, puede ser almacenada en este tipo de servidores, que permite compartir información y facilita la realización de copias de seguridad de la información.

## **Servidor de correo**

Existen múltiples aplicaciones de software para implementar un servidor de correo. Es posible utilizar el servicio gratuito de un proveedor externo, si la titularidad de las cuentas de correo es de la propia organización, no de los usuarios que las utilizan.

## **Servidor web**

Para almacenar y suministrar las páginas web podemos implementar un servidor bajo GNU/Linux con el software "Apache web HTTP server" en nuestra propia red.

Para este caso cabe considerar la opción del hosting como una opción interesante por su relación calidad/precio, en los referidos a páginas de presentación de información.

## **Servidor proxy**

El servidor proxy se utiliza de puerta de enlace entre la red local y la red exterior. Su utilidad consiste en que permite limitar determinados accesos a internet y actúa de cortafuegos, además oculta las direcciones IP de los equipos de la LAN al mundo exterior.

Este servidor no encripta la información y existen formas alternativas de proteger la red local interna; además de que es posible solicitar este servicio adicional, o uno equivalente, a nuestro proveedor de servicios de internet.

## **2.2 Estaciones de trabajo**

Personalmente me decantaría por la utilización de ordenadores con sistemas de código abierto y solamente en los casos que, por motivos técnicos, fuera estrictamente necesario utilizaría sistemas cerrados.

### **Sistemas operativos de código abierto**

Los sistemas operativos de código abierto son aquellos en los que, además de los programas ejecutables, se facilita el código fuente de los mismos. Generalmente son gratis y su mantenimiento acostumbra a ser realizado por la comunidad, en la red.

Los sistemas GNU/Linux y sus distribuciones derivadas ofrecen una amplia gama de software que permite cubrir todas las necesidades habituales y disponen de infinidad de aplicaciones de software libre complementarias que hacen de estos sistemas desktop una potente herramienta de trabajo.

El navegador Firefox, el cliente de correo Thunderbird, la suite de ofimática LibreOffice, el editor de imágenes Gimp, el software de control remoto Anydesk o el gestor de máquinas virtuales Virtualbox, son algunas de las excelentes herramientas disponibles para estos sistemas abiertos.

Concretamente la distribución Debian ofrece unas características y una fiabilidad mejores que otros sistemas operativos cerrados y de pago. El presente documento se ha generado con aplicaciones disponibles en el sistema operativo Debian, y si no está mejor realizado no es debido a las limitaciones del sistema, sino a las del autor.

### **Sistemas operativos de código cerrado**

Los sistemas operativos de código cerrado son aquellos en los que no es posible ni visualizar ni modificar el código fuente de los mismos. Su adquisición es de pago y el mantenimiento lo realizan exclusivamente las empresas propietarias.

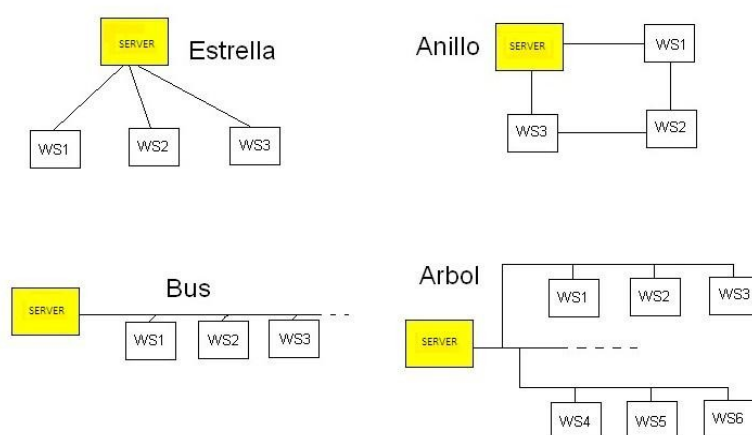
## 2.3 Topología de red

### Topologías

Existen diversas topologías de red, de entre las cuales destacaría la de “árbol” por la facilidad que ofrece en ser segmentada y acotar posibles fallos, además de ser una de las más flexibles para instalar y mantener el cableado de red.

Es recomendable la utilización del standard de red “ethernet” por su fiabilidad y por la facilidad en la adquisición de dispositivos y drivers.

### Topologías de Red



JCB - 2003

### Ancho de banda

En la mayoría de las pymes, la cantidad de usuarios conectados concurrentemente no acostumbra a superar las pocas decenas, por lo que el “ancho de banda” de las líneas actuales no debería suponer ningún problema.

Es suficiente con contratar un proveedor que nos garantice una razonable velocidad de subida y bajada en internet, que es la que está facilitando las fibras ópticas actuales, si bien es cierto que el coste de este servicio es muy superior al de las líneas domésticas, en las cuales no está garantizada la velocidad nominal de la línea.



---

## 2.4 Inventario de hardware

Teóricamente todos los equipos de informática están registrados y controlados, por el departamento de administración, como activos fijos de la empresa. No obstante en algunas ocasiones están incluidos en familias o agrupados de forma que el nivel de detalle que se obtiene no es suficiente para llevar un control exhaustivo del parque informático.

En este caso es necesario disponer de un inventario de hardware interno en el departamento, con un nivel de detalle suficiente que permita saber en todo momento donde se encuentra el material informático, el coste del mismo y su vida útil real; además es muy interesante llevar un seguimiento del material fungible y el uso que se da del mismo.

Una aplicación que nos permita llevar el seguimiento indicado, podrá dar información detallada de aquellos elementos que sufren mayor desgaste, en que ubicaciones se produce dicho desgaste y la repercusión económica que ello conlleva.



En un sistema informático el **hardware** es, según la RAE, *“Un conjunto de aparatos constituido por una computadora y sus periféricos”*.

Seguidamente detallamos una lista de los elementos del parque informático de los que es recomendable llevar un seguimiento.

### Ordenadores

En este grupo entrarían las torres, los portátiles y las tablets, que podemos considerar como un conjunto de procesador, placa base, memorias y periféricos integrados.

### Periféricos de entrada

Cámaras, Gamepads, Joysticks, Lápices ópticos, Micrófonos, Ratones, Scanners y Teclados

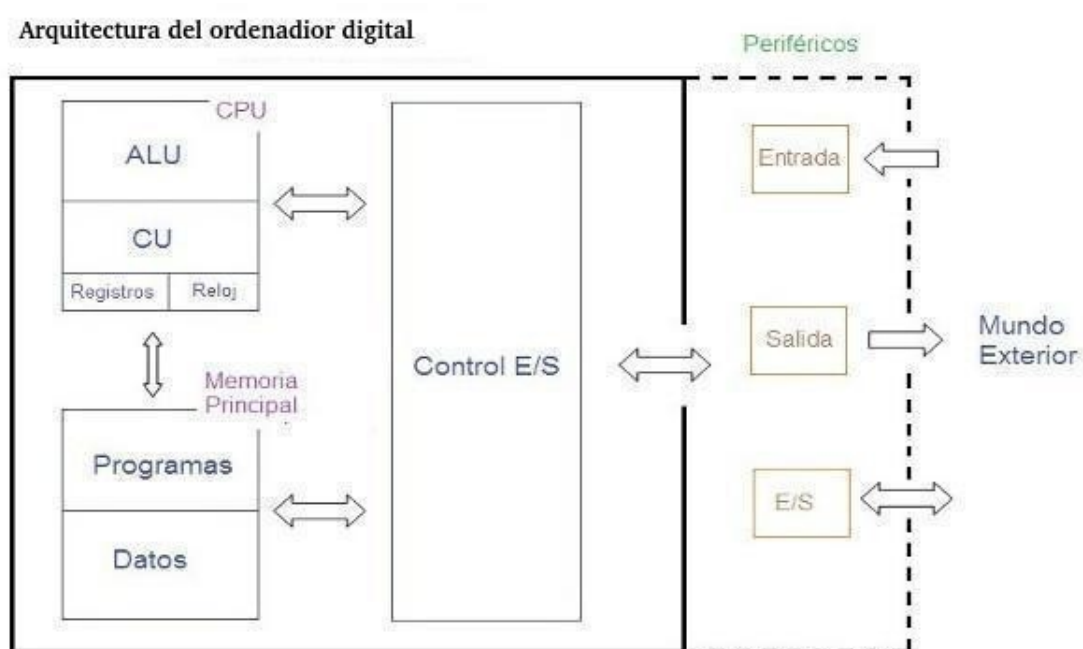
### Periféricos de salida

Altavoces, Impresoras y Plotters

## Periféricos de entrada y salida

Cables , Faxes, Hubs, Memorias USB, Modems , Pantallas, Routers, Switches, Unidades de cinta y Unidades de discos.

Entendiendo por entrada la dirección y sentido que va desde el mundo exterior al ordenador y por salida el sentido contrario, como se muestra en el siguiente gráfico.



JCB - 2003

---

## 2.5 Software

### Aplicaciones de usuario

Existe una gran cantidad de aplicaciones de software libre que permiten desarrollar la mayor parte del trabajo administrativo.

No obstante cuando nos referimos a las aplicaciones para el tratamiento de la información de la actividad principal de nuestra organización es posible que precisemos de un software más específico, para lo que disponemos de varias opciones: Comprar un software en el mercado que cubra las necesidades específicas, desarrollarlo a medida en nuestro propio departamento o buscar un software libre.

- La primera opción (compra) permite disponer del software inmediatamente y no precisamos de un equipo de desarrollo propio para ello, pero no dispondremos de un control del software, ni una garantía absoluta de las funciones que este realiza (p.e. backdoors). Además es habitual que las licencias de software requieran de pagos periódicos para su mantenimiento.
- La opción de desarrollar el software en el propio departamento garantiza el control del software y sus versiones, la adaptación precisa a los requisitos y la garantía absoluta de la funcionalidad que realiza, todo ello con un coste algo más elevado al inicio, pero con independencia de proveedores externos.
- En este punto es preciso contemplar también la posibilidad de utilizar software libre que pudiera acoplarse a nuestras necesidades y a la vez modificable por ser de código abierto. Esta opción, difícil de conseguir, ofrece las ventajas descritas en las dos opciones anteriores.

Personalmente abogaría para que los programas de tratamiento de las bases de datos centrales se desarrollaran en el propio departamento. No obstante aquellas aplicaciones específicas para determinadas estaciones de trabajo podrían ser externas, siempre que dispongan de las herramientas necesarias para la interconexión con las bases de datos centrales.

En el caso de precisar software externo, este software debe ser legalmente adquirido; bien sea por que se opta por software libre, bien sea porque se adquieren las licencias correspondientes. La utilización de software propietario sin la licencia correspondiente denota que la empresa carece de interés en cumplir con la legislación vigente y esto no es un buen indicador.

---

## 3. Selección del lenguaje de programación

Resulta difícil establecer la cantidad exacta de lenguajes de programación existentes, si bien en una primera estimación podríamos decir que son algo más de seiscientos, lo que es, en sí mismo, una dificultad intrínseca al proceso de selección del lenguaje de programación.

Así mismo, en dicho proceso, incurren diversos factores que condicionan dicha selección; Desde aspectos puramente técnicos hasta otros de carácter organizativo y económico.

En este capítulo abordaremos algunos de estos aspectos importantes durante la toma de decisión y al final mostraré algunas posibles opciones, disponibles en los sistemas actuales.

### 3.1 Factores técnicos

#### 3.1.1 Rendimiento de los programas

La velocidad de ejecución debería ser, a priori, uno de los primeros factores a considerar. No obstante, la compleja realidad de las instalaciones informáticas demuestra que no siempre es posible elegir atendiendo prioritariamente a este criterio.

A título de ejemplo citaré que el sistema SAP/R2 fue desarrollado con lenguaje assembler, que es el lenguaje que facilita la mayor velocidad de ejecución de las aplicaciones y permite desarrollar cualquiera de las funcionalidades posibles, dentro de un sistema informático, pero como es evidente el tiempo de desarrollo y los costes inherentes lo convierten en un lenguaje prohibitivo para la mayoría de las empresas.

Observamos pues, que el lenguaje seleccionado deberá ofrecer un equilibrio entre el rendimiento durante la ejecución y la velocidad en el desarrollo de programas. Además, dadas la potencia y la capacidad de almacenamiento de equipos informáticos actuales, creo conveniente destacar que durante la fase de codificación debería primar la calidad del código fuente, por encima de la velocidad de ejecución.

#### 3.1.2 Nivel del lenguaje de programación

Son considerados lenguajes de bajo nivel aquellos que cuyas instrucciones pueden controlar directamente el hardware de las computadoras, como el lenguaje máquina o el assembler, cuyo nivel de abstracción es bajo, dado que sus instrucciones se corresponde casi directamente con las del procesador. Son lenguajes muy potentes y rápidos en tiempo de ejecución, pero requieren de un gran esfuerzo para el desarrollo de programas.

Los lenguajes de alto nivel, son aquellos cuyo nivel de abstracción es elevado y el lenguaje puede expresar los algoritmos de forma inteligible para ser humano, dado que sus

---

sentencias más cercanas al conocimiento humano. No son tan potentes, en tiempo de ejecución, pero permiten el desarrollo de aplicaciones de forma más eficiente.

Parece lógico pensar que el lenguaje seleccionado, debería ser uno de los considerados de alto nivel, para facilitar al máximo las tareas de programación.

### **3.1.3 Entorno de desarrollo (IDE)**

Un aspecto que influye fundamentalmente en el tiempo de desarrollo programas es el entorno de desarrollo integrado (IDE) disponible para lenguaje concreto. Disponer de un buen IDE facilita la labor programación, reduciendo considerablemente el tiempo empleado programación y, por ende, los costes de desarrollo.

### **3.1.4 Claridad del código**

Otro aspecto a considerar, directamente relacionado con el desarrollo de programas, es la claridad del lenguaje, esencial si el mantenimiento de programas es realizado por un equipo de multidisciplinar.

Además de la anotaciones internas, que puede y es recomendable que existan en los programas, la claridad y legibilidad del propio código es un factor determinante en la comprensión del programa y una ayuda inestimable para los programadores, durante el mantenimiento de programas.

### **3.1.5 Conectividad con bases de datos**

Es imprescindible que el lenguaje de programación seleccionado disponga de librerías para la interconexión con el sistema de gestión de base de datos que pensemos utilizar.

### **3.1.6 La interfaz gráfica de usuario (GUI)**

El lenguaje seleccionado debería disponer de una interface gráfica de usuario (GUI) amigable, que facilitara la operación de los ordenadores en los que gestionará la información de las bases de datos.

Este aspecto condiciona la selección del nuevo lenguaje, dado que es importante causar el menor impacto posible, en la utilización de la nueva aplicación por parte de los usuarios. A pesar de ello, es seguro que los usuarios deberán realizar inevitablemente un esfuerzo de adaptación para la utilización de los nuevos programas.

### **3.1.7 Portabilidad**

Sería muy interesante que, el lenguaje de programación seleccionado, dispusiera de versiones para diversas plataformas tecnológicas, dado que ello aumentaría las posibilidades técnicas del departamento y eximiría de la obligatoriedad de utilizar un único sistema operativo, como sucede con algunos lenguajes de programación.

## **3.2 Factores organizativos**

Hasta aquí, hemos anotado los aspectos que deberían ser considerados, desde el punto de vista técnico. Seguidamente intentaremos analizar otros factores, que también influyen sustancialmente en esta decisión y merecen ser estudiados.

### **3.2.1 Disponibilidad de técnicos en mercado laboral**

Contemplar la disponibilidad de técnicos, que conozcan una determinada tecnología, dentro del mercado laboral, es esencial para la continuidad del equipo técnico informático de la organización.

La ausencia de personal disponible, con los conocimientos necesarios sobre una tecnología en concreto, es una de las causas que podría llevar al colapso en las tareas de desarrollo y mantenimiento de programas.

En una situación económica estable, la dificultad de encontrar técnicos expertos en una determinada tecnología sería directamente proporcional a los costes de contratación. Aunque en la situación actual esto no es significativo, es un aspecto a considerar, si en el futuro se estabiliza el mercado laboral.

### **3.2.2 Disponibilidad de información en red**

El hecho de que un lenguaje de programación sea ampliamente utilizado y conocido en el ámbito informático, facilita el acceso a la información compartida en red.

A diferencia de lo que sucedía años atrás, que la información era facilitada exclusivamente por los fabricantes de forma impresa, en la actualidad la información técnica se facilita de forma electrónica en red, tanto por el fabricante como por los grupos de trabajo, que comparten conocimientos tecnológicos en la web.

Quisiera destacar en este punto, la existencia de indicadores de popularidad de los lenguajes de programación, a partir de los resultados obtenidos en los motores de búsqueda, respecto a cada lenguaje.

Es posible que no sean herramientas infalibles, pero si puede darnos información relevante sobre la facilidad de encontrar información en la red.

### **3.3 Lenguajes de programación**

Un departamento puede abordar múltiples proyectos, las tecnologías evolucionan muy rápidamente y además un técnico puede conocer solamente algunos lenguajes. Por ello creo que lo más factible es saber con exactitud y de antemano que lenguajes serán utilizados como herramienta principal, aunque no sea en modo exclusivo. Seguidamente indicamos algunos de los lenguajes especialmente útiles.

#### **Lenguaje Python**

- El lenguaje de programación Python dispone gran variedad de librerías de funciones, entre las que cabe destacar las de entrada de información, cálculo numérico, análisis de datos y presentaciones gráficas de datos estadísticos, así como las librerías de conexión con las bases de datos de SQL-Server, entre otras muchas.
- Es un lenguaje de alto nivel, que se considera de fácil aprendizaje y que permite tanto la programación imperativa, como la funcional y la orientada a objetos.
- Es interpretado, pero dispone de utilidades para compilación, así como de varios GUI existentes, como por ejemplo Tkinter que viene instalada en el propio lenguaje, o bien PyQt5 y wxPython, si bien estas últimas deben ser adquiridas.

#### **Lenguaje C**

C es un lenguaje totalmente estandarizado en los sistemas GNU/Linux y es especialmente útil para la programación de sistemas.

- El código es transportable entre diversos sistemas operativos, incluso algunos sistemas cerrados disponen de compiladores para C.
- Es un lenguaje potente y muy eficiente, es decir de rápidas ejecuciones con poco consumo de recursos, y que interacciona perfectamente con los lenguajes de script de los sistemas operativos.
- Permite y facilita la programación por refinamientos sucesivos (top-down) y permite realizar funciones reutilizables y estandarizadas con un óptimo rendimiento.
- Existe una amplia cantidad de documentación sobre este lenguaje, facilitada por prestigiosas universidades nacionales.

## Lenguaje SQL

Si disponemos de bases de datos relacionales, el lenguaje SQL es de obligado conocimiento, puesto que viene implícito en el propio SGBD.

- Es el lenguaje de tratamiento de información de BD relacionales por excelencia
- Puede ser utilizado en modo nativo o embebido desde muchos otros lenguajes de programación.
- Está muy estandarizado y también existe una amplia cantidad de documentación sobre este lenguaje.

## Lenguaje Bash

Bash es un potente lenguaje de scripting, común en la mayoría de los sistemas GNU/Linux, que permite tanto interpretar las órdenes que recibe directamente del operador como ejecutar procedimientos desde un archivo denominado script (guión).

Si bien no es un lenguaje de programación, permite el tratamiento de variables, la utilización del wildcards, pipelines, parámetros de entrada y dispone de las estructuras de control básicas (secuencia, selección e iteración), que permiten ejecutar tareas planificadas y toda la explotación batch que sea preciso, mediante el uso de scripts.



---

## 4. Seguridad

Cuando nos disponemos a diseñar un sistema de seguridad de todos los datos de nuestra organización deberíamos tener presente que nos referimos a varios grupos de información relevantes:

- Bases de datos de la organización
- Aplicaciones informáticas
- Documentación compartida de usuarios

Toda esta información debe ser resguardada frente a cualquier tipo de evento que pudiera dañarla, bien sea por causas inherentes al propio tratamiento, bien sea por accidentes o causas externas al mismo.

En este capítulo detallamos algunos de los aspectos importantes a tener en cuenta durante la planificación del sistema de seguridad.

### **4.1 Seguridad física de los equipos**

Los equipos servidores deberían estar alojados en una estancia independiente, convenientemente refrigerada y protegida contra incendios, fugas y restantes eventualidades. Aquellos soportes de las copias de seguridad, que se almacenen en la propia sala, deben estar alojados en un armario ignífugo.

Así mismo es preciso prever posibles fallos de suministro. Para minimizar el impacto que podría tener un fallo en el suministro eléctrico del departamento existen varios métodos entre los que podemos destacar:

- Contratar dos compañías de suministro independientes de forma que si falla el suministro principal automáticamente arranque el suministro de la segunda.
- Disponer de un generador eléctrico que arranque automáticamente si falla el suministro principal.

En ambos casos nos referimos al suministro general de todos los sistemas, incluidas las estaciones de trabajo, ubicadas fuera de la sala de servidores. También son recomendable sistemas de alimentación ininterrumpida que además realizan la función de estabilizadores de señal.

El acceso a la sala de servidores debería ser restringido, permitiendo el acceso únicamente a las personas que realizan la administración y mantenimiento de dichos sistemas.

## **4.2 Copias de seguridad**

Diariamente (antes del inicio de la jornada) deberíamos realizar una copia de seguridad total de la información, sobre un tipo de soporte (cinta o disco) que permita ser almacenado fuera de las instalaciones de la propia organización, ubicado en lugar absolutamente seguro, cubriendo de este modo los supuestos de catástrofe: fuego, inundación y sabotaje, entre otros.

Para ello podríamos disponer de siete soportes, sobre los que realizar la copia de seguridad, a los que externamente denominaremos L, M, X, J, V, S, D que identifican los días de la semana respectivamente, y cada día realizaremos el volcado sobre la unidad correspondiente. La nomenclatura interna que reflejará el contenido de la copia, debería incluir información de la fecha en la que esta se realiza (p.e. Backup\_YYYYMMDD).

El soporte del día de la fecha se encontrará en el centro de proceso de datos y los restantes 6 en ubicación de máxima seguridad; Diariamente (después del final de la jornada), se deberá llevar la unidad de respaldo actual al lugar seguro y recuperar la más antigua de ellas, para que vuelva al CPD y permita seguir el ciclo de backups.

En ningún caso deberemos utilizar un único soporte para ir realizando la copia de seguridad sobre el mismo, porque en caso de accidente durante el volcado perderíamos toda la información.

El sistema descrito asegura que, en la peor de las circunstancias, se perdería la información del día en curso. No obstante esto se puede paliar si disponemos de servidores duplicados con un sistema de mirroring de alta disponibilidad. Opcionalmente, cabe estudiar la posibilidad de realizar copias extras semanales o mensuales, siguiendo la misma metodología descrita anteriormente.

## **4.3 Contraseñas de acceso**

En algunas empresas se observa que las contraseñas de acceso a los sistemas informáticos son de bajo nivel de seguridad o incluso que algunos usuarios cuelgan sus contraseñas con un papel en la propia pantalla y los mas precavidos las guardan en la primera página de su agenda, encima de su mesa de trabajo, con un gran rótulo que indica contraseñas.

La vulnerabilidad de estos sistemas informáticos es tan elevada que incluso disponiendo de un buen sistema de copias de seguridad no quedaríamos protegidos frente a pérdidas de información causadas deliberadamente, bien sea desde el interior de la organización o desde sistemas externos.

Es cierto que nunca existe una fiabilidad absoluta en el control de los accesos a los sistemas, pero una política adecuada sobre el uso de contraseñas robustas y el cambio periódico de las mismas minimiza los riesgos de sufrir accesos indeseados a los sistemas informáticos.

Además debería ser programado el bloqueo de las estaciones de trabajo, al objeto de que cuando un usuario deja temporalmente su equipo, este quede bloqueado de forma automática.

## **4.4 Antivirus y conexión de equipos externos**

Es muy importante disponer de antivirus en las estaciones de trabajo de la red y realizar periódicamente un análisis de posible virus; no obstante, en la medida de lo posible, deberíamos evitar conectar discos externos y dispositivos móviles a nuestros equipos, dado que son una posible fuente de entrada de software malicioso que puede perjudicar nuestra instalación informática.

## **4.5 Software externo**

Todo el software externo debería ser adquirido de forma legal y siempre estar certificado por un fabricante de prestigio. También debemos realizar las correspondiente actualizaciones que, en muchos casos, están orientadas a mejorar la protección de los equipos.

## **4.6 Documentación**

La documentación de programas y procesos, en un departamento de informática, es imprescindible para poder llevar a cabo una correcta labor de trabajo en equipo. No documentar rigurosamente todas las aplicaciones y procesos es un grave error, porque en la medida que la instalación va creciendo cada vez se encuentra más dependiente de la memoria de las personas, que falla más que la de los ordenadores, lo que implica un elevado riesgo para la continuidad del servicio, en caso de catástrofe.

Una buena documentación, guardada digitalmente, es una excelente herramienta de ayuda para la recuperación de información si se produce una pérdida accidental.

## **4.7 Sistema de gestión de documentos**

Sería recomendable disponer de un sistema informatizado para el control de documentos digitalizados y de aquellos documentos que se vayan incorporando progresivamente a dicho entorno.

Inicialmente pensado para la gestión de documentos del departamento de informática, pero previendo la posible extensión a toda la organización si fuera preciso. En este sistema se puede incluir toda la documentación apuntada anteriormente. Además en el sector industrial, sería útil para incluir manuales de funcionamiento y mantenimiento de equipos.

Una de las formas, no la única, de llevar una eficiente gestión de la documentación, es utilizar un SGD, que puede ser adquirido o desarrollado en el propio departamento.

Además, en la medida que registramos todas las aplicaciones, programas, procesos y scripts, el propio SGD nos facilita dinámicamente el inventario de software.

## 5 Roles en el centro de proceso de datos

Al objeto de organizar el trabajo en el centro de proceso de datos, es imprescindible conocer los recursos humanos disponibles. Una vez conocidos, podremos organizar las funciones, en virtud de dichos recursos.

Las dos áreas principales de un centro de proceso de datos se podrían establecer en una de "Estudios" que se correspondería con el área de desarrollo y otra de "Explotación" que se ocupa de la producción del departamento. Destacaremos seguidamente los perfiles técnicos más comunes en cada una de ellas.

En las pequeñas y medianas empresas se da la circunstancia de que cada técnico asume varios de los perfiles del departamento, lo que redundaría en detrimento de la calidad del servicio, pero es una realidad, forzada por criterios económicos, que no se puede obviar.

### 5.1 Estudios

#### **Programador**

Es el encargado de desarrollar el software mediante la correcta codificación del mismo en un determinado lenguaje de programación, basándose en el análisis orgánico preparado a tal efecto.

También realizará las pruebas unitarias al objeto de certificar el correcto funcionamiento de los programas.

#### **Analista Orgánico**

Denominado analista de aplicaciones, es el encargado de interpretar los requisitos funcionales y adaptarlos a la infraestructura (software y hardware) disponible, con el objetivo que se cumpla con las especificaciones funcionales.

Verifica la correcta funcionalidad de los programas desarrollados por el programador, mediante la realización de pruebas y da soporte a los programadores en la resolución de incidencias.

## **Analista Funcional**

También denominado analista de sistemas, es el encargado de modelar los procesos, atendiendo a los requisitos de los usuarios, generando como resultado el análisis funcional pertinente, el cual deberá ser independiente de la tecnología utilizada.

También se encarga de diseñar los sets de pruebas necesarios para verificar la correcta funcionalidad de las aplicaciones, una vez desarrolladas estas.

Se ocupa, así mismo, de dirigir la implementación en producción de los nuevos sistemas y colaborar con explotación en la resolución de posibles incidencias.

## **Administrador de Sistemas**

El administrador de sistemas se encarga de todas aquellas funciones que sean necesarias para el correcto funcionamiento de los sistemas. Indicamos en este momento las más significativas.

Instalación de software base y de las actualizaciones, manteniendo la instalación actualizada con las más modernas herramientas en materia de seguridad y rendimiento.

Generar los códigos de usuario con sus perfiles y asignar los permisos de acceso a las aplicaciones y servicios del sistema informático.

Establecer las políticas relativas a la forma en que se organizan y comparten los archivos dentro de la organización y el tipo de acceso a los mismos.

Monitorear todas las infraestructuras de red, incluido los servidores centrales: rendimiento de las CPU, flujo de datos en la red, capacidad de los dispositivos de almacenamiento y realizar las estadísticas de utilización y rendimiento

Guardar la seguridad de toda la instalación en lo que se refiere a la información sensible planificando las copias de seguridad y controlar los accesos a los sistemas, mediante una correcta política de contraseñas. Debe así mismo instalar y mantener los sistemas antivirus y controlar la instalación de software externo.

Mantener la documentación técnica relativa a los servidores, la red y las estaciones de trabajo, todo ello siempre actualizado. Seguidamente apuntamos la documentación de la que sería deseable disponer:

- Manuales de procedimientos (runbooks)
- Procesos batch y scripts relacionados
- Partes de asistencia técnica, si no disponemos de una aplicación específica
- Mapa de estaciones de trabajo
- Mapa de Ips
- Protocolos de utilización de los ordenadores de red
- Directiva de seguridad
- Referencias breves

## **5.2 Explotación**

### **Grabador**

Es la persona que realiza la grabación y entrada de datos (data entry) en el sistema informático.

### **Operador de consola**

Se encarga de monitorizar el correcto funcionamiento de los sistemas informáticos, las comunicaciones y sistemas remotos. También se ocupa de dar soporte técnico de primer nivel a los usuarios.

### **Técnico en microinformática**

Se ocupa de instalar y mantener los sistemas microinformáticos de la empresa y gestionar las incidencias. Es el encargado de instalar el software común de oficina y aplicaciones específicas de usuario. Se ocupa de dar soporte técnico de primer nivel a los usuarios.

### **Administrador de bases de datos**

Es el encargado de administrar las bases de datos de la organización, sea cual fuere la tecnología utilizada (Relacional, Jerárquica, OO...etc), garantizando el buen funcionamiento de las mismas.

Responsable de la seguridad de la base de datos y de los sistemas de backup necesarios para garantizar el contenido de toda la información en caso de desastre. También se encarga de la gestión de usuarios y sus accesos a las bases de datos.

Así mismo es el responsable de diseñar, dimensionar y cargar las nuevas bases de datos que pudieran ser requeridas por la empresa y se encarga de estudiar cuales son los sistemas de extracción y análisis de datos más adecuados, atendiendo a los SGBD utilizados.

## Encargado de explotación

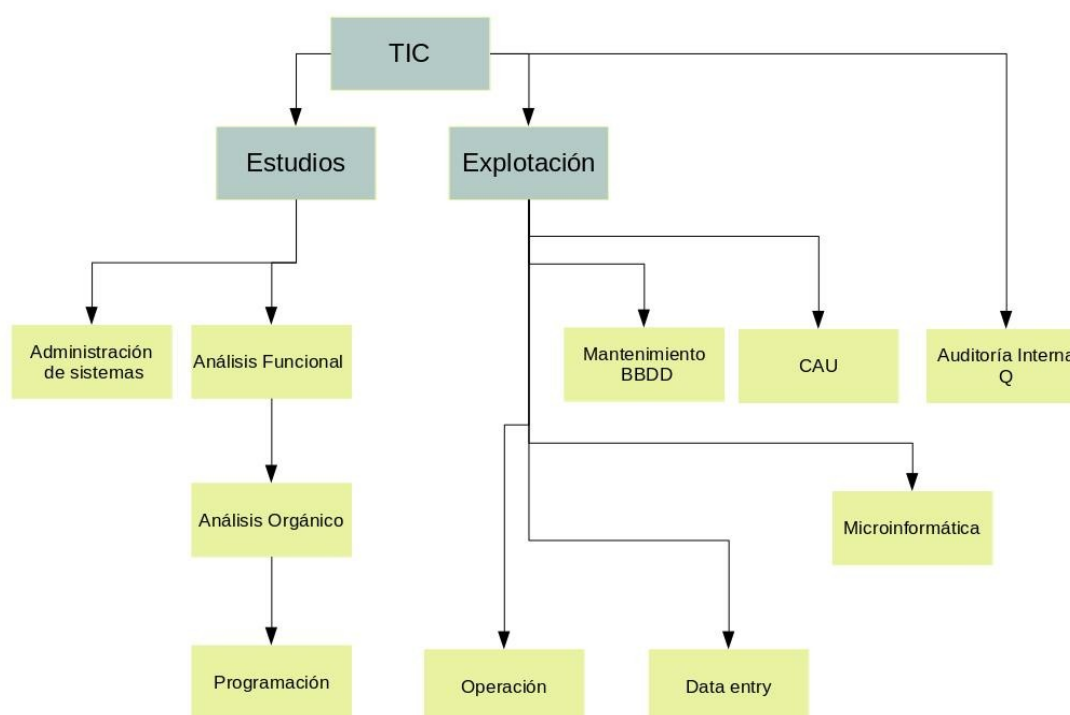
Es el responsable de asegurar el adecuado nivel de servicio de los sistemas informáticos, incluidos los SGBD, así como de supervisar todos los procesos de explotación batch.

También se ocupa de planificar y supervisar el trabajo de los técnicos de microinformática, operadores y grabadores de datos.

Se responsabiliza del centro de atención al usuario (CAU) y da soporte de segundo nivel.

### 5.3 Organigrama

Probablemente existan tantos organigramas como empresas, por lo que presento el siguiente organigrama como un ejemplo genérico de una posible organización.



JCB -1994

## **5.4 El centro de Atención al Usuario (CAU)**

Una de los centros esenciales, del área de producción del departamento de informática es el CAU, por la importancia que este tiene para el conjunto de la organización. El CAU en si mismo no es ninguna unidad productiva, si no un servicio que se presta a los departamentos del front-office de la empresa, para que estos puedan desempeñar su labor con la máxima calidad y eficiencia.

Esto que parece una obviedad, no lo es. Una actitud de servicio hacia todos los usuarios es esencial para diseñar y gestionar un buen servicio técnico porque además de la planificación y de la capacitación de los técnicos, se requiere la mencionada actitud de servicio, hacia todo el equipo humano de la organización.

### **Prioridades de las incidencias y/o mantenimientos**

En algunas organizaciones podría darse la circunstancia de que la mayor parte del tiempo el equipo técnico estuviera dedicado a resolver incidencias y no restara el tiempo necesario para efectuar los nuevos desarrollos y mantenimientos preventivos.

Esto puede ser debido a diversos motivos como por ejemplo que el volumen de incidencias es muy elevado en relación a los recursos humanos disponibles o que es estado de la instalación esté gravemente deteriorado.

Sea cual fuere el motivo es imprescindible realizar una clasificación de los tipos de incidencia y los mantenimientos previstos y establecer unas prioridades para su resolución. Seguidamente mostramos una de las posibles clasificaciones, que definirá el tipo de actuación y tiempo de respuesta, por parte del CAU.

Prioridad 1 - Se asignará a las incidencias de nivel crítico, que inhabilitan el sistema informático y afectan directamente a la producción de la empresa, no permitiendo la continuidad del trabajo productivo (Producción parada - sin solución alternativa).

Prioridad 2 - Se asignará a las incidencias graves, que afectan al sistema informático del usuario, pero existen alternativas para continuar realizando el trabajo, bien sea mediante otros equipos, bien sea con mecanismos alternativos (Producción parada - con solución alternativa).

Prioridad 3 - Se asignará a las incidencias, que no detienen el sistema, aunque impliquen una pérdida grave de efectividad del mismo (Mantenimiento correctivo urgente).

Prioridad 4 - Se asignará a las incidencias leves, que no detienen el sistema, aunque impliquen una pérdida de efectividad del mismo (Mantenimiento correctivo).

Prioridad 5 - Esta prioridad se asignará a aquellas solicitudes de mejora, que no sean propiamente una incidencia del servicio, pero que pudieran llegar a ser un problema en determinadas circunstancias (Mantenimiento preventivo).



Prioridad 6 - Esta prioridad se asignará a aquellas solicitudes que no sean una incidencia del servicio pero que permitan mejorar el sistema en el futuro (Mantenimiento perfectivo).

Prioridad 7 - Estado inicial de entrada. Pendiente de asignación de prioridad.

Esta clasificación (u otra que pudiera ser acordada) debería ser aprobada y respetada por la dirección de la empresa y, por ende, conocida por el conjunto de la empresa, con el objetivo de que toda la organización comprenda los criterios que se siguen en la asignación de prioridades.

Lógicamente las incidencias se atenderán por orden de prioridad (1,2,3...) y dentro del grupo de incidencias con igual prioridad, serán atendidas por orden de entrada (FIFO) en el registro, evitando cualquier tipo de arbitrariedad. En este sentido es esencial la implicación de la dirección para lograr este objetivo.

En este punto, cabe mencionar la importancia de un usuario clave que, con un buen conocimiento de la organización, revise, coordine y canalice todas las solicitudes al objeto de garantizar la adecuada secuenciación de las mismas y evitar duplicidades. El encargado de explotación puede ser uno de estos usuarios clave.

## **Registro de incidencias, mantenimientos y desarrollos**

Independientemente de como se registren las solicitudes, es obvio que deben ser registrados metódicamente para poder proporcionar posteriormente las estadísticas necesarias y la situación de los indicadores de calidad.

Sería recomendable disponer de un aplicativo que permita registrar todas las incidencias, mantenimientos y nuevos desarrollos requeridos; aplicativo que deberá soportar el tratamiento de la información que seguidamente se describe:

### **Registro de cabecera**

OT (orden de trabajo), Usuario y Departamento solicitante, Area geográfica, Tipo (Incidencia/Correctivo/Preventivo/Perfectivo/Nuevo), Nivel de Prioridad, Fecha / Hora de registro, Link documentación, Tick Hardware/Software, Equipo / Marca / Modelo / N<sup>o</sup> serie (si aplica), Programas afectados (si aplica), Descripción y Fecha / Hora de finalización OT

### **Registro de detalle**

OT, Técnico responsable, Fecha/Hora del registro, Tiempo dedicado, Código de servicio / material, Coste del servicio / material y Descripción del trabajo y solución

## Indicadores de calidad del servicio técnico

La única forma de disponer de un control de la calidad de cualquier servicio técnico consiste en disponer de unos indicadores objetivos, y por ende medibles, y que los técnicos registren periódicamente el tiempo y los materiales destinados a cada intervención.

Cabe destacar, en este punto, que los indicadores que seguidamente se detallan tienen una validez relativa en función de la cantidad de recursos humanos disponibles, por lo que se facilitan únicamente a título orientativo, dado que si el equipo humano está infradimensionado carece de sentido revisar unos indicadores que siempre estarán bajo mínimos.

- Cantidad de solicitudes recibidas / tiempo
- Cantidad de solicitudes cerradas / tiempo
- Relación de solicitudes cerradas / recibidas x tiempo
- Distribución de solicitudes / área geográfica (país, región, provincia ...)
- Distribución de solicitudes / departamento (Adm, Mkt, Com, RRHH ...)
- Distribución de solicitudes / tipo de incidencia (Crítica, Grave, Leve)
- Distribución de solicitudes / prioridad (1,2,3...)
- Distribución de actuaciones / técnico
- Tiempo promedio de solución de solicitud (/ Tipo Incidencia, / Prioridad, / Técnico)
- Incidencias durante el servicio
- Solicitudes pendientes

### Nota: Modalidades del soporte técnico

Presencial: Inicialmente optaremos por la modalidad presencial, para atender a los clientes internos, siempre que sea posible.

Remota: En aquellos casos que la incidencia se produzca en otra área geográfica y se trate de una incidencia de software, utilizaremos conexión remota para solucionarla, en la medida de lo posible.

En el soporte remoto y cuando se precise la colaboración del propio usuario para solventar el problema, deberíamos enviar y recibir las indicaciones por correo electrónico, evitando en lo posible la vía telefónica, que acostumbra a ser una fuente de errores en las comunicaciones.

---

## 6. Ciclo de vida de un proyecto y roles

### **6.1 Análisis de requisitos y diseño funcional**

#### **(Analistas funcionales y administradores de bases de datos)**

Durante estas primeras fases, el proyecto puede estar sujeto a modificaciones que puedan surgir de las necesidades del usuario, que no se observaron inicialmente, pero que es preciso incluir por motivos relevantes.

Cabe destacar que esta parte del proceso es esencial para la correcta consecución de las siguientes fases y que los errores u omisiones que pudieran producirse, afectarán al resto del proceso de desarrollo.

El analista debe atender e interpretar correctamente los requisitos del usuario, que es el cliente interno para el que trabaja, diseñando siempre un sistema que cumpla con los requisitos, siendo el mejor de los sistemas posibles, en términos de calidad.

Es imprescindible que el análisis funcional, generado a partir de la toma de requisitos, quede perfectamente documentado y aprobado tanto por el usuario clave de la aplicación como por el analista informático de la misma.

Este resultado del análisis funcional puede quedar plasmado de forma narrativa o bien utilizando cualquier lenguaje de modelado, pero debería contener, como mínimo, la información que seguidamente se describe:

- Detalle de requisitos
- Información de salida deseada
- Modelo de datos
- Componentes del software
- Descripción de procesos administrativos o secuencia de las interacciones

**En cualquier caso el resultado del análisis deberá establecer de forma clara cual es la funcionalidad exacta de la aplicación objeto del estudio.**

---

## 6.2 Diseño de programas y codificación

### (Analistas orgánicos y programadores)

En esta fase se realiza el análisis orgánico del aplicativo y debe ser generada una rigurosa documentación que deberá cumplir dos objetivos concretos:

- Debe ser la herramienta básica de consulta para los procesos de mantenimiento (como lo es la documentación técnica y los esquemas en cualquier mantenimiento de carácter industrial).
- Debe ser suficientemente explícita para que permita regenerar los programas con sus funcionalidades específicas, en caso de necesidad (p.e. en el supuesto de un mantenimiento adaptativo).

Después se aborda la codificación de los programas y la pruebas pertinentes para verificar la correcta funcionalidad del software, antes de su puesta en marcha definitiva en el entorno de producción.

Para la correcta codificación de los programas es necesario que se establezcan unos **estándares de programación** que indiquen cual es la forma óptima de programar, para el departamento en cuestión (programación estructurada, orientada a eventos, orientada a objetos...etc).

Además en dichos estándares se debería incluir, entre otros aspectos, el sistema de **nomenclatura** para bases de datos, tablas, ficheros, programas, scripts y documentos, así como modelos de los algoritmos recomendados para programación y las restricciones, si las hubiere, en términos de codificación.

También es recomendable disponer de **funciones reutilizables**, para aquellos trabajos repetitivos, dado que ahorran muchos errores y tiempo de programación. Lógicamente dichas funciones deben estar correctamente documentadas para facilitar su estandarización dentro del área de desarrollo.

Dadas la velocidad de proceso y la capacidad de almacenamiento de equipos informáticos actuales, creo conveniente destacar en este punto que durante la fase de codificación debería primar ante todo la calidad.

Una herramienta de ayuda para el desarrollo de aplicaciones es disponer de un **diccionario de datos** en el que tengamos registrados todos los programas, scripts, bases de datos, tablas y ficheros, en el que podamos consultar las relaciones entre elementos, de forma que consultando uno de ellos nos indique automáticamente la lista de los elementos con los que interactúa.

## **6.3 Implementación en producción**

### **(Encargado de explotación, administradores y analistas)**

Antes de implementar un programa en el entorno de producción debe ser probado exhaustivamente para depurar posible errores funcionales y de codificación.

Una vez que el software haya sido diseñado, probado y se ha verificado que atiende a los requisitos funcionales es el momento de su lanzamiento, implementándolo en los sistemas de producción.

#### **Programas**

Para el mantenimiento de los programas es muy recomendable disponer de un equipo, bien sea un ordenador o un disco dedicado, en el que se guarden los códigos fuente de los programas implementados en producción.

A partir de este equipo, todos los desarrolladores deberían obtener copias de los programas fuente sobre los cuales efectuar el mantenimiento y posteriormente, una vez realizadas todas las pruebas, devolver el código fuente al equipo central; lógicamente guardando previamente las versiones anteriores.

#### **Bases de Datos**

En lo referente a las bases de datos, cabe destacar en este punto que el escenario ideal sería disponer de tres entornos:

- Un sistema de desarrollo que se utiliza para codificar y realizar las pruebas unitarias de los programas.
- Un segundo sistema de integración, que debería ser una replica del entorno de producción para poder realizar las pruebas de conjunto.
- Por último el sistema de producción real, sobre el que trabajan los usuarios.

Si no fuera posible disponer de los tres entornos indicados, como mínimo deberíamos disponer de dos, uno de desarrollo y otro de producción real; pero en ningún caso desarrollar directamente sobre el entorno de producción.

## **6.4 Mantenimiento**

### **(Analistas y programadores)**

A lo largo de la vida útil de un software, pueden surgir nuevas necesidades que impliquen realizar modificaciones en el mismo.

- **Correctivo:** Cuando se observa alguna disfunción en el software
- **Perfectivo:** Si se precisan mejoras funcionales o se detecta que el rendimiento es mejorable.
- **Preventivo:** Si se desea aumentar la capacidad y/o fiabilidad del sistema, antes de llegar a su propio límite.
- **Adaptativo:** Motivado por un cambio en las plataformas de hardware/software.

En cualquier caso, es evidente que el mantenimiento forma parte del ciclo de vida del software y requiere de los recursos precisos para su correcto desempeño. Como en el caso de la primera implementación, el mantenimiento requiere de una detallada documentación para registrar los cambios producidos en los aplicativos.

En el desarrollo de aplicaciones también es sumamente importante que alguien de la organización realice las funciones de revisión y clasificación de las solicitudes de trabajo, para evitar la duplicidad de tareas (como sucede en algunas obras públicas, que un día abren una zanja para cambiar una canalización y la cierran; al poco tiempo vuelven a abrir una zanja en el mismo lugar para cambiar otra canalización, que estaba al lado de la anterior).

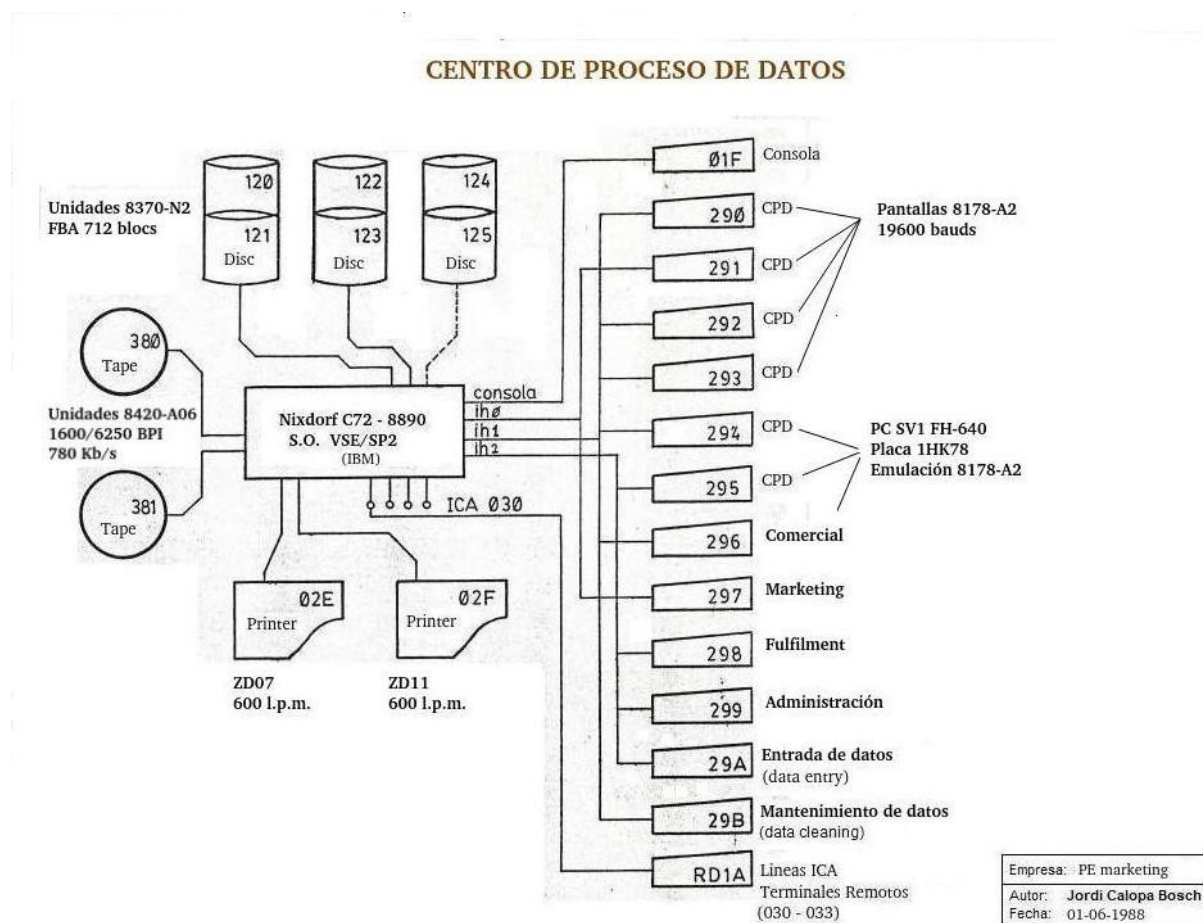
Todas las solicitudes de mantenimiento al CPD, deberían ser canalizadas por medio de correo electrónico o mediante una aplicación online, al objeto de realizar un correcto registro de las mismas y añadirlas a la documentación de las modificaciones que pudieran implicar.

## Anexo I - Ejemplos de CPD

### Ordenador central y estaciones no inteligentes

Este primer ejemplo describe un centro de procesos de datos de una empresa de servicios dedicada al marketing directo.

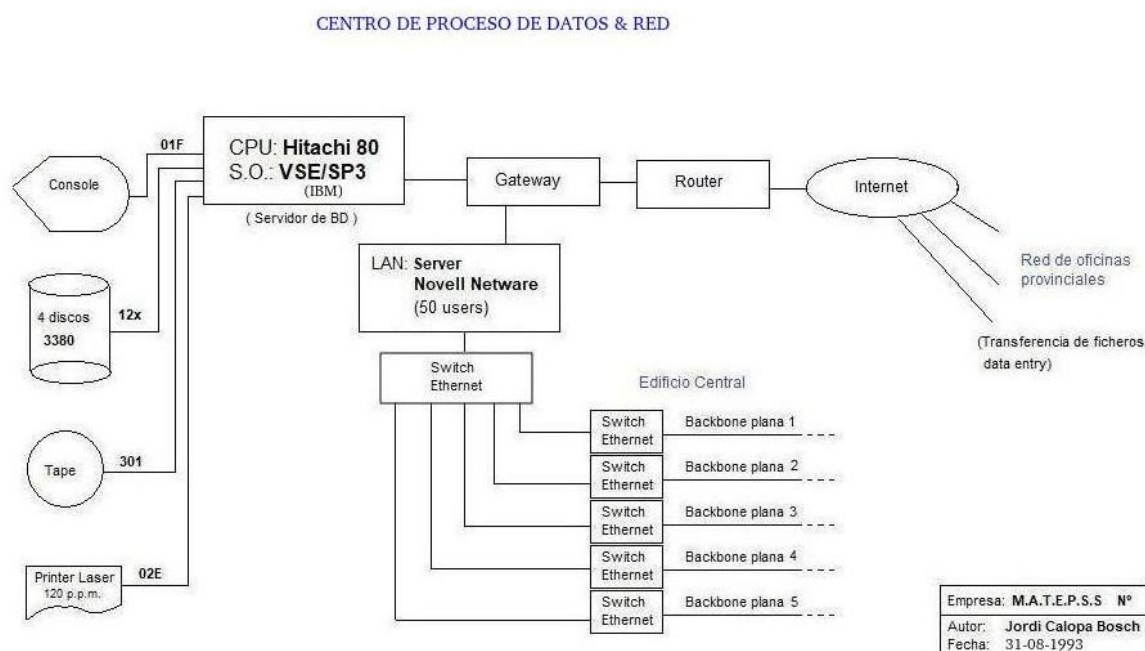
Disponíamos de un ordenador central midrange con sistema operativo VSE/SP2 y un conjunto de terminales no inteligentes, locales y remotos, mediante los cuales se realizaba el data entry y el mantenimiento de las bases de datos, así como la gestión de la organización. El equipo humano de este departamento de informática estaba compuesto por seis personas.



## Ordenador central y redes de área local

Un segundo ejemplo, en el que se muestra un centro de proceso de datos, basado en un entorno midrange, con sistema operativo VSE/SP3 y una red local Novell Network mediante la que todos los departamentos podían acceder al ordenador central, para realizar la gestión de la empresa, que era una mutua de accidentes de trabajo.

Además disponíamos de pequeñas redes locales, en cada oficina provincial, en la que se realizaba la entrada de datos, que posteriormente eran transferidos en batch al ordenador central. El equipo humano estaba compuesto por diez técnicos.

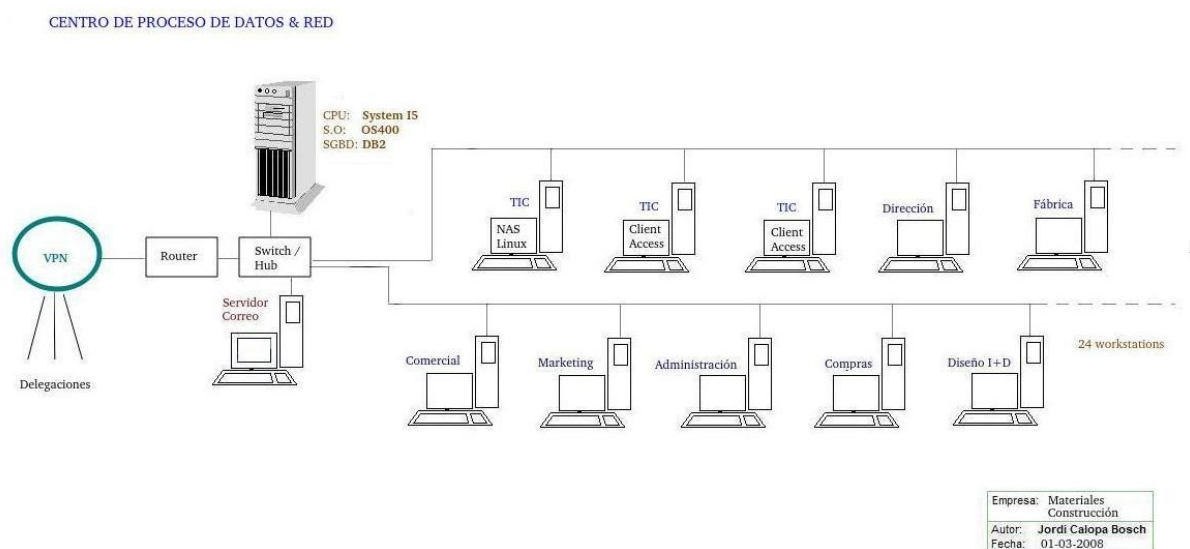




## Red local con servidores de BD y Correo

En este último ejemplo, de una empresa dedicada a construcción de maquinaria industrial, la gestión se llevaba mediante un System I5, con sistema OS/400, integrado en una red local que permitía el acceso al mismo.

También disponíamos de un servidor de correo gestionado internamente en el propio CPD. Mediante una VPN se facilitaba el acceso de los usuarios desde las delegaciones y sus propios domicilios. El equipo humano del departamento lo formaban dos personas.



## Anexo II. Glosario de términos y abreviaturas

**BACKDOOR** - Se denomina “puerta trasera” a determinadas rutinas, insertadas en sistemas y/o aplicaciones, que permiten saltar las medidas de seguridad que el propio software tiene implementadas.

**BACKUP** - Es una copia de seguridad, total o parcial, de la información contenida en el sistema informático, que habitualmente se copia sobre cinta o cualquier otro soporte susceptible de ser almacenado fuera del CPD.

**BASE DE DATOS** - Conjunto de información estructurada y organizada para su rápido acceso, habitualmente almacenada en sistemas informáticos.

**BATCH** - Referido normalmente a los procesos por lotes que se ejecutan en un ordenador y que no interactúan con el usuario, o a lo sumo con el operador del sistema.

**CPD** - Centro de proceso de datos

**DOWNLOAD** - (descargar/bajar) Transferir información desde un servidor de internet al equipo local.

**ETHERNET** - Es un estándar utilizado en redes, que establece las normas lógicas de comunicación definidas por el modelo OSI y además las características físicas del cableado de red.

**HOSTING** - Servicio de almacenamiento de sitios web que facilitan determinadas empresas para el alojamiento y acceso de páginas web.

**INTRANET** - Red interna de una organización basada en los protocolos de comunicación de internet.

**LAN** - (Local Area Network) Red de área local compuesta por servidores, estaciones de trabajos y dispositivos periféricos, circunscrita a un edificio o departamento concreto.

**MAINFRAME** - (Unidad central) Término con el que se designa a grandes computadores utilizados para el procesamiento masivo de información de grandes bases de datos, procesamiento realizado habitualmente en batch.

**MIDRANGE** - Término que se refiere a ordenadores medios, cuya potencia y capacidad estaría a medio camino entre los pequeños ordenadores personales y los grandes computadores mainframe.

**PROGRAMA** - Un programa informático es un conjunto de instrucciones que le indican a un ordenador como realizar una tarea concreta. Normalmente un programa se escribe en un lenguaje inteligible para el programador y posteriormente es convertido, mediante el proceso de compilación, al lenguaje que comprende el procesador.

**ORDENADOR** - Equipo electrónico capaz de manejar grandes volúmenes de información, realizando con dicha información operaciones aritméticas y lógicas a gran velocidad,

habitualmente utilizadas para transformar los datos de entrada (facilitados por el usuario) en la información de salida (esperada por el mismo).

**SAI** - Sistema de alimentación ininterrumpida que permite, en caso de fallo eléctrico, mantener el suministro a los equipos informáticos, por un corto espacio de tiempo. Además también actúa como estabilizador de la señal de salida.

**SAT** - Servicio de asistencia técnica.

**SGBD** - Sistema de gestión de base de datos. Consiste en un conjunto de programas y utilidades que permiten el diseño y mantenimiento de una o más bases de datos.

**SGD** - Sistema de gestión documental.

**SOFTWARE** - Es el conjunto de programas y datos que se procesan en un sistema informático. Cabría distinguir entre el software del sistema, encargado de conseguir el funcionamiento del propio sistema informático, y el software de aplicaciones destinado al tratamiento específico de la información de una organización.

**TIC** - Tecnologías de la información y la comunicación.

**WORKSTATION** - Estación de trabajo. Nombre con el que se designan los ordenadores (no servidores) conectados a una red local o directamente al host.

## Bibliografía

**Cooper, Mendel** (2014). *Advanced Bash-Scripting Guide*. Public Domain.

**Jorba Esteve, Josep & Suppi Boldrito, Remi** (2004). *Administración avanzada de GNU/Linux*. Barcelona: FUOC Fundació per a la UOC.

**Piñeiro Gómez, José Manuel** (2013). *Bases de datos relacionales y modelado de datos*. Madrid: Paraninfo

**Comité Europeo de Normalización** (2007). Indicadores clave de rendimiento del mantenimiento. Norma Europea UNE-EN 15341. Bruselas.

**Real Academia Española de la lengua** (2022) <https://dle.rae.es>