

42  
2ej



**UNIVERSIDAD NACIONAL AUTONOMA  
DE MEXICO**

-----  
ESCUELA NACIONAL DE ESTUDIOS PROFESIONALES  
" ARAGON "

FALLA DE ORIGEN

**SISTEMAS ABIERTOS:  
SUS PROMESAS Y REALIDADES**

**T E S I S**

QUE PARA OBTENER EL TITULO DE  
INGENIERO EN COMPUTACION  
P R E S E N T A  
**MARIA GUADALUPE NESTORA RODRIGUEZ GIL**

SAN JUAN DE ARAGON, EDO. DE MEX.

1995



Universidad Nacional  
Autónoma de México

UNAM



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

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

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

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



UNIVERSIDAD NACIONAL  
AVENIDA DE  
MEXICO

ESCUELA NACIONAL DE ESTUDIOS PROFESIONALES

ARAGÓN

DIRECCION

MA. GUADALUPE RODRIGUEZ GIL  
P R E S E N T E .

En referencia a su atento escrito de fecha 27 de febrero del año en curso por el que solicita cambio de asesor en su trabajo de Tesis intitulado "SISTEMAS ABIERTOS SUS PROMESAS Y REALIDADES", mismo que propone sea dirigido por la profesora, Ing. SILVIA VEGA MUYTOY con fundamento en el punto 6 y siguientes, del Reglamento para Exámenes Profesionales en esta Escuela, y toda vez que la documentación presentada por usted reúne los requisitos que establece el precitado Reglamento; me permito comunicarle que ha sido aprobada su solicitud.

Aprovecho la ocasión para reiterarle mi distinguida consideración.

ATENTAMENTE

"POR MI RAZA HABLARA EL ESPIRITU"

San Juan de Aragón, Mex., marzo 10. de 1995

EL DIRECTOR

M. en C. CLAUDIO C. MERRIFIELD CASTRO

c c p Unidad Académica.  
c c p Jefe de Carrera Ingeniería en Computación.  
c c p Asesor de Tesis.

CCMC' AIR' 11a

*Esta página es especial, está reservada a las dedicatorias.*

*Todas las personas que yo conozca tienen algo especial para mí, no cabrían todos sus nombres en esta solitaria página, tampoco debería yo ocupar más páginas, pues corre el riesgo de que en lugar de hablar de síntomas abiertos, sea sea un libro de dedicatorias. Tal vez sea más fácil decir: Lo dedico a mis parientes, o tal vez lo dedico a mis amigos, pero ellos merecen mucho más que estas breves palabras, así que dejare escrito:*

*A mis amigos, compañeros de trabajo y de la UTAEM con mucho cariño.*

*Con profundo agradecimiento a la Escuela Nacional de Educación Profesional Aragón y a mis profesores porque me mostraron el camino al maravilloso mundo de los bits.*

*A Candy que es mi madre, que con su cariño y apoyo el camino fue menos difícil.*

*A esas 11 personas que han sido mi compañía en esta vida y que junto con mi madre formamos esta maravillosa familia, mis hermanos: Abraham, Flor, May, Rafa, Adán, Eva, Paly, César, Barbly, Luz y Juan.*

*A ese ser tan especial que transformé literalmente mi vida:*

*David, con todo el amor que tu madre siente por ti.*

*A Dios, el creador de todas las cosas y sobre todo, del amor....*

**"SISTEMAS ABIERTOS:**

**SUS PROMESAS Y REALIDADES"**

# **C O N T E N I D O**

**PROLOGO.**

**OBJETIVOS Y ALCANCES.**

**INTRODUCCION.**

## **CAPITULO 1.**

### **EL DEPARTAMENTO DE SISTEMAS DE INFORMACION.**

- 1.1 Evolución de la Administración de la tecnología de la información.
- 1.2 Evolución hacia la integración administrativa.
- 1.3 Ambiente tecnológico.
  - 1.3.1 Tendencias del hardware.
  - 1.3.2 Tendencias del software.
  - 1.3.3 Tendencia en los datos.
  - 1.3.4 Tendencia en las comunicaciones.
- 1.4 El ambiente de negocios.
  - 1.4.1 Cambios en el mercado.
  - 1.4.2 Cambios en el ambiente laboral.
- 1.5 Evolución del Departamento de Sistemas de Información en la estructura organizacional.
- 1.6 Evolución de los Sistemas Abiertos
- 1.7 Problemas típicos.
- 1.8 Conclusiones.

## **CAPITULO 2.**

### **SISTEMAS ABIERTOS.**

- 2.1 Beneficios de los Sistemas Abiertos.**
- 2.2 Definición y conceptos de Sistemas Abiertos.**
  - 2.2.1 Portabilidad-Interoperabilidad-Integración.**
  - 2.2.2 Estándares.**
  - 2.2.3 Organismos Internacionales que trabajan los estándares.**
  - 2.2.4 Abierto.**
  - 2.2.5 Disponibilidad y accesibilidad.**
  - 2.2.6 Arquitectura de sistemas.**
  - 2.2.7 Máquina abstracta.**
- 2.3 Modelo MUSIC.**
  - 2.3.1 M Management (Administración).**
  - 2.3.2 U User Interface (Interface de usuario).**
  - 2.3.3 S Service Interfaces for Programs (Interfaces de servicio para programas).**
  - 2.3.4 I Information and Data Formats (Información y formatos de datos).**
  - 2.3.5 C Communications Interfaces (Interfaces de comunicación).**
- 2.4 Seguridad**
- 2.5 Internacionalización**
- 2.6 Ambiente de Desarrollos de Software**
- 2.7 Perfil de ambiente de aplicación (AEP.- Application Environment Profile).**
- 2.8 Desarrollo de perfiles en organizaciones de estándares.**
- 2.9 Ambientes de sistemas abiertos.**
- 2.10 Conclusiones.**

## **CAPITULO 3.**

### **TECNOLOGIA ACTUAL.**

- 3.1 Perfiles de ambientes de aplicación.**
- 3.2 Desarrollo de perfiles en organizaciones de estándares.**
- 3.3 Ambientes de sistemas abiertos.**
- 3.4 Tecnología actual-**

## **CAPITULO 4.**

### **ANALISIS DE LA SITUACION Y PLANTEAMIENTO DE ALTERNATIVAS.**

- 4.1 Portabilidad.**
  - 4.1.1 Datos-Aplicaciones.**
  - 4.1.2 Personal.**
- 4.2 Interoperabilidad.**
- 4.3 Los estándares.**
- 4.4 Los mainframes.**
- 4.5 Computadoras Personales.**
- 4.6 Proveedores.**
- 4.7 Estructura organizacional para los sistemas abiertos.**
- 4.8 Conclusión.**

## **CAPITULO 5.**

### **METODOLOGIA APLICADA A UN EJEMPLO PRACTICO.**

- 5.1 Metodología.**
- 5.2 Factor Humano.**
- 5.3 Ejemplo.**



- 5.3.1 Objetivo.
- 5.3.2 Análisis del entorno.
- 5.3.3 Identificar los requerimientos de las aplicaciones.
- 5.3.4 Preparar un perfil.
- 5.3.5 Adquisición, desarrollo o migración de aplicaciones.
- 5.3.6 Verificar las características de sistemas abiertos de la aplicación.
- 5.3.7 Verificar los objetivos.
- 5.3.8 Repetir la secuencia.

5.4 Conclusión del ejercicio.

**CAPITULO 6.**

**CONCLUSION.**

**APENDICE A** Índice de figuras.

**APENDICE B** Glosario

**APENDICE C** Bibliografía.

## PROLOGO

Debido al alto costo que significa operar con mainframe, algunas compañías que cuentan con esta tecnología están considerando que el centralizar la información no es lo más acertado y hay casos en que es ineficiente, de tal suerte que están buscando otras opciones para mejorar sus servicios deseando por supuesto que ello no represente fuerte carga presupuestal.

La proliferación de computadoras personales de diferentes marcas en todas las organizaciones, que en un principio representaban islas de información y que por algún tiempo tuvieron limitaciones en los servicios que prestaban, el día de hoy ofrecen recursos casi comparables con los mainframes y también hoy se pueden conectar entre sí.

Adicionalmente a los mainframes y las computadoras personales, también existen las minis, así que hay empresas que tienen que operar con diferentes plataformas de cómputo y parece paradójico que con tanto equipo no sean lo más eficientes y productivas que quisieran ser.

Actualmente la evolución de la industria de la tecnología de la información es muy acelerada, hoy en el mercado se manejan nuevos conceptos, arquitecturas, estrategias, etc., que prometen resolver diversas requerimientos de diferente manera, pero hoy es innegable que quien se perfila como más prometedor es el concepto llamado "SISTEMAS ABIERTOS", pues todo parece indicar que será una de las estrategias de la década.

He aquí que surge una pregunta importante:

¿Qué son los "SISTEMAS ABIERTOS"?

Los Gerentes de Sistemas que deben decidir o apoyar las decisiones acerca de la estrategia a seguir en cuanto a soluciones de información, deben estar lo suficientemente documentados para aportar información veraz y concisa acerca de las diferentes opciones tecnológicas y ésto significa tener una idea clara de cómo las diferentes opciones pueden proporcionar soluciones acertadas para las características específicas de cada compañía.

Para resolver muchos problemas y satisfacer requerimientos nuevos, los "SISTEMAS ABIERTOS" han establecido ciertas promesas, las cuales en determinado momento tendemos a sobrestimar o confundir, es por ello que en las organizaciones que están considerando esta opción como solución a sus problemas, deben tener claro en qué consiste cada una de ellas y de qué manera coadyuvarán a resolver sus requerimientos.

Cualquiera que sea el caso es importante considerar que los costos que implican el adquirir, cambiar o adaptar nueva tecnología no son mínimos, sin olvidar la inversión que deberá hacerse en capacitación a personal, periodos de migración, afinación, ajustes y otros agregados, por eso es importante que decisiones de este tipo estén fuertemente respaldadas con la investigación y el análisis correspondiente.

## **OBJETIVOS**

**Analizar las promesas establecidas por los Sistemas Abiertos y establecer su realidad actual.**

**Plantear una metodología a seguir para adoptar sistemas abiertos en las organizaciones.**

## **ALCANCES**

**El alcance de esta tesis abarcará organizaciones consideradas medianas a grandes que cuenten al menos con un mainframe o varias plataformas de cómputo.**

## INTRODUCCION

Los capítulos están organizados de la siguiente manera:

**CAPITULO 1.- EL DEPARTAMENTO DE SISTEMAS DE INFORMACION.** Analiza el Departamento de Sistemas de Información refiriéndose a compañías medianas y grandes que cuentan con al menos un mainframe y otras plataformas. El análisis se refiere a su estructura organizacional y funcional actual, a la forma como vienen dando el servicio, a los problemas típicos a los que se enfrentan y a las motivaciones principales a las que obedece el deseo de cambio, todo ello referido a la administración de la tecnología de la información.

**CAPITULO 2.- SISTEMAS ABIERTOS.** En este capítulo se habla de lo que son actualmente los sistemas abiertos. Trata de los conceptos, definiciones, arquitecturas, Instituciones que los respaldan y de las promesas que establecen.

**CAPITULO 3.- TECNOLOGIA ACTUAL.** Menciona a la tecnología actual que dice pertenecer a los sistemas abiertos, comprende tanto hardware como software.

**CAPITULO 4.- ANALISIS DE LA SITUACION Y PLANTEAMIENTO DE ALTERNATIVAS.** Hace un análisis de las promesas de los sistemas abiertos, de la factibilidad de implantarlos en diferentes organizaciones, las ventajas, desventajas, repercusiones y la realidad para las organizaciones.

Se establecen conclusiones de acuerdo a los resultados obtenidos.

**CAPITULO 5.- METODOLOGIA APLICADA A UN EJEMPLO PRACTICO.** Se establece un ejemplo utilizando uno de los modelos de solución sobre la problemática planteada para una Institución.

**CAPITULO 6.- CONCLUSIONES.**

# **CAPITULO 1**

## **EL DEPARTAMENTO DE SISTEMAS DE INFORMACION**

## **CAPITULO 1**

### ***EL DEPARTAMENTO DE SISTEMAS DE INFORMACION***

En este capítulo se hace una revisión de como ha venido desarrollándose la administración de la tecnología, de quiénes han sido los principales protagonistas en ella, cuáles han sido y son los factores que participan en su evolución y como ha venido ésta cambiando su posición dentro de las empresas.

Así mismo y, a manera de introducción hacia el capítulo 2, veremos como ha sido la evolución de los sistemas abiertos dentro del marco de necesidades de los usuarios.

## **1.1.- EVOLUCION DE LA ADMINISTRACION DE LA TECNOLOGIA DE LA INFORMACION**

En el desarrollo de esta tesis se hará referencia a datos estadísticos de los Estados Unidos debido principalmente a dos razones, la primera a que la literatura más actual y accesible es de ese país, y la segunda que es el país en donde la tecnología de la información es fabricada y utilizada fuertemente desde que surgió como apoyo al manejo de la información.

La información es la principal protagonista de la tecnología que a finales de los años 50's empezó a aparecer dentro de las empresas. Los empleados que manejaban información empezaron a aumentar en número considerable. Fue en Estados Unidos donde en 1957 se pasó de la era industrial a la era de la información. En aquellos años el número de empleados cuyo trabajo tenía que ver con manejo de información aumentó considerablemente.

Después, durante los años 70's, los empleados de las oficinas superaban el 50% de la fuerza de trabajo total de los Estados Unidos. También se empezó a notar una baja en el trabajo del campo (La fuerza de trabajo se centralizó en las grandes ciudades).

A finales de los años 50's y principio de los 60's, la tecnología de la información empezó a tomar fuerza.

De forma paralela y muy independiente, la telefonía ya era bastante popular, aunque sus costos se consideraban todavía altos. Las computadoras por su parte, ya comenzaban a ser usadas en el proceso de aplicaciones para proceso de información. Su primer papel fue reemplazar a las viejas máquinas eléctricas de registro contable. No obstante su participación incipiente en apoyo al manejo de la información, en aquellos días su impacto era demasiado modesto por un lado, y por otro, se consideraban caras para el servicio que daban.

Las labores cotidianas de oficina todavía se realizaban sin mucho apoyo tecnológico. Las copiadoras empezaban a aparecer en escena y como cuando empieza a aparecer en el mercado un equipo nuevo, sus costos no eran muy bajos. Para aquel entonces las máquinas de escribir eléctricas ya tenían un lugar bien definido en el mercado y en el campo laboral administrativo, fueron éstas las precursoras de los hoy tan populares y útiles procesadores de palabras.



El facsímil que servía para reproducir documentos a distancia como lo hace hoy el imprescindible fax, solo se empezó a generalizar hasta después de los años 70's.

Entre otras formas del manejo de la información también podemos mencionar al correo, el telegrafo, el télex, el teléfono y por qué no a los propios mensajeros. Hay tantas y tantas formas con las que se ha manejado la información dentro de las empresas, que es importante no olvidar que existieron (y existen todavía en algunas empresas) por el solo hecho de que el día de hoy cómodamente contamos con tecnología que nos hacen más amigable su manejo.

Pues bien, la necesidad de trabajar con información creció rápidamente después de los años 60's, impactando considerablemente en la organización de las empresas.

Conforme la tecnología se fué introduciendo en las empresas para apoyar el manejo de la información, se hizo necesaria una forma más eficiente para su administración. Dicha introducción se hizo de forma muy independiente en diferentes puntos de la estructura organizacional empresarial y se pueden identificar plenamente 3 especialidades:

**COMPUTACION ADMINISTRATIVA.**- Se refiere solamente al proceso de datos; en un principio las computadoras se utilizaban solamente para procesos contables, ya que como se menciona antes, reemplazaron a las viejas máquinas eléctricas de registro unitario.

**TELECOMUNICACIONES.**- Este servicio lo daban tradicionalmente empresas ajenas a la compañía y la razón era la alta especialización requerida para su manejo, con ésto se liberaba a la compañía de las molestias de su administración. Esta situación explica el porqué en un principio no era grande el número de personas encargadas de este servicio. Este servicio en un principio era supervisado por el área encargada del mantenimiento del edificio, puesto que tenía que ver con cableado y su conexión, con ductos, etc.

Como consecuencia de lo anterior, en aquel entonces se dió poca importancia a las comunicaciones, debido a que la mayoría de las compañías erróneamente igualaban la importancia de varias herramientas para el apoyo en el manejo de la información con el número de gente

requerida para su operación, subestimando el impacto o el costo que tuvieran para la empresa.

**PRODUCTOS ESPECIALIZADOS DE OFICINA.**- Correo y mensajería, centros de reproducción y grupos de mecanógrafas, generalmente fueron utilizados independientemente por todas las áreas a las que daban servicio y su control y administración era precisamente en el área a la que apoyaban.

Las diferentes tecnologías que hoy consideramos dentro del marco informático, evolucionaron separadamente desde el punto de vista de su administración y control, a continuación se presenta un cuadro que muestra la infraestructura clásica de esa época:

PRODUCTO/SERVICIO	AUTORIDAD	PROVEEDOR	USUARIO
Proceso de datos	Director de Proceso de datos	IBM, DEC, Honeywell Bull	Contabilidad, Nómina, Reservasiones, Proceso de Cheques
Telecomunicaciones, Télex, Teléfono	Administrador de Telecomunicación	AT&T, Gerente	Depto de Télex, Oficinas
Productos especializados de Oficina, Equipo de Correo, Duplicadores, Microfilm, Proceso centralizado de palabra	Vice-Presidente Administrativo	Fieden Bell & Howell, Kodak 3M	Correo, reproducción, Centro de proceso de palabra
Productos generales de Oficina, máquinas de escribir, copadoras, procesadores convencionales de palabra	Vice-Presidente Administrativo	Xerox, IBM	Savin, Varias oficinas

**FIGURA NUM. 1.- INFRAESTRUCTURA CLASICA.**

## **1.2.- EVOLUCION HACIA LA INTEGRACION ADMINISTRATIVA**

La infraestructura organizacional que de manera independiente administró a la tecnología de apoyo a las áreas administrativas que surgiera en los 60's, duró y funcionó bien hasta los 70's, a partir de los 80's se empezó a sentir presión para proceder a su integración administrativa.

Una de las razones era el alto costo que significaban cada una de las tecnologías.

La segunda razón fué que las barreras tecnológicas entre cada una de las herramientas se fueron disolviendo. La mayoría de los procesadores de palabra y otros productos para oficina eran realmente computadoras de propósito general que podían manejar muchas clases de software. En octubre de 1981, Office Products News pudo contar más de 140 productos que combinaban proceso de palabras y proceso de datos.

En los 80's numerosos equipos de oficina que previamente no contenían elementos electrónicos ahora su estructura era computarizada. Las modernas fotocopiadoras ahora no solo sirven para copiar, sino que también se usan como impresoras y terminales de facsimil. Actualmente es difícil encontrar un equipo de oficina que no contenga un microprocesador y no pueda ser programado para manejar múltiples tareas, con esto se provocó la ruptura de viejas fronteras de autoridad a través de toda la estructura organizacional de las empresas.

Las redes rápidamente están eliminando las barreras restantes alrededor de los tradicionales feudos de información. Los productos de oficina actualmente están siendo integrados en sistemas de oficina que aprovechan alguna forma de conectividad entre redes de área local. El proceso de datos centralizado está derivando en proceso de datos distribuidos.

Estas dos razones fueron justificación suficiente para desear administrar de manera integral los servicios que proporcionaban las tecnologías de apoyo a la información.

### **1.3.- AMBIENTE TECNOLÓGICO**

#### **1.3.1.- TENDENCIAS DEL HARDWARE**

A principios de los 60's la preocupación principal en cuanto al hardware entre los administradores del proceso de datos, fue obtener eficiencia en las máquinas y estimular el desarrollo en dicha tecnología. El proceso por lotes (batch) fue predominante, los sistemas en línea surgieron a finales de los años 60's.

Cuando las máquinas pequeñas salieron al mercado, fue posible distribuir el proceso de datos. La perspectiva del proceso distribuido fue el tema a principios de los 70's, aún cuando muy pocas compañías tenían algún sistema distribuido en ese momento.

Fue hasta mediados de los años 70's cuando el poder de proceso comenzó a ser movido, aunque lentamente, fuera del site principal. Este proceso en aquellos años lento, fue provocado por la insistencia de los usuarios, quienes comenzaron a comprar su propias minicomputadoras departamentales y procesadores de palabra.

En los años 80's, principalmente debido al advenimiento de las computadoras personales, esta tendencia se aceleró más allá de las expectativas de la mayoría de la gente y de los propios administradores de los sistemas de información.

La tendencia de movimiento del hardware y del poder del proceso de datos fuera del control del departamento de sistemas de información se estableció desde entonces y se está acelerando cada vez más.

#### **1.3.2.- TENDENCIAS DEL SOFTWARE**

La necesidad de enfoques sistemáticos para el desarrollo y mantenimiento de productos de software se patentizó en la década de los 60's. Durante ésta aparecieron las computadoras de la tercera generación y se desarrollaron técnicas de programación como multiprogramación y el tiempo compartido. Estas nuevas capacidades aportaron la tecnología necesaria para el establecimiento de sistemas computacionales interactivos, de multiusuario, y en línea, gracias a ésto surgieron nuevas aplicaciones para la computación,

como las reservaciones aéreas, bancos de información médica, control y dirección de equipos militares, entre otros.

En los años 70's la preocupación en la programación se centró primero en la programación modular y estructurada. Después se expandió a las metodologías para el desarrollo del ciclo de vida de los sistemas de información aunque desde 1988 se empezó a hablar de la Ingeniería de Software. Uno de los objetivos del desarrollo de metodologías fué el introducir más rigor en las técnicas de administración de los proyectos. Otra meta fué conseguir que los usuarios se involucraran más en el desarrollo, al menos por lo que corresponde a las primeras etapas del proyecto.

Durante cerca de 25 años, la atención se centró en el desarrollo de sistemas grandes, casi no se le prestaba atención a sistemas pequeños, debido en gran parte a que los mainframes eran la tecnología que se consideraba respondía a las necesidades de aquel entonces y porque la tendencia a centralizar estaba en todo su apogeo, no solo en el ambiente informático, sino en toda la estructura organizacional y aún social

A partir de la década de los 70's se empezó a considerar seriamente la programación por parte de empleados ajenos al departamento de sistemas, ya que tradicionalmente los usuarios solamente se involucraban en la etapa de definición de requerimientos, la programación por parte de usuarios finales aún no se consideraba seriamente.

Fué aproximadamente a mediados de la década de los 70's que aparecen otras 2 tendencias: una fué la maduración de la industria del software, así que la compra de software apareció como otra alternativa viable para los requerimientos de sistemas. La segunda fué que se comenzó a poner atención sobre aplicaciones para soporte a las decisiones (DSS.- Decision Support Systems) y de soporte a la administración, pero solo fué a finales de los años 70's que este tipo de aplicaciones empezaron a tomarse en serio. A partir de ese momento el interés sobre este tipo de aplicaciones siguió incrementándose.

Por lo que respecta a las computadoras personales, la tendencia es que como el hardware de esta plataforma ha venido optimizándose fuertemente el software respectivo da muchas posibilidades a los usuarios para hacerse cargo de sus propias necesidades de sistemas de información, con mayor prontitud y al gusto de los mismos.

En conclusión, por lo que respecta al software, las tendencias incluyen más disciplina y rigor para la programación doméstica (la realizada por

programadores de las compañías), alternativas para comprar software en lugar de hacerlo en casa, involucrar a los usuarios finales en el desarrollo de proyectos y finalmente que los usuarios están independizándose y haciendo sus propias aplicaciones.

### **1.3.3.- TENDENCIA EN LOS DATOS**

A principio de los años 60's la preocupación eran las técnicas de administración y organización de los archivos para aquellos que formaban parte de aplicaciones individuales y separadas, después a finales de esa misma década, surgieron los sistemas para administrar archivos que manejaban grandes volúmenes de datos, a partir de este enfoque más generalizado, surgió el concepto de base de datos corporativas que servían a varias aplicaciones, teniendo como consecuencia que surgiera la función de la administración para los sistemas manejadores de bases de datos.

En los 70's el interés de los datos se enfocó a solucionar técnicas para la administración de las bases de datos, como fueron evolucionando estas arquitecturas, se hizo patente que el elemento clave de estos productos fue su diccionario de datos. Las primeras funciones de estos diccionarios fueron solamente de especificación y formato, pero después se diversificaron. Los diccionarios están ahora siendo usados para almacenar en ellos información acerca de la relación entre sistemas, fuentes y usos de los datos, requerimientos de ciclo de tiempo, etc. En 1982 el tema de datos distribuidos fue finalmente puesto a discusión por los expertos con más profundidad.

Como en el software, la mayor parte de los pasados 25 años, la preocupación acerca de los datos se enfocó principalmente en técnicas para manejarlos mejor en ambientes centralizados. Hasta hace pocos años, ha habido el interés en permitir que otros departamentos en las empresas tengan acceso directo a los datos. El día de hoy este tema está tomando mucha fuerza. La administración de los sistemas de información está encarando nuevos requerimientos acerca de proporcionar acceso a recursos importantes y también a distribuirlos. Los usuarios son los que están presionando con estos requerimientos.

#### **1.3.4.- TENDENCIA EN LAS COMUNICACIONES**

Los primeros usos de la tecnología de las comunicaciones para la transmisión de datos a mediados de los 60's se centraron en los sistemas en línea y de tiempo compartido, al entrar a la década de los 70's, el interés en redes tanto públicas como privadas fue notorio. La tecnología de las comunicaciones sufren una gran evolución que paralelamente a la experimentada por la industria de cómputo, abren nuevas posibilidades para usos diferentes de los sistemas de información. El día de hoy la mayoría de la gente considera a las comunicaciones y a las computadoras como elementos inseparables dentro del concepto de tecnología de la información.

Desde 1983 ha habido un gran interés por los sistemas de redes, debido a que ellos proporcionan ventajas estratégicas a las compañías. Así como hoy en día (1994) se habla de alianzas estratégicas internacionales, tratados de libre comercio, el desarrollo en redes también ha incrementado el interés en la computación global.

## **1.4.- EL AMBIENTE DE NEGOCIOS**

La tecnología evoluciona y de forma paralela, el ambiente de negocios también lo hace. Hay cambios en el ambiente económico que están causando que los administradores analicen la situación actual de sus negocios e investiguen formas diferentes para ser más competitivos y exitosos o tan solo para no desaparecer del mercado. Cuando hay cambios en el mercado, también podemos hablar de que hay factores que afecta el actual mercado laboral, respecto a estos dos elementos (cambios en el mercado y repercusiones en el ambiente laboral) hablaremos a continuación:

### **1.4.1.- CAMBIOS EN EL MERCADO**

Los cambios en los negocios son mundiales, es muy sonado en el ambiente informático los pocos éxitos que tuvo IBM en los últimos años y que la llevaron a reestructurar a la organización para ver si de esa manera responde a los nuevos retos del mercado. Como IBM, hay muchos casos y ésto se dejó sentir a partir de la mitad de la década de los 80's. La tecnología de la información ha contribuido fuertemente a tales cambios, ya que permite que la información fluya de manera rápida, provocando con ello que los eventos tomen lugar a diferente velocidad y que el paso al cual deben responder las organizaciones y los individuos sea diferente.

Los principales cambios en el mercado son los siguientes:

#### **DES-REGULACION EN LAS INDUSTRIAS**

Aquí en México se está notando más libertad en la forma en que los industriales pueden hacer negocios, la apertura de las fronteras y la eliminación de cargas arancelarias están dando más flexibilidad a las industrias (al menos eso anuncia el gobierno).

#### **CRUCE DE FRONTERAS INDUSTRIALES**

Este fenómeno consiste en que productos o servicios propios de un campo, están invadiendo otro, como ejemplo podemos mencionar a los grandes almacenes, que antes solo se dedicaban a vender artículos y ahora están entrando al campo de los servicios bancarios, ejemplo: tarjeta de crédito, o



también anuncian en los supermercados que el "banco está en el super". Otro ejemplo palpable es el de compra en casa vía televisión que ya compite con el de compra vía correo por medio de catálogo. Las compañías aseguradoras con las pólizas de vida incluyen servicios de inversión con ciertas características, invadiendo también los servicios bancarios.

### **INCREMENTO EN LA COMPETENCIA EXTRANJERA**

En el caso específico de México, todos estamos enterados de que con el tratado de libre comercio entre Estados Unidos, Canadá y nuestro país, se establecerá oficialmente la apertura al libre comercio. Con esta apertura, y aún con las muchas cláusulas que han desatado polémicas acerca de repercusiones ambientales, laborales y económicas para cada uno de los países involucrados, es bien claro que habrá fuerte competencia extranjera para la industria local.

En los Estados Unidos ya desde hace varios años, los bancos y otros negocios, se han enfrentado a bancos de extranjeros en ese mismo país con quienes tienen que competir. Es bien marcada la competencia en diferentes artículos de origen asiático que invaden nuestro país y muchos otros, tal es el caso de artículos electrónicos, ropa, calzado, los países de origen son Corea, Taiwan, Hong Kong, Singapur, solo por mencionar algunos.

Así que el incremento en la competencia multi-nacionalista es y será fuerte en todos los países que quieran participar en los cambios económicos internacionales y todos sabemos que México no es la excepción.

### **ENSAMBLADORAS**

Ya se ha hecho común que diferentes partes de artículos se fabriquen en un país y sean enviadas a otro para su ensamble, logrando con ésto dividir el costo de la mano de obra total.

#### **1.4.2.- CAMBIOS EN EL AMBIENTE LABORAL**

El ambiente de trabajo tradicional también está cambiando, el arte de manejar a la gente está sufriendo fuertes cambios. He aquí algunos de los cambios en el ambiente de trabajo tradicional que tendrán un impacto importante en la forma como trabaja la gente.

## LA JERARQUIA

La mayoría de nosotros nos incorporamos a la estructura organizacional de la empresa en donde laboramos cuando llegamos a una compañía y bien pronto nos damos cuenta de el nivel que ocupamos dentro de las diferentes capas de la jerarquía. Pues ahora algunos investigadores que estudian los sistemas de oficina, creen que el estilo de trabajo y ambiente de trabajo cambiarán, porque consideran que los viejos estilos establecidos no funcionarán eficientemente en un lugar de trabajo automatizado.

El ambiente de trabajo tradicional sugieren los investigadores, está dominado por estructuras jerárquicas. Se supone que las estructuras jerárquicas funcionan de la siguiente manera:

Los empleados del nivel más inferior tienen la capacitación e información mínima suficiente para realizar solo un tipo de labor.

Hay grupos de personas que trabajan realizando el mismo tipo de labor y son vigiladas por un supervisor.

Un supervisor asigna el trabajo entre los subordinados, maneja los problemas que surgen, hace cumplir la disciplina, promueve gratificaciones, da capacitación, etc.

Los profesionales y el staff son también supervisados, pero supuestamente ellos tienen más responsabilidad en sus métodos de trabajo.

Los Gerentes y ejecutivos coordinan el trabajo de sus grupos subordinados.

Algunos investigadores no están seguros de que esta estructura organizacional continuará como la más apropiada y esto aplica para oficinas y fábricas. Como ejemplo mencionan que grupos manejados por sí mismos que realizan trabajos que van desde líneas de ensamble a entrada de datos en sistemas transaccionales, ha reaccionado más eficientemente sin estar integrados a una estructura jerárquica. Se supone que estos grupos se autoadministran, tienen bajo ausentismo, alta productividad, mejor calidad en el trabajo y están mucho más motivados que los trabajadores en grupos jerárquicos tradicionales.

El comentario anterior se refiere a personal norteamericano con diferente idiosincracia, sabemos que no todo lo que aplique en Estados Unidos, puede aplicarse de la misma manera a personal mexicano.

En una conferencia dada por IBM el 20 de mayo de 1993 en la que exponían su propio concepto de "CLIENTE-SERVIDOR", pusieron como ejemplo a la compañía VHL que había adoptado la mencionada arquitectura. Un ex-dirigente del proyecto que participó en el citado cambio tecnológico, fue presentado en dicha conferencia para que platicara su propia experiencia y, entre muchas otras cosas, mencionó que un factor estratégico para el éxito del proyecto, fue la reestructuración total de la compañía involucrando principalmente, la eliminación considerable de varias jerarquías, en su lugar habían adoptado estructuras planas, logrando con ello grandes ventajas.

En una estructura donde el control jerárquico es estricto, éste se basa en cadenas de órdenes verticales, donde desde los altos niveles que es donde se toman las decisiones hasta los inferiores de la cadena de órdenes, deberán ser observados para que una acción sea ejecutada, así el flujo de información fluye más lentamente debido a los diferentes niveles que conforman la jerarquía.

Con la actual tecnología y los retos que establece el reinante ambiente de negocios, los eventos que determinan decisiones importantes, deberán ser manejados con suficiente rapidez para que produzcan resultados positivos, por lo que se prevé que una estructura jerárquica no responderá con la rapidez que ahora se requiere.

En el caso de las empresas en las que la tecnología de la información ha llegado a influir fuertemente, y en las que los usuarios cada vez tienen más poder de cómputo, es obvio que la estructura deberá examinarse pronto.

### **EL CAPITAL HUMANO**

Hay otros investigadores que ven tres cambios importantes en la forma en que las empresas operarán. Una es en la que ellos creen que las empresas cambiarán su énfasis acerca del capital financiero hacia uno denominado el "CAPITAL HUMANO" y consiste en que los recursos humanos son la verdadera ventaja de una compañía. Se supone que existe una tendencia en la que hay preocupación por parte de las empresas en donde la salud y buena condición de su personal son importantes. Lo mismo sucederá, dicen, con la actitud hacia los clientes y los usuarios.

Cierto es que hay algunas empresas en las que el capital humano es el más importante, pero aquí lo que debemos cuestionarnos es acerca de la importancia del capital humano frente a la tecnología de la información, ¿como va a resolver la empresa actual esta situación?

### **MENOS MANDOS MEDIOS**

Esta tendencia, sumamente ligada a la mencionada anteriormente, es aquella en la que disminuirán los mandos medios. La tecnología de la información le permite a las compañías reducir los mandos medios en la administración gracias a la tecnología. En adición, están apareciendo nuevas formas de trabajo en grupo que se auto-administran, como también se menciona anteriormente, muchos de estos grupos están adquiriendo silenciosamente autonomía.

Esta tendencia tendrá consecuencias graves para las personas nacidas entre 1945 y 1956, quienes están alcanzando la edad donde deberán ser promovidos a mandos medios, pero debido a los efectos provocados por la tecnología de la información, no habrá suficientes puestos para ellos. Se pueden perder muchos talentos a menos que las empresas inventen estructuras de trabajo alternativas.

### **DESEMPLEO**

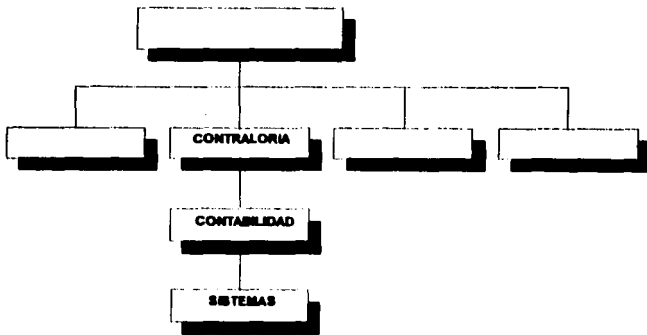
El desempleo es un fenómeno que afecta a todos los países, y no es la excepción el nuestro. Hay especulaciones respecto a los efectos que provocará el tratado de libre comercio en los 3 países, y se supone que la década de los 90 será la más crítica.

Los factores anteriormente mencionados nos hacen reflexionar acerca del papel que deberá desempeñar el departamento de sistemas de información para apoyar firmemente a los negocios.

### 1.5.- EVOLUCION DEL DEPARTAMENTO DE SISTEMAS DE INFORMACION EN LA ESTRUCTURA ORGANIZACIONAL

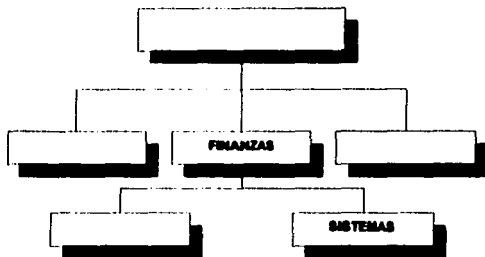
Hemos revisado un poco de la historia de la que ha sido protagonista la información, como surgió la tecnología para el apoyo en su manejo, como evolucionó y cuales son las tendencias. Al hacer dicha revisión nos dimos cuenta que diferentes áreas se encargaron de administrar la tecnología de la información, así que resulta interesante revisar un poco como fue cambiando de posición el departamento de sistemas de información dentro de la estructura organizacional de las empresas.

La posición que tuvo el departamento de sistemas en la década de los 60's fué en el área contable pues se dieron cuenta que la computadora era una herramienta que le ahorra mucho trabajo:



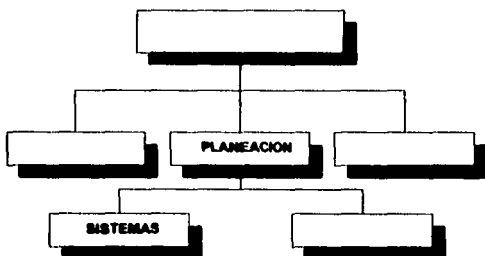
**FIGURA NUM. 2.- POSICION DEL DPTO. DE S.I. EN LOS 60'S**

En la década de los 70 otras áreas de la organización se dieron cuenta del poder que representaban las computadoras, así que en algunos casos el área de Finanzas tuvo bajo su cargo el área de Sistemas, dedicándola aparte de contabilidad a nómina, control de presupuesto, etc.:



**FIGURA NUM. 3.- POSICION DEL DEPTO.DE S.I. EN LOS 70'S**

A partir de los años 80's se hizo patente el poder que tenía la información, así que el departamento de sistemas surgió por el área de Planeación de las organizaciones, dedicándola a tareas adicionales:



**FIGURA NUM. 4.- POSICION DEL DPTO. DE S.I. EN LOS 80'S**

En nuestra década de los 90's, ya es demasiado clara la importancia que significa el área informática para las organizaciones, pero también se hicieron patentes los problemas de autoridad y responsabilidad, comunicación y organización en las diferentes áreas que conforman la estructura organizacional de la empresa, todo esto originado gracias al poder que significa el control de la información. Por tal motivo, se va integrando el área informática hasta tomar la forma que hoy en día tiene y que recibe diferentes nombres e incluso, dependiendo la importancia de los servicios que proporciona y el reconocimiento que se le da a los mismos, ocupará su propio lugar dentro del primer nivel de la estructura organizacional y es considerada en algunos casos como una Dirección:



**FIGURA NUM. 5.- POSICION DEL DPTO. DE S.I. EN LOS 90'S**

## 1.6 EVOLUCION DE LOS SISTEMAS ABIERTOS

La primera necesidad que los usuarios manifestaron acerca de los sistemas abiertos fue la de la PORTABILIDAD de aplicaciones entre diferentes plataformas y uno de los primeros antecedentes que muestra rasgos de portabilidad de sistemas sin lugar a dudas fue IBM, pues tenemos un ejemplo de ello con sus arquitectura de sistemas 360, todos con el mismo juego de instrucciones y capaces de correr el mismo sistema operativo, fué un buen ejemplo de portabilidad dado que el código desarrollado para un sistema en la familia podía correr en otra, adicionalmente IBM licenció su sistema operativo a usuarios quienes podían elegir otro proveedor de hardware de vendedores que fabricaron clones de esos sistemas.

Otro ejemplo de portabilidad lo dan los lenguajes de programación, por ejemplo el FORTRAN y COBOL, que proporcionan no solo elementos de portabilidad para aplicaciones (aunque con limitaciones) sino también de los programadores mismos, ya que éstos pueden moverse de plataforma necesitando un mínimo de capacitación para poder desarrollarse en otra. La madurez de los lenguajes ha sido gracias al trabajo de organizaciones internacionales dedicadas a los estándares, logrando una independencia en cuanto a vendedores del producto.

El desarrollo de este nivel de portabilidad proporcionó el primer ejemplo de las verdaderas capacidades de los sistemas abiertos.

La portabilidad sin suficiente funcionalidad generalmente no es útil, así como suficiente funcionalidad sin portabilidad es útil pero restringe la evolución tecnológica y las opciones de competitividad. La situación óptima para cada usuario es **SUFICIENTE FUNCIONALIDAD Y SUFICIENTE PORTABILIDAD.**

La familia de sistemas 360 de IBM, así como COBOL y FORTRAN, son solo un ejemplo de lo que representa la portabilidad a este nivel y pueden representar la solución para algunos usuarios, pero evidentemente existieron otros que requerían soluciones que iban más allá de las aquí planteadas.

La siguiente fase en la evolución de los sistemas abiertos fué la aparición de los sistemas en red. Un buen ejemplo es INTERNET (TCP/IP). Desde finales de los 60's la Advanced Research Projects Agency (ARPA hoy DARPA) de los Estados Unidos, ha realizado investigaciones encaminadas a crear redes confiables para ligar sistemas de cómputo heterogéneos a través de todo el mundo, entre los resultados de su trabajo se pueden mencionar ARPANET



que es una red internacional, TCP (Transmision Control Protocol), IP(Internet Protocol), FTP (File Transfer Protocol), entre otros.

Hay dos modelos de comunicaciones que surgieron y que han sobrevivido: el modelo OSI (Open Systems Interconnection) de la ISO y el SNA (Systems Network Architecture) de IBM; el primero claramente de enfoque abierto y el último con menor grado de apertura aún cuando es propietario.

Conforme fueron proliferando las redes, los usuarios comenzaron a considerar que la conectividad e integración de los equipos eran un atributo necesario de los sistemas abiertos.

Formando parte de esta evolución no se puede dejar de mencionar al pequeño monstruo llamado computadora personal. El sistema operativo MS-DOS proporcionó el mejor de los ambientes estándar para los usuarios de computadoras personales, gracias a su bajo costo y su amplia distribución, crearon un mercado masivo de sistemas y aplicaciones.

Hasta 1991 se creía que las microcomputadoras estaban limitadas por la arquitectura 80x86 de Intel con un direccionamiento de 16 bits, pero hoy aunque no han salido al mercado, existen procesadores poderosos que ya no la limitan; más aún, las computadoras personales de hoy ya no están limitadas a trabajar una sola tarea. Una limitante que sí tienen hoy en día (1994), es su fragilidad en cuanto al riesgo de propagación de virus entre estas herramientas de cómputo, su sistema operativo cuenta con poca o nula protección a nivel hardware o software. (JUSTO ES DECIR QUE YA ESTAN TRABAJANDO PARA SUPERAR ESA LIMITANTE).

Pero no obstante que hoy todavía encontremos algunas limitaciones en las computadoras personales, no se puede negar que es un gran precedente de lo que es un sistema abierto.

El sistema operativo UNIX es otro ejemplo de ambiente de sistemas abiertos. Históricamente, el sistema operativo UNIX ha sido una opción viable para crear una base común. Satisface diversos requerimientos de un sistema abierto y, con el software de aplicaciones apropiado, teóricamente se pueden producir sistemas altamente portables tanto entre diferentes sistemas UNIX como a otros sistemas que satisfacen los estándares de interfaces tales como aquellos desarrollados por X/OPEN y POSIX.

Una de las razones por las que UNIX es considerado un buen candidato para usarse en sistemas abiertos, es que la mayoría de éste está escrito en lenguaje de alto nivel, es modular y relativamente flexible. El sistema

operativo UNIX está integrado de componentes primarios que incluyen un kernel, herramientas varias y un shell. El kernel que forma el núcleo de UNIX, consiste de un grupo relativamente pequeño de programas que proporcionan los recursos del sistema interactuando directamente con el hardware de la computadora.

Las herramientas realizan acciones básicas de proceso de datos por medio de llamadas a rutinas del kernel en ciertas secuencias. Una sola herramienta para una tarea simple puede ser combinada con otras herramientas para realizar una tarea más compleja. El shell incluye la interface del usuario y opera justo como otro programa.

Aún cuando el sistema operativo UNIX es independiente del procesador, algunos servicios y código dependen directamente del hardware. Las aplicaciones implementadas en un UNIX específico tienen dependencias del hardware.

Otro aspecto atractivo de UNIX es que fué relativamente flexible en cuanto al otorgamiento de su licencia, teniendo el inconveniente que ahora hay una variedad de diferentes e incompatibles implementaciones del mismo. Varios grupos como UniForum, POSIX y X/OPEN están trabajando para identificar un núcleo común de funcionalidad que permitirán la portabilidad entre varios sistemas.

En adición a los dos sistemas operativos aquí mencionados, diversos sistemas nuevos son diseñados específicamente por demanda de ambientes distribuidos en red que se están desarrollando y probando. Uno o más de estos podrían ser candidatos apropiados para usarse en un ambiente de sistemas abiertos.

Las terminales que son el medio físico con el que el usuario interactúa con el sistema, puede ser considerado el primer antecedente de interfaces. A este respecto no todos los vendedores permiten que la interface terminal-mainframe sea abierta, pues cada uno tiene sus propios protocolos o juegos de caracteres.

El enfoque actual de los sistemas abiertos está concentrado no solo en sistemas operativos, también lo hace sobre estandarización de interfaces para ligar sistemas existentes, aplicaciones y usuarios. Se concentra en conseguir estándares aceptados internacionalmente y adicionar una variedad de componentes a un ambiente operativo modular y común.

Las interfaces que se distinguen hoy son:

Interfaces de usuario  
Interfaces de red  
Interfaces de sistema

Por lo que respecta a redes, se deben implementar estándares de la industria aceptados internacionalmente para cada componente de la red, incluyendo cada sistema operativo y aplicación. **Conforme los componentes se adhieran a los estándares, se habrán alcanzado los objetivos de los sistemas abiertos.**

Se cree que la implementación de estándares en cada sistema en una red proporcionarán una estructura común y uniforme que reducirá las dificultades de conectar sistemas heterogéneos.

Si los sistemas abiertos van a cumplir completamente sus promesas, deben proporcionar un ambiente operativo en el cual las aplicaciones puedan ser movidas fácilmente entre diferentes arquitecturas de hardware y software, así como que los usuarios puedan moverse fácilmente entre sistemas, deberán también operar en concierto con todo el software y hardware con el que ya se cuenta para así proteger la inversión hecha en los mismos.

### **1.7.- PROBLEMAS TÍPICOS**

Actualmente los problemas típicos a los que se tienen que enfrentar las empresas son:

- Los costos de la tecnología de la información utilizando grandes mainframes, son muy altos en relación con los resultados obtenidos.
- El desarrollo e implantación de un sistema en plataformas mainframe es muy tardado.
- En plataformas mainframe se cuenta con muy pocas herramientas que hagan amigable y ágil el proceso de desarrollo.
- Las cargas de trabajo en el mainframe son excesivas debido a que todos los procesos se concentran en esta plataforma (proceso centralizado).
- Se cuenta con un gran número de computadoras personales que en algunos casos:

Los mecanismos de seguridad son poco robustos dando lugar a virus o a diversas acciones nocivas premeditadas o involuntarias que afectan la integridad de la información.

Tienen la información aislada (no la comparten).

Tienen información redundante.

NOTA: Si están en red podrá compartirse la información y evitar redundancia, pero la propagación de virus y violaciones a la seguridad siguen siendo problemas latentes.

Podría continuar con la lista de problemas, pero lo importante aquí es señalar que actualmente y por diversos motivos las compañías quieren cambiar su tecnología de información para obtener mejores resultados con su uso.

Para cambiar y que los resultados del cambio den resultados positivos, no es tarea fácil y uno de los factores que contribuyen a ello en este caso, es la relativa juventud de la tecnología hoy disponible, solo por mencionar algo,

recordemos los sistemas operativos que se dicen abiertos y no lo son, los sub-sistemas que se dicen portables y no lo son.

Por otra parte, los grandes mainframes con su larga presencia en el mercado, no responden ya a los requerimientos actuales, estos grandes computadores que todavía son la tecnología con la cual muchas empresas sacan su principal producción informática, siguen teniendo permanencia aún cuando los costos de su operación son muy altos. Hay que recordar que hoy en día migrar aplicaciones de un mainframe a una plataforma de cómputo menor, que es lo que han dado en llamar "**DOWN SIZING**", todavía está en etapa de experimentación por muchas compañías y hay quienes a mitad de este proceso todavía siguen dudando acerca de si podrán o no sustituir sus mainframes.

En una nota del Computer World del 31 de mayo de 1993, señalan que la International Data Corporation (IDC) realizó una encuesta entre 500 usuarios entrevistados del continente europeo acerca de lo que significaban los mainframes para las empresas. En base a la información obtenida en la encuesta, se realizó un estudio en el Centro Europeo IDC en Dinamarca arrojando los siguientes datos:

61% de usuarios europeos piensan adoptar proceso de downsizing (planean).

39% de los encuestados aseguran que seguirán implementando sus aplicaciones en host's empresariales (195 usuarios).

6 de cada 10 usuarios de mainframe no consideran a los host's como plataformas estratégicas.

Del 61% están implementando sus nuevas aplicaciones en otras plataformas o reemplazando a sus mainframes.

Del 61% los 2/3 planean implantar nuevas aplicaciones en plataformas no-mainframe pero aseguran que mantendrán los equipos grandes dentro de la estructura de sistemas de información, lo que harán será invertir en plataformas más pequeñas en lugar de seguir invirtiendo en los mainframes instalados.

El resto: 95 usuarios que significan aproximadamente el 19% del total de 500 encuestados, piensan reemplazar sus equipos con otros más pequeños.

Las compañías medianas (menos de 1000 empleados) son las más inclinadas a sustituir los mainframes.

Los que ya no implementarán aplicaciones en mainframes, lo harán en UNIX o sustituirán los mainframes por equipos basados en UNIX.

El 21% del total de los 500 encuestados implantarán sus nuevas aplicaciones en redes (LAN'S) en la mayoría de los casos.

En conclusión señalan que tienen 3 opciones:

a).- Seguir implementando aplicaciones en mainframe, es decir, considerar al mainframe dentro de la estrategia de sistemas de información. Seguir invirtiendo en las grandes computadoras.

b).- Implementar aplicaciones en otras plataformas. Es decir, descentralizar la inversión de la tecnología de la información.

c).- Sustituir los mainframes.

## **1.8- CONCLUSIONES DEL CAPÍTULO**

Desde que hizo su aparición la tecnología de la información como apoyo en los negocios, ésta ha venido influyendo primero, en la forma como la gente hace su trabajo y después, en el funcionamiento total de las empresas. Hemos llegado a un punto en que el apoyo de esta tecnología es tan importante que incluso la estructura organizacional de las empresas está cambiando para lograr funcionar de manera más eficiente.

En el afán de ser más competitivos en el ambiente de los negocios, las compañías han estado considerando diferentes tecnologías para apoyo al manejo de la información, en la presente tesis se analizan los sistemas abiertos como una de las alternativas disponibles para lograr ésto.

La evolución se da en todos los ámbitos, en lo social, en lo político, en lo económico, pero en cuestiones informáticas que es lo que aquí se está tratando, nos damos cuenta que la evolución de la tecnología es a pasos agigantados así que es interesante hacerse las siguientes preguntas:

***¿Cual será la posición del departamento de sistemas de información después que se adopten las nuevas arquitecturas, conceptos y todas esas alternativas que hoy en día se escuchan con más fuerza?***

***¿Cual será el control y por tanto el poder que tendrá el Departamento de Sistemas de Información después de que el usuario administre sus propios recursos de cómputo?***

***Finalmente ¿Cuál será el papel que desempeñará cada uno de los recursos humanos que integran actualmente el Departamento de Sistemas de Información?***

En el siguiente capítulo veremos lo que son los sistemas abiertos, una vez que tengamos un panorama de lo que significa este concepto, volveremos a plantear estos cuestionamientos y será interesante hacer una aproximación de las posibles respuestas.

**CAPITULO 2**  
**SISTEMAS ABIERTOS**



## **CAPITULO 2**

### **SISTEMAS ABIERTOS**

Debido a la creciente demanda del mercado para contar con ambientes diseñados e implementados de acuerdo a estándares, la industria de cómputo inició su evolución hacia un ambiente que en el medio informático se ha denominado "SISTEMAS ABIERTOS".

Se dice que un sistema abierto ofrece beneficios potenciales a diferentes grupos, dependiendo del grupo que se trate, serán diferentes los beneficios; en este capítulo veremos de qué se tratan los mencionados beneficios.

En las organizaciones los administradores de la tecnología reconocen que no es sencillo obtener beneficios de los sistemas abiertos pues cada usuario tiene diferentes necesidades y se encuentra en diferentes situaciones; no habrá un manual ni un libro en el que todos encuentren todas las respuestas a sus interrogantes.

El administrador de la tecnología de la información deben entender todos los conceptos relacionados con sistemas abiertos y aplicarlos a sus negocios en una forma apropiada.

Pero para un usuario, un sistema abierto, debe responder eficientemente a sus necesidades y objetivos. El usuario no sabe de características intrínsecas de los sistemas, lo que a él le interesa es que el producto o productos que le ofrecen le proporcionen suficiente funcionalidad.

Así que para que los usuarios, que prácticamente son nuestros clientes, queden satisfechos, es necesario trabajar arduamente en un plan para el desarrollo e implementación de un ambiente de sistemas abiertos.

Y yo insisto, definir tal plan, requiere de identificar metas, prioridades, así como de examinar cada elemento del ambiente que armoniza con nuestros objetivos y sobre todo, entender bien cada uno de los conceptos relacionados con sistemas abiertos.

## **2.1 BENEFICIOS DE LOS SISTEMAS ABIERTOS**

Pero....

**¿por qué cambiar a sistemas abiertos?**

**¿qué gano con ello?**

Evidentemente los sistemas abiertos deben darnos algún beneficio, y de acuerdo a sus promesas no solo alguno, sino muchos beneficios.

Los usuarios buscan flexibilidad y justificación en los costos en que se incurre durante el ciclo de vida de los sistemas de información. Necesitan una forma de adaptarse a los cambios sin perder las inversiones hechas con anterioridad en tecnología de la información. Adicionalmente, necesitan sistemas que les permitan reducir trabajo y recursos redundantes, para producir datos solamente una vez en cualquier plataforma y tenerlos disponibles para ser usados por cualquier usuario autorizado donde y como él lo necesite.

A continuación se establecen algunos beneficios que de acuerdo a las promesas de los sistemas abiertos, serán obtenidos por diferentes grupos (usuarios, proveedores y desarrolladores):

- Portabilidad de las aplicaciones, datos y personal entre diferentes sistemas de hardware.

Esta característica le permitirá a la compañía preservar sus inversiones y tomar ventajas de tecnologías y aplicaciones nuevas y competitivas. La portabilidad de los usuarios se refiere a la reducción de costos asociados con la re-capacitación para que éste pueda manejar nuevos sistemas y aplicaciones, esto mejora la productividad.

- Interoperabilidad e integración de aplicaciones y sistemas. Permite que mezclas de sistemas diferentes operen juntos, compartiendo datos y herramientas en una forma útil y transparente.

- Independencia de un ambiente de hardware o software en particular.

- Independencia de un proveedor en particular.

- Escalabilidad que representa la flexibilidad para cambiar y expandir un ambiente de cómputo cuando sea necesario hacerlo y para elegir la mejor plataforma de aplicación para cada problema o necesidad específica de negocios.
- Capacidad de integrar aplicaciones, información y sistemas provenientes de diferentes fuentes en un ambiente productivo.
- Habilidad para soportar múltiples plataformas y generaciones de hardware provenientes de vendedores diferentes.
- Eficiencia de servicio y soporte gracias a las interfaces consistentes para diferentes plataformas de hardware.
- Estándares producidos internacionalmente que norman a la industria dando la oportunidad de certificar productos que cumplan con especificaciones de sistemas abiertos.
- Reducción en costo de desarrollo y mantenimiento.
- Incremento en la disponibilidad de software de terceras partes.

Como se puede observar, son múltiples los beneficios que se espera nos brinden los sistemas abiertos, habrá que analizar todos los elementos que intervienen en los aspectos relacionados con ellos.

## **2.2 DEFINICION Y CONCEPTOS DE SISTEMAS ABIERTOS**

Antes se mencionó que un sistema abierto es: aquel que satisface las necesidades de sistemas abiertos de un usuario específico y que no necesariamente van a ser las mismas para otro usuario, pues precisamente en eso estriba la dificultad de definir lo que es un sistema abierto.

Diferentes grupos, dependiendo su punto de vista, cada uno tiene su propia definición, ya sea que se trate de un vendedor de hardware, de software o de aplicaciones, todos dan su propia definición, es fácil observar cuando un vendedor define lo que es un sistema abierto, como está cargado de una clara tendencia mercantilista (obvio, lo que desea es vender).

Así que para fines prácticos, y porqué no si los usuarios son los que tienen la palabra: **un sistema abierto es aquel que satisface completamente las necesidades de una organización.**

Sin embargo, para propósitos del análisis que se hará en este documento, tendremos que estandarizarnos en lo que a la definición se refiere y a continuación daremos la que acordó la IEEE en su estándar POSIX 1003.0:

**"a system that implements sufficient open specifications for interfaces, services and supporting formats to enable properly engineered applications software to be ported across a wide range of systems with minimal changes, to interoperate with other applications on local and remote systems, and to interact with users in a style which facilitates user portability".**

De la anterior definición tenemos que:

**"Es un sistema que instrumenta suficientes especificaciones abiertas para interfaces y servicios, que soporta formatos que permiten que aplicaciones de software definidas adecuadamente, sean portadas a través de un amplio rango de ambientes con cambios mínimos, que interoperen con otras aplicaciones en un ambiente remoto o en ambiente local y, finalmente, que interactúan con los usuarios en un estilo que facilite la portabilidad de los mismos."**

Un elemento clave de esta definición es **"especificación abierta"** misma que a continuación se define:

**"Una especificación abierta es aquella que se hace pública y que es mantenida por medio de un proceso de consenso público con la finalidad de adaptar nueva tecnología a través del tiempo y que es consistente con estándares".**

**Una especificación de apertura por esta definición es la independencia tecnológica, que se refiere a no depender de un hardware, software o producto específico proveniente de un vendedor en particular.**

**También se refiere a que está disponible a cualquier parte interesada en base a una igualdad de acceso a la información.**

**Más aún, está bajo el control de un forum público así que todas las partes afectadas pueden participar en su definición, mantenimiento y evolución.**

**La definición de sistemas abiertos dada por IEEE enfatiza la portabilidad e interoperabilidad basada en especificaciones de apertura.**

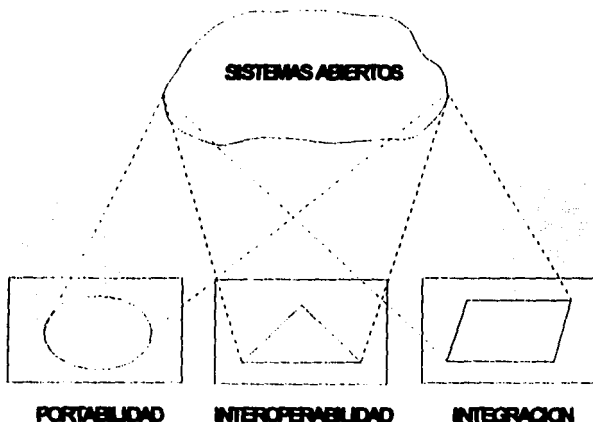
**La importancia de este estándar es que fué desarrollado en un forum en el que participó un amplio rango de la industria que incluyó representantes de diferentes sectores de la industria de la tecnología de la información, proveedores y usuarios.**

**El consenso usado por la IEEE asegura que la definición aceptada abarca un amplio rango de perspectivas. La definición no está limitada o excesivamente influenciada por intereses de ningún grupo en particular. Como resultado de la participación amplia que se tuvo en el proceso de la definición de POSIX para un sistema abierto, es que dicha definición puede ser acomodada a una gran variedad de ideas de lo que significa "ABIERTO".**

**El estándar POSIX también permite que la implementación de los sistemas abiertos crezca con el tiempo debido a que no se enfoca a un producto en específico o en áreas de tecnología, proporciona a los usuarios un marco que puede ser expandido como los cambios lo vayan necesitando.**

### 2.2.1 PORTABILIDAD-INTEROPERABILIDAD-INTEGRACION

Hasta aquí hay tres conceptos fundamentales que se asocian insistentemente con nuestro tema, PORTABILIDAD, INTEROPERABILIDAD E INTEGRACION y los sistemas abiertos se pueden ver desde estas tres perspectivas:



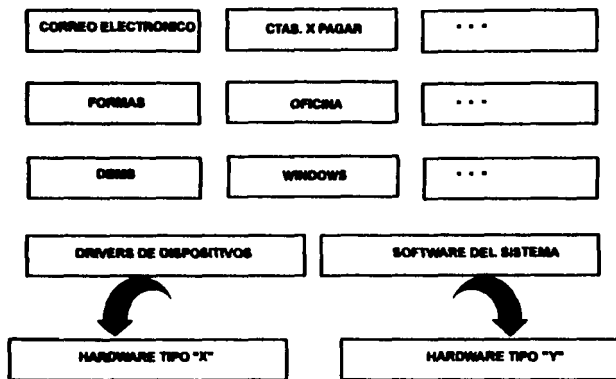
**FIGURA NUM.6.- SISTEMAS ABIERTOS.- PERSPECTIVAS**

## PORTABILIDAD

Desde un punto de vista muy general, es el aspecto de los componentes del sistema que permite que sea usado en diferentes ambientes.

La portabilidad del software es un elemento importante porque proporciona la funcionalidad básica deseada por el usuario final.

A continuación se esquematiza la portabilidad del software:



**FIGURA NUM. 7.- PORTABILIDAD DEL SOFTWARE**

La portabilidad la podemos entender partiendo de la base de que todos los sistemas (incluyendonos nosotros como seres humanos) existen en algún ambiente:

- Una computadora para operar adecuadamente requiere energía, temperatura y humedad adecuadas para operar.
- Un sistema operativo requiere un ambiente de hardware específico en el cual pueda ejecutarse.

- Un software de aplicación requiere un cierto sistema operativo para trabajar.

Si el sistema puede ser movido de un ambiente a otro, entonces se dice que es portable.

La portabilidad es determinada por las características tanto del sistema como de su medio ambiente.

Actualmente se está trabajando en estándares que se refieren específicamente al aspecto de portabilidad, lo están haciendo en 4 áreas:

- Sistemas Operativos.- POSIX, AT&T SVID, X/Open XVS, OSF.

- Administración de datos.- FTP, ISAM, SQL.

- Interfaces de programación para interfaces directas con los usuarios.- X, NeWS, Microsoft Windows, Open Look, Motif.

- Lenguajes de programación.- C, Fortran, Cobol, Pascal.

### **INTEROPERABILIDAD**

Se refiere a la capacidad de los componentes individuales en un sistema abierto para intercambiar información, ya no se habla solo de conectar, en este punto además de conexión se necesita que los diferentes componentes operen de manera armónica y compatible (es decir, se hablen, se entiendan e interactúen).

Lo anterior se esquematiza a continuación:



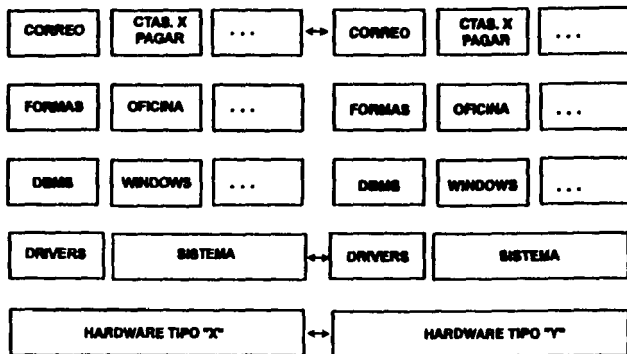


FIGURA NUM. 8.- INTEROPERABILIDAD

En el nivel más inferior la interoperabilidad implica comunicación entre tecnología de redes. En el nivel superior se refiere a operaciones UNO A UNO entre programas sobre diferentes computadoras.

Esta perspectiva de interoperabilidad se refiere al efectivo intercambio de información uno a uno.

Un sistema abierto soporta interoperabilidad si proporciona los mecanismos para intercambiar información entre componentes del sistema. Existen algunos ejemplos de componentes que permiten la interoperabilidad:

- Dos computadoras que son interconectadas utilizando protocolos tales como RS-232, X.25, IEEE 802.3, TCP/IP o SNA LU 6.2.
- Un programa de aplicación que usa una base de datos con la interface SQL.
- Una aplicación que usa un sistema de archivos con la interface POSIX.

### INTEGRACION

Este aspecto de los sistemas abiertos se refiere a la consistencia de varias interfaces desde el punto de vista máquina-humano, es decir, entre individuos y todo el hardware y software en el sistema.

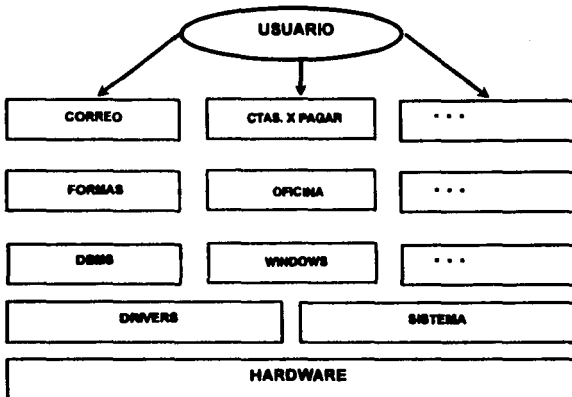


FIGURA NUM. 9.- INTEGRACION

La integración tiene que ver con la consistencia de las aplicaciones respecto a como las percibe el usuario, su presentación, comportamiento y funcionalidad en todas las aplicaciones, así tenemos que:

**La PRESENTACION de la aplicación.-** El usuario deberá realizar la misma operación en las diferentes aplicaciones con las que trabaje, ya sea que se trate de una hoja de cálculo, un procesador de palabras, etc., si éste desea borrar un elemento, la operación que lleve a cabo para el borrado, deberá ser la misma no importa en qué aplicación esté trabajando.

**El COMPORTAMIENTO de la aplicación.-** Se refiere a que en cada aplicación se deberá usar el mismo modelo de operación lógica.

La **FUNCIONALIDAD** de la aplicación.- Las aplicaciones deberán existir como si ellas fueran subcomponentes de un solo sistema, por ejemplo, una facilidad de oficina en la que se programen juntas, deberá hacer uso de las entradas del sub-sistema de calendario electrónico en lugar de que se le den a él por un medio diferente los datos de fecha.

### **2.2.2 ESTANDARES**

En diferentes partes de este documento ya se ha hecho uso de la palabra estándares, justo es que hablemos ahora de ellos.

Como entendimiento común de la palabra estándar podemos decir que es un patrón o un modelo al que se puede hacer referencia para determinadas cosas.

En nuestro caso, los estándares son un elemento sumamente importante para los sistemas abiertos y el futuro de la tecnología de la información.

Más de 250 subcomités de organismos oficiales trabajan con estándares referentes a la tecnología de la información, más de 1000 estándares han sido adoptados por estas organizaciones o están en proceso de desarrollo.

En el ambiente actual de la tecnología de la información se distinguen varios tipos de estándares:

**ESTANDARES DE FACTO**  
**ESTANDARES DE JURE O INDUSTRIAL**  
**ESTANDARES PROPIETARIOS**

En lo que se refiere a este último tipo, la considero dentro de los estándares de facto, por las razones que a continuación expongo.

Un **ESTANDAR DE FACTO** es el término aplicado a productos o sistemas que un proveedor ha puesto en el mercado, ha monopolizado su comercialización y que por circunstancias diversas otros proveedores tienden a emular, copiar o usar a fin de poder compartir dicho mercado.

El abuso de los estándares DE FACTO ha sido una de las razones que le han dado más importancia al hecho de que ya existan organismos reconocidos internacionalmente y que cuentan con el apoyo del sector más importante en la industria de cómputo para que de manera formal se establezcan estándares.

En los años 60's y 70's el surgimiento de estándares DE FACTO forzó a los usuarios a adoptar dependencias de proveedores que a la larga resultaron indeseables.

Un "ESTANDAR DE JURE" o "INDUSTRIAL" es creado por un organismo que desarrolla estándares y que tienen reconocimiento formal e internacional. Son desarrollados bajo estrictas reglas y son producto de importante consenso y de foros abiertos en los que sus integrantes tienen la oportunidad de intervenir en el proceso de su creación.

Ningún grupo aislado puede operar de forma independiente para crear un estándar para la industria, debe haber consenso. Si un grupo de proveedores crea un estándar para el cual no hay demanda por parte del usuario, fallará.

De la misma forma si los usuarios crean un estándar que los proveedores no puedan cumplir, tampoco se formalizará dicho estándar.

Cuando la participación en el desarrollo de los estándares es balanceada entre varios grupos, el consenso se aproxima a un estándar DE JURE.

Los estándares DE JURE no pueden ser cambiados sin la participación balanceada de varios grupos y para ellos se lleva a cabo un proceso formal de consenso.

Como ejemplos de estándares mencionaremos algunos:

- Open System Interconnection (OSI).
- Ethernet
- Posix.
- SQL.
- La mayoría de los lenguajes estándar.

### **2.2.3 ORGANISMOS INTERNACIONALES QUE TRABAJAN LOS ESTANDARES**

A continuación se relacionan algunas de las organizaciones que tienen que ver con los estándares.

**IEC.-** International Electrotechnical Commission. Fundada en 1906, esta organización se dedica a la estandarización en el área electrónica y eléctrica. La Secretaría General está en Genova y agrupa 43 países. Su función es idéntica a la que tiene la ISO. La ISO y la IEC han fundado el "Joint Technical Committee (JTC1)".

**CCITT.-** International Telegraph & Telephone Consultative Committee.- Fundada en 1956 y tiene su sede en Genova, este comité desarrolla estándares sobre servicios públicos de telecomunicaciones.

**IEEE.-** Institute of Electric and Electronics Engineers.- Es una asociación científica con 250,000 miembros distribuidos al rededor de todo el mundo. Desarrolla estándares referentes a los campos de la electrónica y a la computación.

**CEN.-** European Standards Committee.- Fundada en 1961, este comité comprende 18 organizaciones de estandarización. Está situada en Bruselas, desarrolla estándares europeos provenientes de estándares publicados por la EN (European Norm)

**CENELEC.-** European Committee of Electrotechnical Standardization (CENELEC). Es un cuerpo suplementario del CEN y un equivalente del IEC para Europa.

**ETSI.-** European Telecommunications Standards Institute.- Fundada en 1988 y establecida por Sophie Antipolis, este instituto desarrolla estándares europeos en el campo de las telecomunicaciones y especificaciones relacionadas al uso de redes públicas.

**ANSI.-** American National Standards Institute.- Este cuerpo coordina las actividades de especificación en los Estados Unidos y representa al país a nivel internacional. Juega un papel muy activo en la promoción de especificaciones de gráficas y lenguajes.

**BSI.-** British Standards Institute.- Es el equivalente al ANSI en Inglaterra.

**EIA.-** electronic Industry Association.- Esta asociación americana definió los estándares RS-232C y el RS-422.

**ITSC.-** Information Technology Steering Committee.- Fundada en 1988 para desarrollar perfiles funcionales basados en los estándares europeos y la tecnología de la información involucrada, este cuerpo desarrollo sus trabajos através de eventos denominados "Workshop" (Exposición de equipo) en

Europa sobre el forum de Sistemas Abiertos (EWOS). El National Institute of Standards and Technology (NIST) es el equivalente americano del ITSC.

ECMA.- European Computer Manufacturere's Association.- Fundada en 1960, comprende algunos fabricantes de computadores. Sugiere planes para estándares a la ISO y a la CCITT y está involucrado en la definición de perfiles funcionales del ITSC.

X/OPEN.- Fundada en 1984 y con base en Inglaterra, compuesta de un consejo de fabricantes, proveedores y usuarios. Define un perfil funcional basado en especificaciones de facto y estándares: El Common Applications Environment (CAE).

ISO (International Standards Organization) fué creada en 1946 para hacerse cargo de la estandarización a nivel internacional en todos los dominios excepto de la tecnología eléctrica y electrónica.

La secretaria general de dicha organización en Genova agrupó 91 países y 20,000 expertos trabajando sobre 2,400 comités; el comité técnico (TC), el sub-comité (SC) y el grupo de trabajo (WG).

Algunos proveedores consideran que la aplicación de los estándares tiene limitaciones potenciales y afirman que ellos adoptan para sus productos estándares comercialmente viables y no necesariamente adoptarán aquellos formulados por organismos internacionales como los antes mencionados.

Algunas razones por las que no los adoptan son las siguientes:

- a).- Los mecanismos para el desarrollo de los estándares que llevan a cabo grupos internacionales puede limitar la utilidad de un estándar.
- b).- Los estándares DeJure y por lo tanto, de productos de sistemas abiertos que implementan estándares De Jure aún no existen para todos los requerimientos.
- c).- El rendimiento de una aplicación o del sistema puede verse seriamente afectado por querer cumplir con estándares.

Atrás de todo ésto existen muchos intereses de tipo comercial que obviamente obligan a los fabricantes a no sujetarse a los estándares, dado que entre más opciones hay en el mercado, ellos tendrán más competencia entre sí. La tendencia hace suponer que llegará el día en que todos los

fabricantes se sujeten a los estándares. La realidad de hoy es que no todos se sujetan a todos los estándares y como consecuencia el usuario es el que se ve afectado.

#### **2.2.4 ABIERTO**

En el tema que se trata en este documento la palabra ABIERTO tiene un significado muy amplio, en diferentes grupos se tiene un enfoque diferente, dependiendo cual sea su perspectiva de lo que es un sistema abierto. A continuación se mencionará con qué se asocia la palabra ABIERTO:

**CON PLATAFORMAS ESTANDAR.-** Es común asociar el término "abierto" con aquel sistema basado en hardware estandar o en sistema operativo estandar. Un ejemplo de ello es una microcomputadora IBM o cualquiera compatible con ella. La mayoría de estas plataformas usan el mismo sistema operativo y corren muchas de las mismas aplicaciones.

También existe la creencia de que el sistema operativo determina si un sistema es o no abierto, esta creencia se basa en la teoría de que los sistemas que corren el mismo sistema operativo, correrán automáticamente las mismas aplicaciones, en función de esto se supone que:

- Se obtendrá así la portabilidad de las aplicaciones.
- Que interactuarán uno con otro logrando con esto interoperabilidad.
- Que la interface del usuario (monitor, teclado, mouse) aparecerán de la misma manera al usuario consiguiendo con ello la portabilidad del usuario mismo.

Lo anterior es verdad en teoría, pero hay muchas características necesarias para las aplicaciones de hoy que no son soportadas totalmente por el sistema operativo por si solo. Gráficas, redes, correo, computación distribuida, administración e intercambio de datos son ejemplos de características nuevas de las aplicaciones. En adición, los sistemas operativos serán alterados con cambios en el tipo de procesador, número de procesadores y número y tipo de terminales para soportar todas estas nuevas necesidades. Cualquiera de los cambios mencionados pueden afectar la portabilidad y la interoperabilidad.

**REDES.-** Algunos creen que un sistema abierto significa solo conectar juntos varios tipos de sistemas provenientes de diferentes fabricantes. Este

enfoque es enfatizado en muchos casos por el modelo de referencia OSI (Open Systems Interconnection).

Tanto los sistemas operativos como la tecnología de redes son elementos importantes en los sistemas abiertos pero por sí solos no proporcionan todas las capacidades que prometen los sistemas abiertos.

### **2.2.5 DISPONIBILIDAD Y ACCESIBILIDAD**

Hay varias formas de entender lo que es la disponibilidad.

La tecnología que está disponible de muchas fuentes, por ejemplo, las computadoras personales IBM compatibles, son consideradas por muchos como sistemas abiertos.

La tecnología que viene de una sola fuente pero que está disponible sobre varias plataformas, por ejemplo el MOTIF de la Open Software Foundation, es otro ejemplo de tecnología que se considera abierta. Cualquier vendedor puede mercadear Motif.

Otro ejemplo es el sistema operativo UNIX, originalmente ofrecido por la AT&T y ahora proporcionado por UNIX System Laboratories (USL), una subsidiaria de AT&T. USL es propietario de UNIX y controla sus licencias. Las implementaciones basadas en el sistema operativo UNIX de AT&T están disponibles desde muy diferentes fuentes, incluyendo tanto vendedores de hardware que ofrecen sistemas basados en UNIX como vendedores de software que proporcionan implementaciones para diferentes plataformas. En ambos casos, el proveedor puede adaptar la base tecnológica en su implementación.

Un ejemplo diferente es ORACLE, el sistema manejador de base de datos. Está disponible solamente por parte de la Oracle Corporation que es la compañía que lo vende o a través de vendedores de hardware quienes tienen un acuerdo cooperativo de mercado con Oracle, pero está disponible sobre una gran variedad de plataformas de diferentes vendedores, pero en este caso, los compradores de la licencia no tienen derecho de alterar la tecnología Oracle.

Otro término que también tiene que ver con disponibilidad es el conocido como "SISTEMA PROPIETARIO". En el sentido histórico y legal, este término indica que una tecnología es propiedad de una sola compañía y está



sujeta a licencia y control técnico por parte de la misma. Esto aplica en general para todos los productos hoy en día. Un ejemplo lo vemos con AT&T que tiene control sobre el sistema operativo UNIX System V, Microsoft tiene control sobre el sistema operativo MS-DOS, etc.

Las compañías controlan las dependencias arquitectónicas de sus productos así como la evolución futura de interfaces para los mismos.

Aquí surge un nuevo término que es importante aclarar:

### **2.2.6 ARQUITECTURA DE SISTEMAS**

**"A SYSTEM'S ARCHITECTURE DESCRIBES THE STRUCTURE OF INTERCONNECTION OF THE PARTS IN THE SYSTEM. THE ARCHITECTURE SPECIFIES COMPONENT INTERACTIONS, AND ULTIMATELY THE DETAILS OF THE INTERFACE AMONG INTEROPERATING COMPONENTES".**

Esta definición, mencionada en el libro "Open Systems" de Gary J. Nutt, nos indica que una arquitectura de sistemas nos describe la estructura de interconexión de las partes en el sistema, es decir, la forma en que están interconectadas las diferentes partes del sistema. La arquitectura también especifica las interacciones de los componentes y los detalles de la interface entre los componentes que están interoperando (como funcionan entre sí los componentes).

La arquitectura de sistemas así entendida y, como lo menciona J. Nutt, puede ser considerada como un sistema de diversas capas, en las cuales los componentes son producidos desde componentes de nivel inferior por ejemplo:

Los circuitos lógicos son utilizados para construir circuitos integrados.

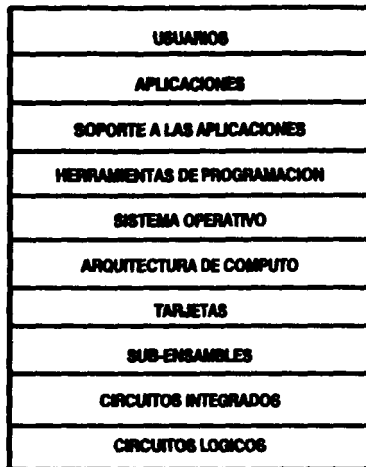
Los circuitos integrados son utilizados para construir sub-ensambles.

Los sub-ensambles son combinados para construir tarjetas.

Las tarjetas son utilizadas para construir computadoras.

De manera similar el software también puede ser conceptualizado en capas.

A continuación se visualiza el modelo en capas de una arquitectura de sistemas, como lo concibe Nutt:



**FIGURA NUM. 10.- ARQUITECTURA DE SISTEMAS**

El modelo anterior es útil para esconder los detalles de la implementación entre componentes que no se relacionan entre sí.

Por ejemplo, la implementación de "HERRAMIENTAS DE PROGRAMACION", no deberá depender de la implementación de las tarjetas en el hardware, aún cuando ambas son partes importantes de un sistema de cómputo.

Por otro lado, la funcionalidad del sistema está dividida de tal forma que hay funciones de bajo nivel y de alto nivel, donde las de alto nivel son diseñadas en términos de la interface para las funciones de bajo nivel, pero no de sus implementaciones, es decir, si la funcionalidad está dividida en capas

numeradas del 0 a la N-1 (donde 0 es la capa más inferior), entonces la funcionalidad del sistema que es implementada en la capa "i" depende solamente de la interface presentada por la capa i-1.

El diseñador de la capa "i" no conoce la implementación de la máquina i-1 y tampoco la naturaleza de la interface de las máquinas en niveles abajo del i-1 (A cada nivel se le puede llamar también máquina).

En el lenguaje de los sistemas abiertos, las capas individuales también son frecuentemente llamadas **PLATAFORMAS** para la capa superior.

Esto es, una plataforma proporciona una base sobre la cual alguna nueva capa puede ser construida.

Con el afán de facilitar la comprensión del concepto plataformas de cómputo, J. Nutt considera a las arquitecturas de cómputo desde dos puntos de vista:

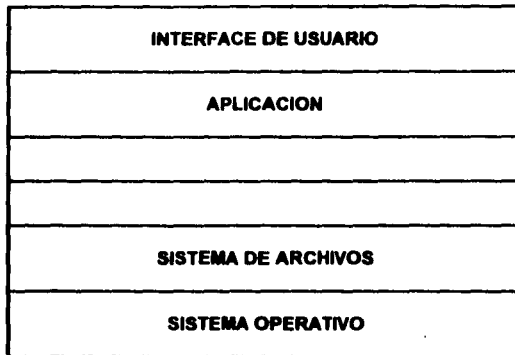
- a) Por medio de una **DIVISION VERTICAL** de la arquitectura en donde se divide a las partes de la misma en base a las funciones y cómo ellas son implementadas:

Se puede ejemplificar ésto con un sistema de oficina en el que se tienen varias aplicaciones tales como editor de documentos, calendario, correo electrónico y herramientas de soporte a decisiones. Sería como en el siguiente diagrama:



**FIGURA NUM. 11.- DIVISION VERTICAL**

- b) Por medio de una DIVISION HORIZONTAL de la arquitectura en donde se divide a la arquitectura en interfaces completas, mientras que se ignora lo referente a las implementaciones:



**FIGURA NUM. 12.- DIVISION HORIZONTAL**

Los enfoques anteriores los he mencionado porque, específicamente del enfoque de división horizontal, ha surgido el concepto de capas de la "MAQUINA ABSTRACTA".

### 2.2.7 MAQUINA ABSTRACTA

La arquitectura en este caso, es descrita como una familia de  $n+1$  máquinas abstractas,  $\{A(i) \mid 0 \leq i \leq n\}$ , donde  $A(i)$  es construida arriba de la interface de  $A(i-1)$ .

La división de capas de la máquina abstracta se ejemplifica en el siguiente esquema:

$A(n)$ :	USUARIO DEL SISTEMA
$A(n-1)=A(i)$ :	INTERFACE DEL USUARIO
... VARIAS CAPAS DE SOFTWARE DE APLICACION ...	
$A(s)$ :	INTERFACE DEL SISTEMA OPERATIVO
... VARIAS CAPAS DEL SISTEMA OPERATIVO ...	
$A(i+1)$ :	NUCLEO DEL SISTEMA OPERATIVO
$A(i)$ :	INTERFACE DEL SOFTWARE/HARDWARE
... VARIAS CAPAS DE COMPONENTES DE HARDWARE ...	
$A(0)$	

FIGURA NUM. 13.- MAQUINA ABSTRACTA

Cada  $A(i)$  es una descripción del comportamiento de un mecanismo que puede ser usado por un implementador de  $A(i+1)$ .

El usuario de  $A(i)$  no necesita saber los detalles de la implementación de  $A(i)$ , solamente su interface.

Los **USUARIOS FINALES** del sistema están en la capa superior de la jerarquía de la máquina abstracta.

La interface para  $A(n-1) = A(U)$  es la **INTERFACE DEL USUARIO** que define todos los aspectos de las interacciones máquina-humano.

Por su parte, el **HARDWARE** también está arreglado en sub-jerarquías.

En la figura el  $A(H)$  define la **INTERFACE PARA EL HARDWARE** de la máquina, esto es, el mecanismo que implementa  $A(H+1)$  es **SOFTWARE**.

En seguida,  $A(H)$  es descrito en terminos de instrucciones de máquina, registros, buses, controladores, interrupciones y memoria.

Puede haber varias máquinas abstractas entre  $A(n-1)$  y  $A(H)$ .

En sistemas sencillos, la jerarquía entre  $A(n-1)$  y  $A(H)$  es agrupada en una sola máquina abstracta; no hay distinción entre el sistema operativo y el programa de aplicación.

Para sistemas de propósito general siempre serán al menos dos máquinas abstractas: el software del sistema y el software de aplicaciones.

Conforme cada una de las máquinas aquí conceptualizadas lleguen a ser más complejas, será necesario considerar más capas.

La máquina abstracta correspondiente al **SOFTWARE DE APLICACION**, también puede ser subdividida en máquinas abstractas más pequeñas para manejar terminales virtuales (ventanas), herramientas de interface de usuario, almacenar y recuperar datos y para comunicación entre procesos.

Los conceptos de arquitectura, plataforma e inerfaces, son importantes pues tienen que ver directamente con el aspecto de **PORTABILIDAD** de las aplicaciones ya que como se mencionó en este enfoque de máquinas abstractas el uso de cada una debe ser independiente de su implementación, por lo que es posible construir nuevas funciones arriba de la implementación de una plataforma, y luego copiar la aplicación a otra implementación de la misma plataforma sin tener que invertir esfuerzo extra en la aplicación.

Con lo anterior, se supone que la aplicación es **PORTABLE** a través de diferentes implementaciones de la misma plataforma.

Las plataformas de cómputo deberán establecer un ambiente portable para los diseñadores de software en los que un componente de software pueda ser diseñado e implementado.

Se puede establecer una arquitectura general:

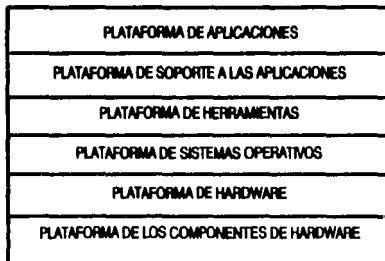


FIGURA NUM. 14.- PLATAFORMAS DE COMPUTO

En la figura anterior, la PLATAFORMA DE LOS COMPONENTES DE HARDWARE define los sub-ensambles que pueden ser usados para configurar el hardware de la computadora.

La PLATAFORMA DE HARDWARE define el juego de instrucciones para un sistema de cómputo.

La PLATAFORMA DE SISTEMAS OPERATIVOS es una máquina abstracta que realiza administración de los recursos y proporciona un sencillo sistema de archivos.

La PLATAFORMA DE HERRAMIENTAS extiende la capacidad de la plataforma de sistemas operativos mejorando la funcionalidad respecto a las facilidades de administración de datos.

La PLATAFORMA DE SOPORTE A LAS APLICACIONES extiende la capacidad de la plataforma de herramientas con paquetes de software horizontal como administradores de desktop.

La **PORTABILIDAD** es un elemento que aparece en cualquiera de las interfaces que existen entre cada plataforma.

La estandarización en la interface de las **PLATAFORMAS DE HARDWARE** y de **COMPONENTES DE HARDWARE** puede ser realizada fijando interface para los circuitos integrados, buses y dispositivos.

La interface entre las **PLATAFORMAS DE SISTEMAS OPERATIVOS Y HARDWARE** es la que comunmente es asociada con la **COMPATIBILIDAD DE CODIGO OBJETO**.

*Dos plataformas de hardware son compatibles a nivel código objeto si una puede ejecutar el mismo código objeto sobre cualquier implementación de dicha plataforma (sin recompilar, re-ligar o cambiar formatos).*

Por ejemplo, se pueden escribir programas para un 8088 y ser ejecutados en 8086, así como en otros microprocesadores tales como los 80186, 80286, 80386 y 80486. Si un fabricante piensa construir una computadora que sea compatible a nivel código objeto con una computadora producida por un fabricante diferente, entonces la máquina clone debe usar el mismo microprocesador y el mismo software para manejo de dispositivos (usualmente almacenado en ROM).

La interface entre las **PLATAFORMAS DE SISTEMAS OPERATIVOS Y HERRAMIENTAS** podrá estandarizarse sobre cualquier interface de sistema operativo de propósito general, tal como DOS, POSIX o VMS. En cada uno de estos casos, la interface es hecha para simular el llamado de procedimientos en un lenguaje procedural.

La implementación de la plataforma de herramientas usa un juego fijo de llamadas al sistema para aislarse a sí misma de implementaciones del sistema operativo en particular, y para simplificar la tarea de portar la plataforma de un caso de **PLATAFORMA DE SISTEMA OPERATIVO** a otro.

Ninguna interface entre las **PLATAFORMAS DE SOPORTE A APLICACIONES Y HERRAMIENTAS** han aún ganado gran aceptación, aún cuando hay ejemplos de algunas partes de la interface, se pueden mencionar interfaces de sistemas de ventanas o la capa de transporte del modelo OSI. La interface entre las plataformas de aplicación y soporte a aplicaciones también es muy inmadura para que se estandarice.



Es interesante mencionar como Andrew S. Tanenbaum en su libro "Organización de Computadoras un enfoque Estructurado", aborda el tema para comprender más claramente la estructura de las computadoras; establece que conviene imaginarse la existencia de una computadora hipotética o MAQUINA VIRTUAL en la que existen diferentes niveles, dando origen así al término "Máquina multinivel", así mismo, enfatiza en que cada máquina tiene algún lenguaje de máquina que consiste en todas las instrucciones que puede ejecutar (cada nivel), así que una máquina con niveles puede verse como n máquinas virtuales diferentes, cada una de las cuales tiene un lenguaje de máquina especial.

Indica que la mayoría de las computadoras actuales constan de dos o más niveles, el ejemplo que él esquematiza es el siguiente:

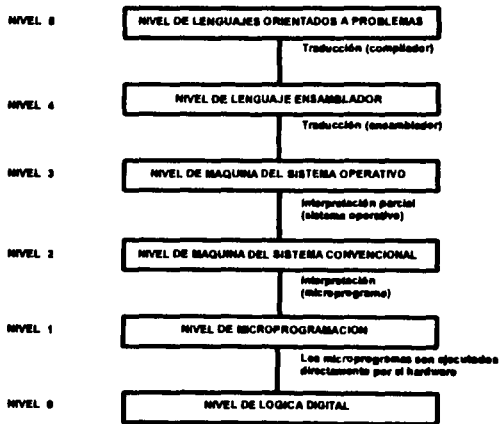


FIGURA NUM. 15.- MAQUINA MULTINIVEL

El método que soporta cada nivel está indicado por debajo de él y entre paréntesis el nombre del programa que lo soporta.

El nivel 0 que es el más inferior, es el hardware de la máquina, sus circuitos ejecutan los programas escritos en el lenguaje de máquina del nivel 1.

El nivel 1 es el que conforma el verdadero nivel del lenguaje de máquina, como no sucede en el nivel 0 en donde todavía no existe el concepto de programa como conjunto de instrucciones a realizar, en el nivel 1 existe ya un programa llamado microprograma, cuya función es interpretar las instrucciones del nivel 2.

Cada máquina de nivel 1 tiene uno o más microprogramas que pueden ejecutarse en ella. Cada microprograma define implícitamente un lenguaje de nivel 2.

Al nivel 2 se le denomina nivel de máquina convencional ya que tiene mucho en común entre máquinas de distintos fabricantes. A este respecto señala que cada fabricante publica un manual para cada una de las computadoras que vende, denominado "manual de referencia del lenguaje de máquina", o "principios de funcionamiento de la computadora", dichos manuales tratan de la máquina virtual del nivel 2, no de la máquina real de nivel 1. Cuando describen el juego de instrucciones de la máquina, de hecho describen las instrucciones que el microprograma lleva a cabo de manera interpretativa y no las verdaderas instrucciones del hardware. Si un fabricante proporcionara para una de sus máquinas dos intérpretes que interpretaran dos lenguajes de máquina de nivel 2 diferentes, necesitaría proporcionar dos manuales de lenguaje de máquina, uno para cada intérprete.

El nivel 3 es un nivel híbrido porque la mayoría de las instrucciones de su lenguaje están también en el lenguaje de nivel 2 (señala que no hay razón que impida que una instrucción de un nivel se presente también en otros niveles).

En este nivel existe un nuevo conjunto de instrucciones, una diferente organización de la memoria y la posibilidad de ejecutar dos o más programas en paralelo, estas nuevas posibilidades las lleva a cabo un intérprete que actúa en el nivel 2 al que tradicionalmente se le llama sistema operativo.

Por lo que respecta a las instrucciones del nivel 3 que son idénticas a las del nivel 2, las lleva a cabo directamente el microprograma en lugar de efectuarlas el sistema operativo, es decir, algunas de las instrucciones del nivel 3 las interpreta el sistema operativo y otras las interpreta directamente el microprograma.

**El nivel 4 tiene una diferencia fundamental con respecto a los niveles inferiores ya que éstos tres últimos (1,2, y 3) no están diseñados para que los utilicen los programadores corrientes, están concebidos principalmente para ejecutar los intérpretes y traductores necesarios para que sean soportados los niveles superiores.**

**Los niveles 4 y superiores, están pensados para los programadores de aplicaciones.**

**En los niveles 2 y 3 siempre se interpreta.**

**En los niveles 4, 5 y superiores normalmente, aunque no siempre, se soportan por traducción.**

**En los niveles 1, 2 y 4 los lenguajes de máquina siempre son numéricos, los programas escritos en ellos constan de largas series de números.**

**A partir del nivel 4 los lenguajes contienen palabras y abreviaturas.**

**En el nivel 4 o nivel de lenguaje ensamblador es más bien una forma simbólica de uno de los lenguajes subyacentes. Los programas escritos en lenguaje ensamblador se traducen primero a un lenguaje de nivel 1, 2 o 3 y luego se interpretan por la máquina real o virtual apropiada. El programa que realiza la traducción se llama ensamblador.**

**El nivel 5 contiene los lenguajes diseñados a ser usados por programadores de aplicaciones con problemas a resolver, a estos lenguajes se les denomina LENGUAJES DE ALTO NIVEL, ejemplos: BASIC, C, COBOL, FORTRAN, LISP, etc. Por lo general los programas escritos en estos lenguajes se traducen a los niveles 3 o 4 por medio de traductores conocidos como compiladores, aunque en ocasiones también pueden ser interpretados.**

**El nivel 6 y superiores consisten en paquetes de programas diseñados para proporcionar máquinas específicamente adaptadas a ciertas aplicaciones y que contienen gran cantidad e información en ellas.**

**Tanenbaum resume:**

**Lo esencial es recordar que las computadoras están diseñadas como una serie de niveles, cada uno construido sobre su predecesor.**

**Cada nivel representa una abstracción distinta, con objetos y operaciones diferentes.**

**Arquitectura es el conjunto de tipos de datos, operaciones y características de cada nivel. La arquitectura se refiere a aquellos aspectos que son visibles para el usuario de cada nivel.**

**Las particularidades que ve el programador, como la cantidad de memoria disponible, forman parte de la arquitectura.**

**Los aspectos de implementación tales como qué tecnología de chip se usa para elaborar la memoria, NO PERTENECEN A LA ARQUITECTURA.**

## **2.3 MODELO MUSIC**

Por un lado, son demasiados los conceptos y definiciones que tienen que ver con los sistemas abiertos y por otro, los requerimientos de las aplicaciones deberán estar claros y bien definidos para traducir éstos hacia un ambiente de cómputo que soporte dichas aplicaciones dentro de un marco de sistemas abiertos.

Actualmente hay varios enfoques a este respecto, pero realmente no contamos con un modelo generalizado ni bien experimentado al cual apearnos a la hora de concebir nuestro ambiente de sistemas abiertos, sin embargo, en el libro *Open Systems Handbook, A Guide to Building Open Systems*, editado por Digital Equipment Corporation, menciona uno que me pareció bastante interesante y que servirá para enfocar la discusión de sistemas abiertos.

El modelo referido es una herramienta desarrollada por la Central Computer and Telecommunications Agency (CCTA), que es un organismo británico.

Este modelo se enfoca específicamente sobre aquellas áreas de interacción e interface que son críticas para que una plataforma de aplicación funcione como un sistema abierto, proporciona un marco de referencia claro y entendible para la discusión de los elementos de sistemas abiertos, pues permite organizar y categorizar las necesidades y se comparan con alternativas que ya tienen disponibles organizaciones de estándares y proveedores.

Este modelo usa el acrónimo MUSIC para categorizar los elementos base de un sistema abierto, así como sus estándares relacionados. Aunque es poco conocido en la industria nacional, la fuente consultada señala que por sus características, podría llegar a ser una herramienta muy útil para los que están tratando de implementar sistemas abiertos. A este respecto, considero que para determinar si es o no adecuado, solo con la experiencia de utilizarlo podremos normar nuestro criterio.

MUSIC está integrado por diferentes elementos que agrupan componentes diseñados para proporcionar capacidades apropiadas para cada dominio funcional, algunos componentes incluyen capacidades que abarcan varios elementos.

La agrupación de elementos que integran el modelo, es de donde se originó el acrónimo:

### **2.3 MODELO MUSIC**

Por un lado, son demasiados los conceptos y definiciones que tienen que ver con los sistemas abiertos y por otro, los requerimientos de las aplicaciones deberán estar claros y bien definidos para traducir éstos hacia un ambiente de cómputo que soporte dichas aplicaciones dentro de un marco de sistemas abiertos.

Actualmente hay varios enfoques a este respecto, pero realmente no contamos con un modelo generalizado ni bien experimentado al cual apearnos a la hora de concebir nuestro ambiente de sistemas abiertos, sin embargo, en el libro *Open Systems Handbook, A Guide to Building Open Systems*, editado por Digital Equipment Corporation, menciona uno que me pareció bastante interesante y que servirá para enfocar la discusión de sistemas abiertos.

El modelo referido es una herramienta desarrollada por la Central Computer and Telecommunications Agency (CCTA), que es un organismo británico.

Este modelo se enfoca específicamente sobre aquellas áreas de interacción e interface que son críticas para que una plataforma de aplicación funcione como un sistema abierto, proporciona un marco de referencia claro y entendible para la discusión de los elementos de sistemas abiertos, pues permite organizar y categorizar las necesidades y se comparan con alternativas que ya tienen disponibles organizaciones de estándares y proveedores.

Este modelo usa el acrónimo MUSIC para categorizar los elementos base de un sistema abierto, así como sus estándares relacionados. Aunque es poco conocido en la industria nacional, la fuente consultada señala que por sus características, podría llegar a ser una herramienta muy útil para los que están tratando de implementar sistemas abiertos. A este respecto, considero que para determinar si es o no adecuado, solo con la experiencia de utilizarlo podremos normar nuestro criterio.

MUSIC está integrado por diferentes elementos que agrupan componentes diseñados para proporcionar capacidades apropiadas para cada dominio funcional, algunos componentes incluyen capacidades que abarcan varios elementos.

La agrupación de elementos que integran el modelo, es de donde se originó el acrónimo:

<b>M</b>	<b>Management.</b>
<b>U</b>	<b>User Interface.</b>
<b>S</b>	<b>Service Interfaces for Programs.</b>
<b>I</b>	<b>Information and Data Formats.</b>
<b>C</b>	<b>Communication Interfaces.</b>

A continuación veremos cada uno de los elementos del modelo.

### **2.3.1 M.- MANAGEMENT (ADMINISTRACION)**

Este elemento se refiere a las siguientes funciones:

- Administración del sistema operativo.
- Seguridad.
- Administración de la red.
- Administración de los recursos.
- Contabilidad (Accounting).
- Administración de la configuración.
- Rendimiento.

Elementos tales como seguridad y contabilidad, necesitan ser soportados a través de un rango amplio de diferentes tipos de sistemas, para que todos los recursos puedan ser accedidos y administrados a través de un ambiente de cómputo completo y distribuido a través de toda una organización.

Los usuarios beneficiados con los componentes del elemento "M" son precisamente los administradores del sistema, operadores y administradores de red ya que la promesa es dotarlos de portabilidad en esta área y facilitar la administración de los recursos en ambientes distribuidos ya que es deseable que se centralicen este tipo de funciones.

La tecnología va cambiando, y ya no es común que aplique aquel esquema en el que teníamos a un administrador del sistema y a un administrador de la red (como en ambientes de mainframe), el alcance de la administración total de la tecnología de la información en una empresa, hoy requiere de un soporte para ambientes distribuidos, de tal forma que de manera centralizada se tenga el control total de los recursos: sistema, aplicaciones y redes.

La administración deberá ser corporativa, es decir, que abarque toda la compañía, los conceptos y herramientas utilizados para manejarlos deberán quedar aplicables aún cuando se experimenten cambios importantes, crecimiento en cualquier parte de la configuración o incluso, en su totalidad.

Haremos una revisión del concepto de administración de redes OSI para analizar las diferentes áreas funcionales de la administración.

La administración de redes OSI fundamentalmente se concentra en la configuración de la red, fallas de los componentes, nivel de rendimiento, seguridad y contabilización de su uso. En OSI estos elementos de la administración de la red, algunas veces son conocidos como áreas funcionales.

Cada área funcional de la administración se describe a continuación:

#### **ADMINISTRACION DE LA CONFIGURACION**

Se refiere a mantener un conocimiento exacto de la topología física y lógica de la red, que incluye el inventario de los componentes y la interconectividad. Esta información podrá ser utilizada para generar un desplegado de la topología de la red en una consola para administrarla. También incluye las facilidades para poner parámetros, inicializar y deshabilitar recursos, reconfigurar, así como recolectar información acerca del estado de los componentes de la red.

La administración de nombres se incluye a veces en esta área funcional, de tal forma que los recursos puedan ser manejados por nombres en lugar de manejarlos por direcciones. Sin embargo, esta función parece mejor ubicada en los servicios de directorio.

#### **ADMINISTRACION DE FALLAS**

Se refiere a mantener el conocimiento del estado actual del sistema, de la activación y desactivación de cada uno de los componentes y la forma en que están interconectados en la red y de las actividades actuales respecto a la recuperación de cualquier unidad que falle. Esta información puede ser utilizada con el mapa de la topología para indicar que tramos de la red están funcionando y como lo están haciendo. Las herramientas para administración de fallas deberán incluir las características necesarias para



que por medio de desplegados de diferentes colores en la pantalla, se puedan localizar los puntos neurálgicos de la red.

En este punto también se debe considerar el registro de las ocurrencias de falla (log's), guías para diagnosticar fallas, herramientas de diagnóstico para determinar el origen y tomar las fallas en consideración para posteriores diagnósticos.

### **ADMINISTRACION DEL RENDIMIENTO**

Deberá mantenerse un conocimiento exacto (guardar la información) del rendimiento anterior y actual de la red, que incluirá parámetros estadísticos tales como retardo, THROUGHPUT (\*), disponibilidad, paquetes por segundo, bits por segundo y número de retransmisiones.

(\* La tasa de información sostenible a la que la información puede ser enviada y recibida, la cantidad de datos que pueden ser transferidos a través de un dispositivo.

### **ADMINISTRACION DE LA SEGURIDAD**

Consisten en saber exactamente quién está utilizando la red y que el acceso a los recursos esté dentro de los límites autorizados para cada usuario.

La terminal del administrador de la red deberá proporcionar alertas para tener conocimiento de cuando se detecten eventos que violen la seguridad del sistema.

También deberá contarse con una base de datos en línea de auditoría para realizar un análisis más detallado. La administración de la seguridad también incluye controles de procedimientos de claves confidenciales (por ejemplo: forzar a que se cambien las claves de acceso a los recursos).

### **ADMINISTRACION DE LA CONTABILIDAD**

Se refiere a mantener un conocimiento de cómo y por quien están siendo utilizados los recursos de la red. Esta información puede ser necesaria para propósitos estadísticos o de facturación de servicios. Generalmente no se desplegará, excepto en respuesta a una consulta.

**La administración de la contabilización puede también proporcionar límites sobre el uso de la red, sus recursos y mecanismos administrativos para poner estos límites.**

**En cada párrafo donde se describen cada uno de las 5 principales áreas funcionales de la administración de la red, se hace énfasis en "Manteniendo el conocimiento de", para enfatizar el punto de que la administración de la red intenta fundamentalmente proveer conocimiento, no números. Este conocimiento puede ser mejorado por la manera en la que la información se le despliega y esté disponible al administrador de la red. Estos son temas importantes independientemente de que se trate o no de una red OSI.**

**El interés básico aquí es que el administrador no necesariamente deberá ser una persona de alto grado técnico (un guru).**

**Es necesario tener suficiente conocimiento para poder aislar, reconocer y resolver problemas. Se debe tener concentrada la información para que el administrador base en ella sus decisiones.**

**Los datos pueden estar en forma de resumen de datos de bajo nivel y pueden resultar en información aún más condensada que se envíe al siguiente nivel más alto en el esquema de administración.**

**El manejar demasiada información obtenida de la operación de una red, puede ser demasiado complejo, por lo que respecta a OSI la información manejada es denominada Base de Información de Administración (Management Information Base o MIB).**

**La MIB incluye todos los datos discutidos, incluyendo la configuración de la red, estado de fallas, datos de rendimiento, de seguridad y de contabilidad. La naturaleza de la información y su formato son los temas de OSI.**

**Las herramientas para agrupar la información de los dispositivos de la red y para controlar los parámetros en los dispositivos de la red también son temas de OSI. Sin embargo, las herramientas para manipular los datos y para desplegarlos a un operador no son de la competencia de OSI, incluyen la utilización de un sistema de administración de base de datos relacional para acceder los datos basados en consultas estructuradas. La información en la MIB incluye los objetos manejados, sus atributos, las operaciones que ellos realizan y las notificaciones que ellos pueden proporcionar.**

Los parámetros de la MIB son definidos como estructuras de datos utilizando ASN.1.- Notación de sintaxis abstracta (ASN.1 es el estándar ISO 8824).

La mayoría de estas estructuras de datos son bastante simples.

Sin embargo, es útil tener una definición precisa de los datos de la MIB, especialmente cuando múltiples vendedores están involucrados en la agrupación, diseminación y proceso de datos MIB.

### **ESTANDARES PARA ADMINISTRACION DE REDES**

El concepto de sistemas abiertos basa sus fundamentos en el manejo de estándares y por lo que respecta a esta área, el trabajo que se viene haciendo es muy reciente, la mayoría está enfocado sobre la definición de objetos manejados y administración de funciones en un ambiente en donde por diferentes lugares se hallan distribuidos equipos originarios de diferentes proveedores.

Comenzó a mediados de los 80's cuando empezó a trabajarse en la parte de administración de redes OSI.

A finales de los 80's, la Internet Engineering Task Force (IETF) de la Internet Activities Board (IAB) comenzó a trabajar sobre un protocolo estándar para administración de redes TCP/IP. La IAB es la organización responsable del mantenimiento y evolución del protocolo Internet.

Tanto la ISO (International Standard Organization) y el CCITT (Consultative Committee for International Telephone and Telegraph) son los organismos que han estado muy ocupados en la definición y desarrollo de estándares para la componente de administración.

La ISO considera que la estructura general de administración sea una extensión del modelo de referencia OSI, está por lo tanto numerado como ISO 7498-4.

El CCITT lo considera como parte introductoria de una nueva serie de recomendaciones X.700 acerca de la administración de redes, por lo tanto, el X.700 corresponde al 7498-4 de la ISO.

Ambos estándares proporcionan un marco general para la administración de sistemas, éste es, el 10040 de la ISO y el X.701 del CCITT.

**ISO 10040 System Management Overview (SMO).** Este estándar complementa al 7498-4, especifica un modelo para administración de sistemas y la relación entre estándares de administración de sistemas. Identifica los servicios de capas OSI usado por las entidades administradas.

Los siguientes documentos en la secuencia, son los servicios comunes de administración y los protocolos, los cuales tienen el número 9595/9596 en la OSI, por lo que respecta al CCITT, son los número X.710/X.711.

**ISO 9595.- Common Management Information Services (CMIS).** Este estándar define los servicios de administración común básicos: notificación, operación y asociación. También define la selección de objetos manejados. Junto con CMIP, estos servicios permiten a las herramientas de administración de red provenientes de diferentes vendedores, comunicarse unas con otras como si fueran pares.

**ISO 9596-1.- Common Management Information Protocol (CMIP).** Este estándar se refiere a un juego de reglas que gobiernan como se intercambia la información entre aplicaciones de administración en redes separadas. Está enfocado a los requerimientos de administración de equipo de comunicaciones en una red grande heterogénea, ya sea centralizada o distribuida.

Estos dos documentos y su condición de estándares internacionales, son la base para la mayoría de los enfoques de administración de red sean o no OSI. Se supone que los implementadores de productos de administración podrán basarse tranquilamente en estos estándares para sus desarrollos.

Algunas partes de los estándares para administración de red son los ISO (DIS.- Anteproyecto de Norma Internacional) 10165-1 al -4 y las series X.720 del CCITT, los cuales cubren el modelo informacional de administración.

Otras partes en una condición similar, incluyen los comprendidos en la serie (DIS) 10164-1 al -11 y las series X.730 del CCITT, los cuales cubren una amplia variedad de funciones detalladas tales como administración de objetos, administración del estado, administración de las relaciones, alarmas, eventos y logs.

---

<sup>1</sup>DIS.- Draft International Standard

Hay otras funciones que también están siendo desarrolladas:

Guías para auditoría de la seguridad, controles de acceso, medidas de contabilidad, monitoreo de cargas de trabajo, administración de pruebas, análisis estadístico y administración del tiempo en la red.

### **IMPLEMENTACIONES DE ADMINISTRACION EN REDES OSI**

La implementación de la administración en redes OSI ha estado en desarrollo por algún tiempo por parte de las principales industrias de la tecnología de la información. Sin embargo la carencia de estándares establecidos y acuerdos sobre su implementación han limitado severamente la producción de este tipo de productos a la fecha. Al igual que con otros protocolos, la participación por parte de las organizaciones de estándares: U.S. National Institute of Standards and Technology (NIST) y por la European and Asian, ha sido de gran importancia.

Varios vendedores han intentado llenar este vacío proporcionando herramientas para administración de redes OSI. Estos productos incluyen:

El Open View de HP.

El accumaster de AT&T.

El Enterprise Management Architecture (EMA) de Digital.

Por su parte IBM ya integró su producto denominado NetView para administración tanto de redes OSI como su ambiente propietario SNA.

### **ACTIVIDADES SOBRE ADMINISTRACION EN TCP/IP**

Las actividades acerca de la administración de redes, por parte de la Internet Engineering Task Force (IETF), se han enfocado sobre un protocolo para administración de redes TCP/IP y tienen la intención, en un futuro no muy lejano, de utilizar para la administración de redes TCP/IP, los estándares que para este mismo fin está trabajando la OSI.

**RFC<sup>2</sup> 1157 Simple Network Management Protocol (SNMP).**- Este estándar se enfoca al agrupamiento de estaciones de administración de red y elementos de red (hosts, ruteadores/compuestas, servidores de terminal). Es usado para comunicar información de administración entre estaciones de administración de red a los que denomina "agente administrador" y a los elementos administrados en la red.

SNMP explícitamente minimiza el número y complejidad de funciones de administración realizadas por el agente administrador.

También con este enfoque, el grado de funciones de administración que es remotamente soportado se incrementa, imponiendo así menos restricciones posibles sobre la forma y sofisticación de herramientas para administración.

Los documentos Internet para SNMP son:

RFC 1157 Simple Network Management Protocol.

RFC 1155 Structure and Identifications of Management Information for TCP/IP-RFC 1156 Management Information Base for Network Management of TCP/IP-based Inter nets.

### **COEXISTENCIA DE TCP/IP Y OSI**

Una de las necesidades más urgentes de la comunidad Internet, es determinar una coexistencia estratégica entre TCP y OSI, las investigaciones a este respecto están realizándose por parte de ambas comunidades, Internet y OSI.

Hay varios enfoques acerca del problema, algunos basados en protocolos y otros basados en servicios. De momento el RFC 1006, que se refiere a los servicios de transporte ISO sobre la TCP, permite a los usuarios Internet correr aplicaciones OSI sobre la pila de protocolos TCP/IP. Sin embargo el RFC no proporciona comunicación con la pila completa OSI. Eventualmente, implementaciones de la capa de comunicaciones OSI se espera reemplace las capas de comunicación del protocolo TCP/IP.

---

<sup>2</sup> RFC. Request for Comments

## **ADMINISTRACION DE SISTEMAS EN LA IEEE**

### **POSIX 1003.7**

El grupo de trabajo del estándar POSIX 1003.7, está desarrollando un juego común de utilerías y de interfaces del sistema destinadas a los administradores para instalar, configurar y dar mantenimiento a los ambientes de sistemas operativos de cómputo. Los tópicos que están siendo considerados son:

Administración de:

- Usuarios.
- Dispositivos.
- Medios.
- Sistema de archivos incluyendo respaldo y recuperación.
- Condición del sistema.
- Procesos.
- Rendimiento.
- Monitoreo del sistema.
- Contabilidad.
- Spool.
- Software.
- Servicios de comunicación.

Los objetivos importantes del grupo de trabajo son proporcionar interfaces consistentes orientadas a objetos y desarrollar la interoperabilidad acerca de redes de múltiples vendedores. El modelo OSI es el punto de partida para este enfoque de sistemas en red.

### **POSIX 1003.15**

Este grupo de trabajo está concentrado sobre colas batch y, como parte de esta tarea, está definiendo una interface administrativa.

### **AMBIENTE DE ADMINISTRACION DISTRIBUIDA DE LA OSF (DME- DISTRIBUTED MANAGEMENT ENVIRONMENT)**

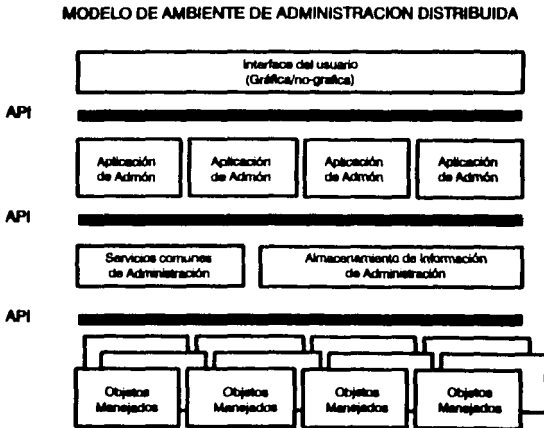
La OSF ha emitido un RFT (request for technology) para que el DME sea incluido en su Distributed Computing Environment (DCE).

El objetivo principal del DME RFT es identificar un marco de referencia común para la administración de sistemas en red. Este marco de referencia soportará la administración consistente de un amplio rango de sistemas, desde sistemas solos a aquellos en que se trate de un ambiente distribuido.

El marco de referencia DME definirá interfaces de programación de aplicaciones (API.- Application Programming Interfaces), que las aplicaciones deberán utilizar para:

- Invocar servicios comunes de administración.
- Almacenar y recuperar información de administración.
- Intercambiar información de administración con objetos manejados en sistemas remotos y locales.

Los componentes del modelo DME se representan en la siguiente figura:





De acuerdo a OSF, los componentes incluidos en el DME son:

**Interface del Usuario.-** Soportará interfaces de carácter (no gráficas) y gráficas. Será fácil de usar y será consistente para todas las aplicaciones de administración.

**Aplicaciones de administración.-** El DME incluirá aplicaciones que proporcionen el medio básico para manejo de sistemas abiertos. Realizará tareas de administración tales como reinicialización remota de nodos en la red, reconfiguración remota de parámetros de los nodos en la red.

**Servicios comunes de administración.-** Soportarán el modelo de administración proporcionando interfaces de programación usadas por las aplicaciones de administración. Estos servicios incluirán funciones tales como administración de:

Comunicaciones, eventos, registro de los mismos y objetos.

Con este modelo la promesa es que podrán desarrollarse aplicaciones de administración portables y permitirán la interoperabilidad entre sistemas heterogéneos.

**Objetos.-** Un objeto es la representación de recursos de cómputo o cualquier otra entidad conceptual en un ambiente de cómputo. Un ejemplo de recursos del sistema son un sistema de archivos (file system), el cual es representado y manejado a través de uno o más objetos asociados. Para hacer portable a una aplicación de administración, extensible y adaptable dentro de este modelo, se requerirá de una interface general para manejo de objetos.

### ***EVOLUCION HACIA ADMINISTRACION DE REDES OSI***

Tres enfoques dominan hoy el mercado por lo que respecta a redes sofisticadas, dos de éstos son propietarios: el SNA de IBM y el DECnet de Digital. El tercero es el estandar de la industria: Transmission Control Protocol/Internet Protocol (TCP/IP).

La fuente consultada señala que los tres tienen planes para evolucionar hacia OSI, incluyendo la parte de administración de redes OSI.

La base para la mayoría de las estrategias de migración hacia administración de redes OSI está en el uso de aquellas partes de la administración de redes OSI que está actualmente disponibles y en proporcionar soluciones provisionales cuando los estándares OSI aún no están disponibles.

La MIB es la llave para la mayoría de las estrategias. Si un sistema de administración de red es construido para usar una MIB OSI, entonces las herramientas de soporte de software que los usuarios y los programas de aplicación usan para operar con la MIB pueden ser salvadas cuando la solución total OSI llegue a estar disponible.

Estas herramientas típicamente incluyen sistemas de administración de base de datos relacionales con capacidades robustas de consulta y manipulación de datos.

La otra interfase clave con la MIB es la que se refiere a los dispositivos que están siendo manejados. Este punto es donde la mayoría de las soluciones difieren.

Por otra parte, SNMP ha llegado a ser el enfoque predominante en cuanto a la administración de redes en la comunidad TCP/IP. Ha sido implementado por los principales vendedores de ruteadores IP, ruteadores multi-protocolo y muchos puentes para LAN.

### **2.3.2 U.- USER INTERFACE (INTERFACE DE USUARIO)**

Una de las promesas con las que deberá cumplir el concepto de sistemas abiertos es lograr la portabilidad del usuario, esto significa mover al personal de un sistema o aplicación a otro sin tener que volver a capacitarlo o si requiere de capacitación, que ésta sea mínima. Lo anterior puede lograrse en parte, desarrollando estándares que permitan que la presentación y comportamiento de los sistemas de cómputo sean consistentes para el usuario.

La interfase del usuario es uno de los componente clave en un ambiente de cómputo abierto ya que ésta significa interacción directa entre el ser humano y la computadora. La interfase en este sentido cubre aspectos muy amplios que abarcan diferentes teclados y dispositivos de interacción que tienen que ver con elementos ergonómicos y de manejo de la computadora.

La inconsistencia actual en la presentación de los sistemas hacia el usuario tiene inconvenientes tales como:

- Errores al maniobrar los sistemas.
- Frustración del usuario al tener que trabajar con un ambiente nuevo.
- Productividad más retardada como consecuencia del entrenamiento necesario en nuevos ambientes.
- Resistencia al cambio por parte del usuario.

En el modelo MUSIC el elemento interface de usuario se ha dividido en:

- El conjunto de interacciones que generalmente ocurren entre el usuario y la plataforma de cómputo independientemente de la aplicación.

Ejemplo: Forma en que el usuario activa la aplicación o la forma en que salva un archivo.

- Interface entre el usuario y la aplicación misma.
- Interfaces de programación de aplicaciones, que establece la interacción requerida para que las aplicaciones obtengan servicios de la interface del usuario.

Actualmente han llegado a ser muy populares los sistemas de ventanas (WINDOWS) y gran parte del trabajo actual en el área de interface de usuario está enfocado en el desarrollo de estándares para sistemas de ventanas. Todavía no hay estándares formales disponibles en esta área pero hay uno que se considera estandar de facto para los sistemas basados en ventanas: "X WINDOW SYSTEM".

### **SISTEMA "X WINDOW SYSTEM" ( Conocido también como "X")**

El sistema "X Window" o también conocido como "X", fué creado en el Instituto de Tecnología de Massachusetts (MIT) en 1984 para trabajar en ambientes UNIX. Este fué derivado del sistema "W" de la Universidad de Stanford. En enero de 1986, Digital Equipment Corporation anunció el primer uso comercial de "X" en su VAXstation-II/GPX. Conforme otras compañías se involucraron con este producto, se le hicieron diferentes contribuciones y

después en 1986, Digital contribuyó con recursos humanos al proyecto "X", de tal forma que "X" fué reescrito a la versión 11. Durante 1987 el MIT y 9 grandes corporaciones se unieron para crear el "Consortio X", fundado por las compañías participantes y que tiene como objetivo soportar y controlar el desarrollo y evolución de "X".

El código fuente para "X" está disponible públicamente.

El comité X3H3.6 y el comité técnico X3 sobre gráficas de computadoras de la ANSI, han seleccionado al "X Window System" como el protocolo para ventanas y a la biblioteca Xlib como un punto inicial para la definición de sus estándares de interface de usuario. NIST ha adoptado la versión 11 del "X", la Xlib, los intrínsecos "Xt" (X toolkit) y el formato de distribución "bitmap" en FIPS 158.

NOTA: FIPS.- NIST Federal Information Processing Standard<sup>3</sup>.

La X/Open también ha apoyado el protocolo "X", los intrínsecos del Sistema "X Window" y la Xlib como parte de su "COMMON APPLICATIONS ENVIRONMENT".

El "X toolkit" es comunmente usado en la implementación de productos comerciales. Ha sido adaptado como parte de la interfaces gráficas de usuario tales como Motif de OSF y de OpenLook de AT&T.

El objetivo de este documento no es lograr ser un experto en el sistema "X", pero justo es mencionar en terminos generales de que se trata:

Es un sistema de ventanas que se maneja de forma transparente en una red, esto significa que el usuario puede acceder programas de aplicación que están corriendo en otra u otras máquinas localizadas físicamente en otros lugares pero que forman parte de la red como si la aplicación estuviera corriendo localmente. Con el "X", pueden correrse múltiples aplicaciones simultáneamente en varias ventanas siendo independientes de los dispositivos.

Las aplicaciones funcionan como clientes y el programa "X" que responde a las aplicaciones es conocido como el "Servidor X". Cada "Servidor X" controla una o más pantallas, así como un dispositivo apuntador, por ejemplo un ratón (mouse). "X" proporciona facilidades para generar texto

---

<sup>3</sup> NITS.- National Institute of Standards and Technology (Antes NBS.- National Bureau of Standards)

con múltiples tipos de letras y gráficas bidimensionales en una jerarquía de ventanas rectangulares.

El protocolo de comunicaciones X es la base fundamental de "X Window".

El protocolo es principalmente asíncrono y los datos pueden ser transmitidos en ambas direcciones simultáneamente. La operación asíncrona es una de las diferencias más significativas entre "X" y otros sistemas de ventanas.

Aunque "X" está definido fundamentalmente por un protocolo de red, incluye una biblioteca que puede considerarse la interface que enmascara el detalle de la codificación del protocolo y de las interacciones de transporte. La biblioteca también proporciona varias funciones de utilería que no están directamente relacionadas con el protocolo pero que son importantes al momento de la construcción de aplicaciones. La biblioteca "Xlib" es la biblioteca que contiene el lenguaje de programación "C", contiene más de 300 funciones y genera requerimientos del protocolo de "X" para el servidor.

El protocolo de red "X" y la Xlib pueden ser considerados como el conjunto que proporciona los mecanismos para implementar una gran variedad de políticas para las interfaces de usuario. Utilerías, bibliotecas para gráficas de alto nivel y el sistema de administración de interface, pueden todos ser implementados sobre la Xlib.

La interface entre el usuario y la plataforma de aplicación está construida en muchos sistemas de ventanas, pero en el caso de "X" no es así. El protocolo de "X" no define esta interface en su totalidad, en su lugar, proporciona los mecanismos con los cuales una gran variedad de interfaces pueden ser construidas. Estos mecanismos son diseñados de tal forma que un solo cliente llamado un manejador de ventanas puede proporcionar una interface independiente de todos los otros clientes.

Un administrador de ventanas "Window Manager" puede automáticamente proporcionar barras con títulos, bordes, otras decoraciones de ventanas para cada una de las aplicaciones, un medio uniforme para mover y quitar ventanas, iconos uniformes para las aplicaciones, una interface uniforme para cambiar el teclado entre aplicaciones y puede reforzar una política estricta de ventanas si así lo desea.

El "X Toolkit" es un juego de intrínsecos usados para crear "widgets" y así construir una interface de usuario, facilita la escritura de aplicaciones para el "X Windows". Contiene ejemplos de implementaciones "widget".

El "X Toolkit" es un ejemplo de arquitectura de software en capas. Una aplicación consiste de varios módulos de código distintos, comunicándose uno con otro a través de interfaces bien definidas.

Widgets son componentes de la interface de usuario, tales como un menú, scrollbar (barra de desplazamiento, vertical u horizontal), un campo para meter texto, una etiqueta o un botón.

Los intrínsecos son módulos de software que definen la estructura de "widgets", proporcionan el medio necesario para ensamblar "widgets" en una interface de usuario y despachar eventos de entrada en los "widget" correctos.

El "X Toolkit" permite que una aplicación tome widgets y los ponga en una interface de usuario, así mismo, incluye el soporte para la definición de nuevos "widgets".

---

<sup>44</sup> "X Toolkit" es un sistema de programación orientado a objetos

En la siguiente figura se esquematiza el "X Window System".

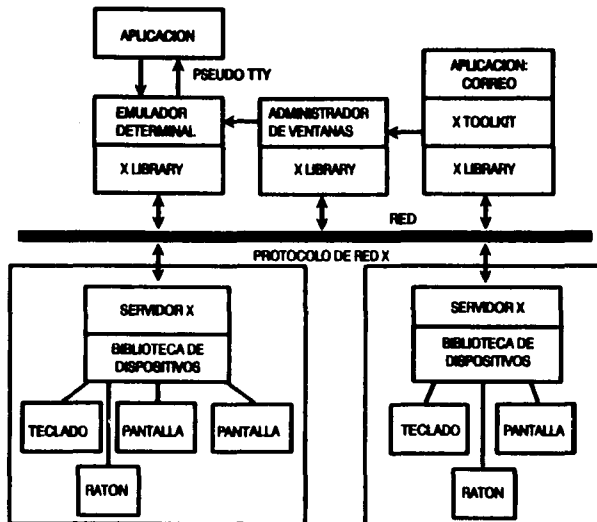


FIGURA NUM. 17.- El "X Window System"

## **OSF/MOTIF**

Una de las interfaces gráficas de usuario basadas en el "X" es la denominada "MOTIF" respaldada por la Open Software Foundation, quien la desarrolló y liberó en 1989. "Motif" es proporcionada por OSF para vendedores de hardware, desarrolladores de software y organizaciones de usuario final. Incluye un grupo de herramientas para interface de usuario compuestas de objetos GUI (Widgets y Gadgets) construidos sobre los intrínsecos X11 del "X Window System"; un lenguaje de interface de usuario es usado para describir los aspectos visuales de objetos tales como menus, etiquetas y formas usados en la interface y para funciones específicas a ser llamadas cuando la interface cambie como resultado de las interacciones con los usuarios; y el "OSF/Motif Window Manager", el cual permite a los usuarios configurar la apariencia y comportamiento de la interface.

## **AT&T-SUN/OPEN LOOK**

La interface gráfica de usuario "Open Look, fué desarrollada en forma conjunta por Sun Microsystems y AT&T.

Como con Motif, la interface Open Look es independiente de cualquier implementación particular.

El último release de UNIX de la USL para el mercado de workstations basadas en procesador Intel, llamado UnixWare, soporta ambos estándares: Motif y OpenLook. Este release está siendo producido por "UNIVEL", que es la compañía en la que se asociaron USL y Novell.

Diferentes versiones de UNIX, aún aquellas que comparte la interface Motif, pueden usar diferentes iconos (representación gráfica de objetos) para representar tipos de archivos, tales como archivos de datos y de programas.

## **POSIX 1003.2: EL SHELL Y SERVICIOS DE UTILERIAS-POSIX 1003.2**

A este respecto, el grupo de trabajo POSIX 1003.2 de la IEEE, también tiene que ver con la interface del usuario, cubriendo dos áreas amplias: El lenguaje de comandos "Shell" usado por los programas de aplicaciones, así como las características adicionales y utilerías que promueven el uso de la portabilidad de los sistemas por los usuarios.

## **DESCRIPCION DE LOS ESTANDARES POSIX 1003.2**



El estándar de aplicación de POSIX 1003.2 se funda en la definición del lenguaje de comandos "Shell" y basado principalmente en el "Bourne Shell" con algunas mejoras del "Korn Shell".

La siguiente figura ilustra la relación de varios componentes del trabajo de POSIX, específicamente aquel incluido en el POSIX 1003.2.

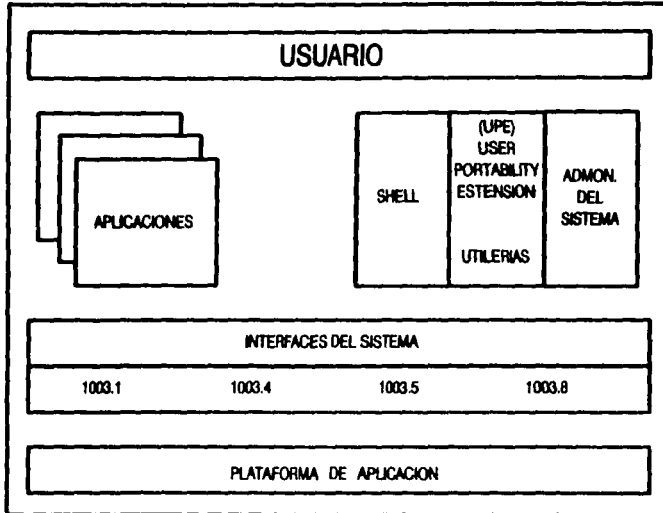


FIGURA NUM. 18.- COMPONENTES DEL ESTANDAR POSIX 1003.2

El lenguaje de comandos puede ser accesado vía funciones de programas en "C" o alojados en un archivo "script" en "shell", ejemplo:

```
system()
popen()
```

Usando lenguaje "shell" o por medio de llamadas con "exec()", una aplicación puede acceder 80 diferentes utilerías. Las utilerías están agrupadas en un grupo obligatorio de 72 y aquellas para desarrollo de software en general en lenguaje C y en Fortran.

La descripción de utilerías en POSIX 1003.2 tiene algunas características comunes:

- Están descritas en detalle, de modo que debería de estar la información adecuada para escribir aplicaciones estrictamente normalizadas sin tener que experimentar en línea o leer código fuente.
- Se ha hecho un considerable esfuerzo para definir un marco de referencia para la internacionalización de las aplicaciones. POSIX 1003.2 ha proporcionado la definición local extendible, herramientas de construcción locales, archivos de descripción de conjunto de caracteres y notaciones, expresiones regulares internacionalizadas y chequeo de patrones de archivos.
- Han sido eliminadas las dependencias arquitectónicas y de sistema.
- No están incluidas características o utilerías para administración.
- Una serie de lineamientos para la sintaxis de utilerías que dan una vista consistente al usuario, a través de la especificación de sus opciones y operandos.
- Se están desarrollando pruebas para todas las utilerías en un esfuerzo conjunto con el grupo de trabajo POSIX 1003.3.

En adición a las utilerías, POSIX 1003.2 contiene 15 funciones en lenguaje C que accesan el shell y proporcionan servicios comunes tales como opciones de "parsing" (análisis), expansión de palabras shell y expresiones regulares.

- POSIX 1003.2a: USER PORTABILITY EXTENSION (UPE).

El grupo de trabajo POSIX 1003.2 de la IEEE, también está trabajando sobre una extensión para la portabilidad de usuarios (USER PORTABILITY EXTENSION). Este estándar contiene características y utilerías en adición a aquellas definidas por POSIX 1003.2 que promueven el uso portátil de los sistemas por los usuarios.

La "USER PORTABILITY UTILITIES OPTION" en el POSIX 1003.2A adiciona 37 utilerías para el conjunto base definido por el POSIX 1003.2. Este soporta terminales de usuario de manera consistente a través de todos los sistemas que adoptan este estándar. La opción está limitada a dos restricciones:

- Los usuarios en este contexto están limitados a aquel grupo de individuos quienes ya con anterioridad se han familiarizado con el estilo de la interacción que se lleva a cabo con algunos de los sistemas operativos UNIX. Los típicos usuarios deberán ser desarrolladores de programas, ingenieros o usuarios de tiempo compartido y de propósito general.

- El ambiente es un sistema de tiempo compartido multi-usuario que soporte terminales de video orientadas a caracter. Los estándares no incluyen soporte orientado a terminales de mapas de bits o gráficas excepto cuando tales terminales emulan la orientación de caracter requerida para este ambiente.

### **FORMAS ELECTRONICAS: "FIMS"**

La manipulación de formas electrónicas tienen en gran parte que ver con la interacción que muchos usuarios tienen con las aplicaciones de la computadora y los sistemas.

Una forma electrónica es un grupo de campos y texto en un display (un video). Las formas son usadas con todo tipo de dispositivos así como con una variedad de aplicaciones para mejorar la captura y despliegado de información.

El estándar para el Sistema de Administración de Interface para formas (FIMS.- FORM INTERFACE MANAGEMENT SYSTEM) está siendo propuesto por dos organismos: ANSI e ISO. Este trabajo estandarizará la interface entre una aplicación y las formas que la usan, de tal manera que los usuarios tendrán consistencia sin importar la aplicación que usa las formas. La interface de programación de aplicación que proporciona FIMS ofrece independencia de dispositivos, independencia de lenguaje, facilidad de uso, así como flexibilidad y control.

### **DESCRIPCION DE FIMS**

La "USER PORTABILITY UTILITIES OPTION" en el POSIX 1003.2A adiciona 37 utilerías para el conjunto base definido por el POSIX 1003.2. Este soporta terminales de usuario de manera consistente a través de todos los sistemas que adoptan este estándar. La opción está limitada a dos restricciones:

- Los usuarios en este contexto están limitados a aquel grupo de individuos quienes ya con anterioridad se han familiarizado con el estilo de la interacción que se lleva a cabo con algunos de los sistemas operativos UNIX. Los típicos usuarios deberán ser desarrolladores de programas, ingenieros o usuarios de tiempo compartido y de propósito general.

- El ambiente es un sistema de tiempo compartido multi-usuario que soporte terminales de video orientadas a caracter. Los estándares no incluyen soporte orientado a terminales de mapas de bits o gráficas excepto cuando tales terminales emulan la orientación de caracter requerida para este ambiente.

### **FORMAS ELECTRONICAS: "FIMS"**

La manipulación de formas electrónicas tienen en gran parte que ver con la interacción que muchos usuarios tienen con las aplicaciones de la computadora y los sistemas.

Una forma electrónica es un grupo de campos y texto en un display (un video). Las formas son usadas con todo tipo de dispositivos así como con una variedad de aplicaciones para mejorar la captura y despliegado de información.

El estándar para el Sistema de Administración de Interface para formas (FIMS.- FORM INTERFACE MANAGEMENT SYSTEM) está siendo propuesto por dos organismos: ANSI e ISO. Este trabajo estandarizará la interface entre una aplicación y las formas que la usan, de tal manera que los usuarios tendrán consistencia sin importar la aplicación que usa las formas. La interface de programación de aplicación que proporciona FIMS ofrece independencia de dispositivos, independencia de lenguaje, facilidad de uso, así como flexibilidad y control.

### **DESCRIPCION DE FIMS**

FIMS define un modelo simple de "RUN TIME" que corresponde al comportamiento normal de una forma electrónica. FIMS especifica los elementos relacionados a la apariencia así como a los elementos de datos y procedurales tales como verificación de campos y opciones con iconos o palabras clave que controlan la interacción del usuario.

Las formas son descritas por el "INDEPENDENT FORM DESCRIPTION LANGUAGE" (IFDL), así llamado porque es independiente de cualquier lenguaje de programación o sistema. IFDL proporciona una definición estándar de las características de interface de la forma y un lenguaje fuente para intercambio de formas entre las organizaciones. El nivel de control de campo, navegación de pantalla, administración de pánel y validaciones en los campos son ejemplos de las actividades realizadas por el IFDL.

FIMS permite que una sola aplicación soporte múltiples tipos de interfaces de usuario utilizando una estructura llamada "layout". También soporta un rango muy amplio de dispositivos de display, basados en "character-cell" como en "windows".

Este estándar contempla ligas (bindings) hacia los lenguajes COBOL y C.

El estándar FIMS no especifica ninguna herramienta para desarrollo de formas. En su lugar especifica las características que un grupo de herramientas deben tener, tampoco define controles de proceso de transacciones, acceso a bases de datos o facilidades de lenguajes de propósito general. Es un estándar complementario para los lenguajes y el servicio de éstos, un ejemplo son el COBOL y el SQL.

Lo que se espera es que un implementador proporcione un conjunto de herramientas que cumplan con FIMS y ayuden al diseñador de formas en la construcción de interfaces de usuario.

### **HISTORIA Y CONDICION ACTUAL**

El comité FIMS de CODASYL, reconocido por los organismos ISO y ANSI, comenzó a trabajar en 1979 en la definición de un modelo y un lenguaje que soportaran una interface estándar para formas electrónicas.

Los objetivos de este trabajo fueron que se tuvieran ventajas como:

- Independencia de dispositivos.
- Independencia de lenguaje.

Facilidad de uso.  
Flexibilidad y control.

En 1988 el Comité CODASYL comenzó a trabajar en un subconjunto de FIMS, que es más simple y fácil de instrumentar, pero que no contempla todas las capacidades ofrecidas por el estándar original FIMS

Como vimos hasta aquí, las interfaces de usuario son un elemento sumamente importante para el logro de los objetivos de los sistemas abiertos, pero es una lástima que no exista todavía un estándar formal en esta área, no obstante, las GUI'S más aceptadas en la industria son aquellas cuyo diseño está basado en el "X", que es considerado ya el estándar de facto para los sistemas de ventanas en ambientes distribuidos.

### **2.3.3 S.- SERVICE INTERFACES FOR PROGRAMS (INTERFACE DE SERVICIOS PARA PROGRAMAS)**

El elemento servicios del sistema, incluye programas que son típicamente parte del sistema operativo: interfaces para aplicaciones y programas del sistema, así como servicios de la plataforma de aplicación.

Incluye especificaciones de los lenguajes, ligas hacia los mismos y las interfaces de programación de aplicación (API's) para gráficas y otros servicios.

Los componentes de este elemento afectan también la portabilidad de los sistemas y de los mismos programadores.

### **LENGUAJES**

Los lenguajes de computadoras proporcionan el medio de comunicación más inmediata entre un desarrollador y las interfaces del programa. Un lenguaje establece un paradigma para la descripción de una aplicación, tanto en términos de datos como de su función.

Los lenguajes deben tener ciertas características: describir datos y el rango de datos que puede ser manipulado. La estructura de esta descripción crea una diferencia significativa entre lenguajes. Como resultado de éstas diferencias los lenguajes tienden a ser asociados con tipos específicos de aplicaciones, por ejemplo:

Fortran para aplicaciones ingenieriles.

**Cobol para aplicaciones administrativas.**

**Otras funciones definidas por lenguajes incluyen:**

- Control de flujo en la aplicación (ciclos, goto).
- Control de decisión (if, case).
- Manipulación de datos (cálculos aritméticos, manejo de cadenas).
- Operaciones de entrada/salida para dispositivos de memoria y de despliegue a usuarios.

En las capacidades de entrada/salida se encuentra otra divergencia entre los lenguajes, por ejemplo, COBOL y PL/I ambos incluyen método de acceso secuencial indexado para manejo de archivos (ISAM), pero no existen estándares para esto con otros lenguajes.

Por otro lado, ya existen estándares tales como SQL para acceso de bases de datos.

Los lenguajes también varían en cuanto a la capacidad de interacción con el usuario, están limitados en interacción estructurada y la independencia de los dispositivos. Estándares independientes tales como el "Form Interface Management System" (FIMS), consideran elementos adicionales para la interacción con el usuario. Ambos SQL y FIMS, son lenguajes en sí, dando algo de independencia y variación en el paradigma de la programación.

Para obtener la portabilidad tan prometida en los sistemas abiertos, las aplicaciones deben ser desarrolladas utilizando lenguajes y otras herramientas de desarrollo, que también sean soportadas por estándares formales. Los lenguajes populares que han obtenido su estandarización a nivel internacional se muestran en la siguiente tabla:

Lenguaje	Estándar Fuente Referenciado	Derivación	Uso	Tipo de Datos Significativos	Características clave
ADA	ISO 6652 U.S. DoD 1970	Pascal, PL/I, Algol 68	Proceso de Tiempo Real	Crea su propio tipo de datos	Multitarea Concurrente, Modular
BASIC	ISO 6373 Kerny and Kurtz, 1980's		Estudiantes	Enteros (Byte, Palsbra), Punto Flotante (Sencillo, doble, G-Flotante, H-Flotante); String (le 65535 caracteres); Decimal Empacado	Interprete, no compilador, capacidades de generación de sonidos y gráficas.
C	ISO 9899 Bell Labs, 1970	B y BCPL	UNIX, Alternativa de lenguaje Ensamblador para programas de microcomputadoras, programación estructurada.	Entero, punto flotante, doble precisión, estructura de carácter, apuntadores.	Permite acceso a nivel de máquina, orientado a programación permite tipos de datos definidos por el programador así como registros.
COBOL 85	ISO 1889 U.S. DoD, Varios vendedores de Sistemas 1980's		Aplicaciones de tipo administrativo.	Decimales	Usa estructura de datos de registro fácil de leer.
FORTRAN 77	ISO 1339 IBM, 1950's		Científico Ingeniería	Enteros, punto flotante, booleano.	Permite al programador describir cálculos en términos de fórmulas.
Pascal	ISO 7185 Niklaus Wirth		Popular para microcomputadoras, programación modular.	Reales, enteros booleanos, carácter	Nuevo tipo de datos definidos por el programador.
PL/I	ISO 6180 IBM, 1960's	Algol 60	Programas complejos en mainframes IBM	Muchos tipos de datos, incluyendo punto flotante, binarios fijos, decimales fijos, cadenas de carácter, cadenas de bits	Gran nivel de independencia de máquina implementación eficiente máquinas de todos tamaños.

**FIGURA NUM. 19.- LENGUAJES ESTANDARIZADOS INTERNACIONALMENTE**

ESTA TESIS NO DEBE SALIR DE LA BIBLIOTECA



Lenguajes como C++, Common LISP y Prolog, no cuentan con estandarización internacional de momento.

Para que una aplicación sea portable, las relaciones y dependencias de los lenguajes con respecto a otros estándares también debe estar definida. Estas relaciones están estandarizadas y definidas y son conocidas como "binding standards", la tabla siguiente, resume las ligas de los lenguajes que existen para lenguajes estandarizados (ISO) comunmente implementados:

		BASE DE DATOS				GRAFICAS	
Lenguaje	POSIX DIS9945	SQL 9075	SQL2 (ip)	GKS 7942	GKS-3D DIS8805	PHIGS DIS9592	CGI DP9636-1 ■ 6 DP9637-1 ■ 7
Ade 8652	P1003.5		(ip)	DIS8551-1	DIS8806-3	9593-3	DP9636-3 DP9639-3
BASIC 8373							
CDP9699	1003.1		(ip)	8651-4	8806-4	9593-4	DP9636-4 DP9639-4
COBOL		9075	(ip)				
FORTRAN 1359	1003.9	9075	(ip)	8651-1	DP8806-1	DIS9593-	DP9636-1 DP9639-1
Pascal 7185		(ip)		8651-2	DP8806-2	9593-2	DP9639-2
PL/16180		9075	(ip)				

ip.- In progress (No concluidos)

FIGURA NUM. 20.- ESTANDARES RELACIONADOS A LENGUAJES

## **POSIX.- Portable Operating System Interface**

Los fabricantes han producido diversos sistemas operativos UNIX y con el objeto de estandarizarlo un grupo llamado /USR/GROUP empezó en 1981 a encargarse de ésto. Se ha llegado a una especificación del sistema operativo conocida como **POSIX: PORTABLE OPERATING SYSTEM INTERFACE FOR COMPUTER ENVIRONMENT.**

El objetivo de POSIX es soportar la portabilidad, a través de diferentes máquinas, del código fuente de aplicaciones que cumplan con POSIX. Cualquier variante de UNIX o aún otro sistema operativo, puede ser bajo la interface POSIX. Conforme los sistemas operativos soporten la interface POSIX, las aplicaciones deberán correr debido a que ésto es todo lo que ellas ven: **LA INTERFACE.** El programador interactúa solamente con la interface POSIX y, en teoría, no necesita aprender nuevas interfaces de programación de sistema operativo para diferentes máquinas.

Sin embargo, el resultado de la portabilidad del código fuente es difícil bajo POSIX porque incluye algunas implementaciones opcionales de un número de llamadas del sistema, y porque no incluye muchas de las funciones necesarias para el desarrollo de sistemas.

A pesar de sus limitaciones, la especificación POSIX es ampliamente soportada en varias implementaciones de UNIX y ha sido prometida para un número diverso de sistemas operativos propietarios, incluyendo HP MPE/IX (Anteriormente MPE XL), DEC VMS, y OS/2.

Entre los servicios que realiza un sistema operativo, podemos citar aquellos ofrecidos a los programas de aplicación, tales como: lecturas/escrituras a disco, creación de procesos, comunicación entre programas, administración de archivos. Si observamos en detalle "como son solicitados" estos servicios a los diferentes sistemas operativos disponibles en el mercado, llegaremos a la conclusión de que, en cada uno de ellos es diferente, dado que la interfase programa-sistema operativo, es distinta en cada caso. Esta incompatibilidad se traduce en que los programas de aplicación no son portables entre los diferentes sistemas operativos.

En 1985 la IEEE, por medio del Comité Técnico sobre Sistemas Operativos (TCOS.- Technical Committee on Operating Systems) siguió este trabajo de estandarización, formando el comité P1003 para concretar dicho esfuerzo. Este trabajo está enfocado a la definición de un ambiente de sistemas abiertos conocido como OSE.- Open Systems Environment.

Los objetivos del comité POSIX son el de especificar las funciones y servicios que debe soportar un sistema operativo (ambiente operativo), así como las interfaces de programación de las aplicaciones a estos servicios. En la mayoría del trabajo que se viene realizando tradicionalmente sobre estándares, los estándares individuales son relativamente independientes uno del otro. En el caso de POSIX se trata de un proyecto que involucra a un gran grupo de estándares interrelacionados.

El "OSE" de POSIX, es un modelo conceptual que proporciona un contexto para especificación de estándares y requerimientos de usuario. Proporciona un conjunto estándar mínimo de elementos conceptuales de construcción de sistemas de información con interfaces y funcionalidad asociados.

Los estándares POSIX están numerados por la IEEE. Cada estándar tiene asignado un grupo de trabajo que está encargado de definir los requerimientos para determinada área.

Algunos grupos de trabajo están laborando en más de un estándar o área de estándares. Cada estándar tiene un subconjunto de estándares relacionados y todos ellos forman parte de la totalidad de POSIX.

El comité P1003, POSIX, funciona dividido en 12 subcomités, los cuales están trabajando en la definición de estándares de industria.

Hay tres grandes áreas que están siendo manejadas por los grupos de trabajo de POSIX.

La primera es la definición de los servicios y características necesarias por el "OSE" para la portabilidad de las aplicaciones.

La segunda se refiere a las ligas para aquellos servicios de los lenguajes específicos.

La tercera está dedicada a temas tales como seguridad que se divide a través de grupos de servicios o de la definición de perfiles para dominios específicos de aplicaciones tales como supercómputo.

Los grupos de trabajo POSIX han identificado los servicios requeridos que se consideran son necesarios para la portabilidad de las aplicaciones así como para la interoperabilidad de los sistemas, estos servicios incluyen:

- Administración de procesos.

- Administración de tareas.
- Ambiente.
- Sincronización y comunicación interna de nodos.
- Generalizados de entrada/salida.
- Orientados a archivos.
- Administración de eventos, error y excepción.
- Administración de tiempo.
- Administración de memoria.
- Nombres lógicos.
- Inicialización del sistema.
- Reinicialización y shutdown.
- Lenguajes de programación.
- Servicios generales del procesador y administración de los componentes del sistema.

A continuación una breve explicación del contenido de cada una de las áreas en que trabaja POSIX:

**1003.0**      Guía para el "POSIX Open Systems Environments".

Esta guía es un resumen del trabajo de los otros grupos, está pensado como un esbozo general de los estándares POSIX, contiene los objetivos y puede utilizarse como una guía para los desarrolladores de perfiles.

**1003.1**      Interfaces de servicios del sistema POSIX y ligas en el lenguaje C.

Define las interfaces entre programas de aplicación portables y el sistema operativo. Describe como pueden ser utilizados por aplicaciones POSIX los

servicios del sistema. Este trabajo fué el área original de estándares POSIX y está basado en modelos históricos del sistema UNIX y el lenguaje C.

**1003.2 Shell POSIX y servicios de utilerías.**

Especifica un lenguaje de comandos shell basado en el Bourne Shell de UNIX con algunas características basadas en el Korn shell. Proporciona algunas funciones para acceder los servicios del shell desde aplicaciones y cerca de 80 utilerías a ser llamadas desde archivos de shell (shell script) o directamente desde aplicaciones.

**1003.3 Métodos generales de prueba para POSIX.**

Este estandar define las especificaciones de métodos de prueba para probar conformidad de un producto a las normas POSIX.

**1003.4 Servicios de aplicación (tiempo real).**

**1003.5 Ligas del lenguaje Ada para POSIX.**

Define las ligas para el lenguaje Ada para servicios básicos del sistema definidos por POSIX 1003.1.

**1003.6 Las extensiones de seguridad para POSIX.**

Este juego de estándares definirá el ambiente de seguridad para el "POSIX OSE". El trabajo está basado sobre criterios para interfaces de un sistema publicado en el Departamento de Defensa de los Estados Unidos: "Trusted Computer System Evaluation Criteria".

El grupo de trabajo está desarrollando interfaces en 4 áreas: control discrecional de acceso, control de acceso obligatorio, auditoría y privilegios.

**1003.7 Extensiones de administración del sistema.**

Proporcionará estándares para una interface común para la administración de sistemas, particularmente redes de múltiples vendedores en "POSIX OSE". El modelo de administración OSI está siendo usado como punto de partida para este trabajo.

**1003.8 Acceso transparente a los archivos.**

Este grupo de trabajo proporcionará estándares para interface del sistema de archivos y semántica para acceso transparente de archivos en un ambiente de archivos distribuidos. El comité está examinando mecanismos de acceso ya existentes de archivos remotos tales como NFS, RFS, AFS Y FTAM.

**1003.9      Ligas con lenguaje Fortran.**

Define las ligas del lenguaje Fortran para servicios básicos del sistema.

**1003.10      Perfil de ambiente de aplicaciones de supercomputo-  
(AEP.- Application Environment Profile).**

Definición de un perfil para soportar aplicaciones y portabilidad del programador en ambientes desupercómputo basados en POSIX.

**1003.11      Perfil de ambiente de aplicación de transacciones on-line.**

La responsabilidad de este grupo es producir un AEP para OLTP (On-line transaction processing).

**1003.12      Interfaces independientes de protocolo para comunicaciones.**

Este grupo de trabajo está laborando sobre dos interfaces estándar de red para comunicaciones proceso a proceso independiente de protocolo:

DNI.- Detailed Network Interface.

SNI.- Simple Network Interface.

El trabajo comprenderá TCP/IP, OSI y otros sistemas de comunicaciones.

**1003.13      Perfil de ambiente de aplicación de tiempo real.**

Producir un AEP para cómputo de tiempo real.

**1003.14      Perfil de ambiente de aplicación de multiproceso.**

Producir un AEP POSIX para multiproceso y proponer extensiones específicas para basar estándares de otros grupos.

**1003.15      Extensiones de colas de lotes (batch).**

**Este grupo de trabajo está definiendo una interface de usuario, una interface de administrador y un protocolo para manejo de colas batch.**

**1003.16 Uniones al lenguaje C.**

**El trabajo de este proyecto es desarrollar una unión al lenguaje C (ISO 9989) para API independiente de lenguaje que corresponde exactamente al POSIX 1003.1-1990 (ISO 99445-1).**

**Esta labor la está realizando el grupo de trabajo 1003.1.**

**1003.17 Servicios de directorio y espacio de nombres.**

**El cargo de este grupo de trabajo es examinar el espacio del nombre y las implicaciones de servicios de directorio en un contexto X.500 que se relaciona con sistemas POSIX distribuidos.**

**1003.18 Perfil de Ambiente de Aplicación de la plataforma POSIX.**

**Este proyecto está trabajando para establecer un AEP basado en el trabajo POSIX que describe una base simple para una plataforma de aplicación multiusuario e interactiva.**

**El grupo de trabajo 1003.1 tiene a su cargo esta responsabilidad.**

**De esta manera POSIX se convierte en un elemento fundamental en el mundo de los estándares de la industria de cómputo. De hecho, X/Open lo reconoce y promueve como un elemento base de su ambiente de Sistemas Abiertos, razón por la cual la OSF y UI lo han tomado como punto de referencia en sus implementaciones de interfase de sistema operativo.**

## **GRAFICAS**

**El trabajo sobre estándares relacionados con los lenguajes, intercambio de datos y las areas de interface de usuario, algunas veces se sobreponen con el trabajo de estandarización de gráficas se viene haciendo. Por ejemplo, ISO está trabajando sobre las ligas a los lenguajes para SQL, SQL2, GKS-3D, PHIGS, AND CGI.**

## **GKS**

El Graphical Kernel System (GKS, ISO 7942-1985) es un sistema básico de gráficas para aplicaciones que produce gráficas de 2 dimensiones. Soporta entradas del operador e interacción por medio de funciones básicas para entrada de gráficas y segmentación de dibujos. Proporciona al programador la habilidad de crear gráficas sobre una amplia variedad de dispositivos para gráfica. Esto incluye monitores blanco y negro, impresoras, plotters y sistemas de cámara. Un mouse, joysticks, tablas de datos y digitalizadores pueden también ser dispositivos de entrada.

A fin de permitir que aplicaciones particulares elijan un paquete de gráficas con la capacidad apropiada, GKS fué definido con diferentes niveles. Una estructura de 3 niveles es definida tanto para entrada como salida. El nivel de entrada más alto frecuentemente no es soportado debido a que requiere facilidades asíncronas de entrada que no todos los sistemas operativos ofrecen.

GKS le proporciona un juego de funciones al programador para la administración de imágenes gráficas. Proporciona un ambiente para controlar dispositivos de display y de entrada, así como para almacenar datos de gráficas y registrar el progreso de una sesión de proceso particular. El estándar es un lenguaje independiente pero el programador debe configurar una aplicación usando este estándar para dispositivos de entrada y salida.

## **HISTORIA Y CONDICION ACTUAL**

GKS comenzó como un diseño propuesto para la ISO por DIN (German Institute for Standards, miembro de la ISO) a finales de los 70's. Fué editado, alterado y finalmente publicado como un estándar ISO en 1985. El objetivo original de GKS fué definir una interface común para gráficas de cómputo interactivo para programas de aplicación. La totalidad del estándar proporciona especificaciones funcionales para algunas 200 subrutinas las cuales realizan entradas y salidas de gráficas.

Desde que se implementó el diseño básico de GKS, las gráficas en las computadoras han evolucionado dramáticamente, por ello, surgió una extensión tri-dimensional para GKS, el GKS-3D, fué estandarizado por ISO en 1988.

## **GKS-3D**



Este estándar, el ISO 8805-1988, proporciona un juego de funciones para definición y despliegue de datos de gráficas de 3 y 2 dimensiones, almacenamiento y manipulación de datos gráficos, así como la entrada de datos relacionados gráficamente. El estándar es un juego de extensiones del GKS-2D.

## **PHIGS**

Programmer's Hierarchical Interactive Graphics Standard (PHIGS, ISO 9592) es una especificación funcional de la interface entre una aplicación y subsistema de soporte de gráficas.

Phigs controla la definición, modificación y despliegado de datos gráficos jerárquicos. En adición especifica las descripciones funcionales de capacidades de los sistemas incluyendo la definición de estructura interna de datos, capacidades de edición, operaciones dedespliegado y funciones de control de dispositivos.

Phigs fué diseñado para que se utilice en programas de aplicación con las siguientes necesidades:

- Un alto grado de interactividad.
- Definición y despliegue de 3D así como primitivas de 2D.
- Estructura de datos multinivel/jerárquica.
- Rápida modificación de datos gráficos y las relaciones entre los datos.
- Articulación geométrica.

Los objetos son definidos en la base de datos gráfica de PHIGS por medio de una secuencia de elementos incluyendo primitivas de salida, atributos, transformaciones e invocaciones a otros objetos y definiciones de objetos pares. Estos elementos son agrupados en entidades llamadas estructuras. Las estructuras pueden estar relacionadas con el número de maneras geoméricamente, jerárquicamente o de acuerdo a propiedades inherentes o características definidas por la aplicación.

PHIGS permite un rápido y dinámico acceso a bases de datos centralizadas de gráficas. Esto permite a PHIGS soportar programas de aplicación

interactivos y, con la plataforma apropiada, definición de tiempo real y modificación de datos gráficos.

Una extensión de PHIGS llamada PHIGS PLUS le dió más funcionalidad al estandar original.

### **HISTORIA Y CONDICION ACTUAL**

El trabajo inicial sobre PHIGS fué realizado por los grupos de trabajo ANSI X3H3.1. Durante los ultimos pasos de su estandarización, el trabajo fué movido para el ISO SC24 WG2.

PHIGS llegó a ser un estándar nacional americano (American National Standard) y un estándar internacional en 1988.

Muchos de los conceptos de PHIGS son derivados del GKS y del GKS-3D, así que la curva de aprendizaje de lo programadores que ya tienen conocimientos del GKS es más corta.

La importancia de PHIGS para el mercado de las workstation es la motivación principal del desarrollo de PEX, que es una extensión de PHIGS para X.

#### **PEX**

El protocolo del "X Window System" tiene un nivel limitado de funcionalidades gráficas. Pero existe el estándar PHIGS y GKS que son mucho más funcionales. PEX es una extensión del "X Window System" para soportar gráficas 3D a través de PHIGS.

#### **2.3.4 1.-INFORMATION AND DATA FORMATS (INFORMACION Y FORMATO DE DATOS)**

Este elemento incluye los servicios necesarios para realizar el acceso e intercambio de datos, se puede subdividir en tres áreas principales:

- Definición y acceso de datos.
- Depósitos centrales de información (Repository).

- **Formatos de intercambio de información.**

Los lenguajes orientados a objetos tienen que ver con este elemento.

### **DEFINICION Y ACCESO DE DATOS**

Se han hecho esfuerzos a nivel internacional (ISO/IEC JTC1/SC 21/WG3) para desarrollar y mantener un modelo de referencia para la estandarización de base de datos. El modelo define un marco de referencia para:

- Coordinar el desarrollo de actuales y futuros estándares para la administración de datos.
- Identificación de interfaces.
- Posicionamiento de las interfaces relativas una de otra.
- Identificación de facilidades y procesos que soportan cada intrface.
- Identificación de conexiones alternativas asociadas con cada interface.
- Definir terminología común y conceptos pertinentes a todos los datos manejados en sistemas de información.

Este modelo todavía está a nivel ISO DIS de estandarización (DIS.- Draft International Standard, es un paso en la metodología de estandarización de la ISO).

El acceso a base de datos es crítico cuando se trata de numerosas aplicaciones. Han existido por cerca de 30 años, modelos sofisticados y métodos para la organización y acceso a los datos, por lo que algunos estándares tales como lenguajes de base de datos son bastante maduros. Otras áreas están evolucionando y el trabajo de estándares está en su fase inicial.

### **ESTANDARES PARA LENGUAJES DE BASE DE DATOS**

El propósito de estos estándares es proporcionar la portabilidad de definición y programas de aplicación de base de datos entre implementaciones estandarizadas. El SQL (Structured Query Language) y el NDL (Network Database Language) son lenguajes de base de datos que han sido estandarizados a nivel internacional.

## **SQL (ISO 9075-1989, ANSI X3.135-89)**

Este estándar es apropiado para todas las aplicaciones de base de datos donde los datos serán compartidos con otras aplicaciones, donde el ciclo de vida de la aplicación es mayor que la vida del equipo en el que inicialmente se creó la aplicación, o donde la aplicación va a tener que ser comprendida y mantenida por programadores diferentes a los que originalmente la implementaron. Este estándar es particularmente apropiado para aplicaciones de base de datos que requieren de flexibilidad en la estructura de datos y en las trayectorias de acceso a la base de datos. Es adecuado tanto para aplicaciones en plena producción como para aquellas que todavía necesitan bastante manipulación.

SQL especifica la definición y manipulación de datos, chequeo de integridad y otras facilidades asociadas al modelo relacional.

Adicionalmente, el estándar SQL especifica componentes que soportan:

- Control de acceso.
- Interface a lenguajes de programación.
- Administración de datos.

Proporciona facilidades para los lenguajes para requerimientos de aplicaciones específicas en cuanto a la vistas de los datos.

SQL cuenta con un lenguaje para interface hacia otros lenguajes. Cada estatuto SQL puede ser empacado como un procedimiento que puede ser llamado y puede tener parámetros pasados a él desde un lenguaje externo. Una especificación SQL (ANSI X3.168) incorporada (embedded) proporciona una interface SQL hacia lenguajes de programación, específicamente: Ada, C, COBOL, Fortran, Pascal y PL/I.

SQL fué originalmente desarrollado como un lenguaje prototipo para el proyecto "systemVR" en IBM.

El primer estándar de SQL se publicó en 1986.

Se comenzó la revisión de dicho estándar para adicionar más capacidades y esta revisión fué publicada en 1989. SQL-89, que es el estándar actual, es especificado para un ambiente único de base de datos.

Las especificaciones para acceso a sites heterogeneos remotos está bajo el desarrollo de un reciente estándar de la ISO, el Remote Database Access (RDA).

Una mejora al estándar SQL (SQL2), está actualmente en revisión pública en los Estados Unidos en condición de DIS (Draft International Standard).

EWOS (European Workshop for Open Systems -OSI Functional Standards) está trabajando también en esta área. En general, la existencia de implementaciones del actual estandar SQL tendrá que ser extendida significativamente para soportar SQL2.

El SQL-86 y el SQL-89 son idénticos para las versiones ANSI e ISO. El nuevo estándar (SQL2 o SQL-9x) también se espera sean idénticos para ambas versiones ISO y ANSI.

Una segunda mejora para SQL frecuentemente llamada SQL3, está bajo desarrollo por la ANSI y los comités ISO SQL la cual se espera se publique después de 1995. Las mejoras de SQL están destinadas a proporcionar facilidades adicionales para el manejo de datos orientados a objetos y para formar la base de sistemas de administración de base de datos "Inteligentes".

El lenguaje SQL es el estándar de facto y de jure para interfaces de sistemas de base de datos relacionales. Las implementaciones pueden estar en cualquier lado y están disponibles en todos tamaños y tipos de equipo.

#### **NDL.- NETWORK DATABASE LANGUAGE (ANSI X3.133-86, ISO 8907-87)**

El estándar NDL es apropiado para aplicaciones de base de datos altamente estructuradas y estáticas que requieren de acceso rápido a lo largo de trayectorias predefinidas, especifica sintaxis y semántica para la definición y manipulación de datos, restricciones de integridad muy limitadas y otras facilidades, todas ellas asociadas con el modelo de datos de red.

El estandar NDL está reconocido por organismos internacionales de estandares como:

ANSI: X3.133-86.  
OSI 8907-87.

En virtud de que las tendencias en la utilización de base de datos indican que el modelo de red está siendo reemplazado por el relacional y últimamente también por el orientado a objetos, este estándar solo lo menciono para que nos quede claro que en este ámbito también existe el deseo de obtener más flexibilidad en la portabilidad de aplicaciones, de ahí el interés por los estándares en esta área.

### **RDA.- REMOTE DATA BASE ACCESS**

Este estándar puede ser utilizado para establecer una conexión remota entre un cliente de base de datos actuando en favor de una aplicación, y un servidor de base de datos interconectado para un proceso que controla la transferencia de datos hacia y desde una base de datos. El objetivo es promover la interconexión de aplicaciones de base de datos entre ambientes heterogéneos.

Este estándar consta de dos partes propuestas, el genérico RDA para conexión a base de datos arbitraria y una especialización SQL para conexión de bases de datos normalizadas a SQL. RDA es una especificación para servicios y protocolo para aplicaciones en ambientes de base de datos, no una especificación de base de datos distribuida.

El estándar genérico propuesto RDA, da una interface de servicio RDA y un elemento de comunicación RDA que existe tanto en el cliente como en el servidor. El servicio genérico no especifica la sintaxis o semántica de operaciones de base de datos enviadas desde un cliente a un servidor, en su lugar, el estándar asume la existencia de un lenguaje especializado que especifica la sintaxis de transferencia exacta para operaciones estándar.

La especialización RDA/SQL complementa a la parte genérica RDA para usarse cuando está presente un manejador de datos SQL en la parte del servidor. El lugar del cliente puede también tener un manejador de datos conforme a SQL, pero no es requerido.

Muchos vendedores de SQL planean tener clientes y servidores disponibles que adopten este estándar. Vendedores tales como SQL Access y X/Open esperan tener prototipos operacionales para demostrar la interoperabilidad entre diferentes servidores SQL.

Un grupo de trajo de NIST está trabajando sobre RDA.

## **DEPOSITOS CENTRALES DE INFORMACION (REPSTORY)**

### **IRDS.- Information Resource Dictionary System.**

El IRDS es el estándar X3.138-1988 de ANSI. IRDS fue desarrollado originalmente como una definición estándar de diccionarios de requisitos (este estándar también puede utilizarse para depósitos). Se centra en la gestión de recursos de información corporativa y se caracteriza por su metamodelo multinivel. Este estándar ayuda a la creación de "puentes" entre herramientas complementarias, tales como las herramientas de análisis/diseño y los generadores de código, así como en la portabilidad de herramientas CASE entre diferentes plataformas.

### **INDEXED SEQUENTIAL ACCESS METHOD (ISAM)**

X/Open ha especificado interfaces para la creación, administración y manipulación de archivos indexados como parte de su "Common Applications Environment". Dichas especificaciones son conocidas como "Indexed Sequential Access Method" (ISAM). Las especificaciones ISAM proporcionan el acceso a sistemas de administración de base de datos relaciones vía SQL. Esta especificación es una implementación independiente.

### **FORMATOS DE INTERCAMBIO DE INFORMACION**

Proceso e intercambio de documentos:

En la actualidad también tenemos intercambio de documentos (categorizado dentro de intercambio de datos). El subcomité ISO/IEC está trabajando en la estandarización formal de esta área. El propósito de estos estándares es facilitar el intercambio de documentos.

Los documentos son elementos tales como: cartas, facturas, formas y reportes, los cuales pueden incluir a su vez, dibujos y material tabular, así como texto. Los elementos contenidos en los documentos pueden incluir caracteres gráficos, elementos gráficos geométricos, todos potencialmente en el mismo documento.

ODA.- The Office Document Architecture / ODIF.- Office Document Interchange Format.

**ODA/ODIF es un estándar múltiple ISO (ISO 8613) que incluye:**

- Secciones en estructuras de documento.
- Perfiles de documentos.
- ODIF.- Office document interchange format.
- CCA.- Character content architectures.
- RGCA.- Raster graphics content architectures.
- GGCA.- Geometric graphics content architectures.

**ISO 8613-2 Describe en detalle las estructuras del documento.**

**ISO 8613-4 define el perfil del documento que es siempre el primer elemento de la cadena de datos que se intercambia.**

**8613-5 La notación de sintaxis abstracta (ASN.1) de esta arquitectura (ODIF) se describe en esta sección.**

ODA no solo estandariza un medio para describir la estructura de los documentos, también proporciona un modelo para formatear un documento desde su estructura lógica específica, estructura genérica de su disposición (layout), y de los estilos de presentación y disposición. ODA está pensado para describir documentos que contengan diferentes tipos de contenido. Como la adición de nuevos tipos de contenido fué antes de que el estándar fuera desarrollado, ODA proporciona una interface generalizada para el contenido, la cual permite que la descripción completa del documento prosiga sin importar el tipo de contenido.

ISO 8613 actualmente tiene 3 partes que ven lo del contenido y cada una de estas partes especifica varios niveles de arquitectura o grados de procesabilidad para un tipo particular de contenido.

El ISO 8613-6, Tiene que ver con la inclusión de contenido de texto de caracteres basados en el juego estandar de caracteres ISO y un font estandar, el ISO 9541.

El ISO-8613-7 Se refiere a gráficas tipo "raster" y está basado sobre las recomendaciones de facsimile de la CCITT (T.4 y T.6).

El ISO 8613-8 soporta gráficas geométricas basadas en codificación binaria de CGM (Computer Graphics Metafile), ISO 8632.



**ODA/ODIF proporciona tres estados para el intercambio de documentos: procesable, formateado y formateado-procesable.**

**El estandar ODA no es aislado, sino que referencia a y es referenciado desde, otros estandares ISO.**

**El desarrollo de estándares para proceso e intercambio de documentos de oficina se ha realizado por varios años. En los Estados Unidos el comité X3V1 está trabajando en esta área. Este comité alinea su trabajo con el realizado por la ISO/IEC JTC1/SC18.**

**La norma ODA (ISO 8613-89) ha sido manejada por organismos europeos, particularmente en Inglaterra y Alemania.**

**Varias extensiones de ODA han sido adoptadas o están casi por liberarse. Se han incorporado áreas en el estandar ODA que incluyen color, seguridad, representación alterna de texto, estilos de disposición, descripción formal de ODA (FODA.- Formal Description of ODA), datos en documentos (spreadsheets) y API's para manipulación y proceso de documentos ODA.**

**Un gran número de organizaciones están produciendo perfiles de aplicaciones de documentos (DAP's), para intercambio de documentos ODA. Dichas organizaciones son: EWOS, CCITT, NIST, TOP (de MAP/TOP), INTAP (The Interoperability Technology Association for Information Processing) y PAGODA (The Profile Alignment Group for ODA).**

**Otros grupos que tienen interés o influencia sobre el estandar ODA, son:**

- ODA Consortium (ODAC), que es un grupo de vendedores.
- Professional Publishing Interchange Standards Committee, el cual es principalmente de organizaciones de vendedores de la industria de la publicidad de los Estados Unidos.
- PODA (Promote ODA).
- Object Management Group.
- OSF
- X/Open.
- COS.

**SGML.- STANDARD GENERALIZED MARKUP LANGUAGE.  
SDIF.- STANDARD DOCUMENT INTERCHANGE.**

**El SGML trata a los documentos como una estructura de varios tipos de elementos. Las aplicaciones tratan a estos elementos en diferentes formas.**

SGML es más relevante donde el contenido del texto de un documento sirve para unir el flujo de un todo. SGML puede ser utilizado muy bien para publicidad convencional, publicidad de base de datos multimedia y proceso de documentos de oficina donde se requiera legibilidad e intercambio entre sistemas de publicidad. No es apropiado para anuncios de display y material promocional estilizado.

El formato de intercambio para SGML, conocido como SDIF, es estandarizado como el ISO 9069.

SGML tiene una importante aceptación en ambientes técnicos de publicidad.

### ***ELECTRONIC DATA INTERCHANGE (EDI)***

Este estandar se enfoca al intercambio de datos de tipo comercial, está pensado para comunicar órdenes de compra y facturas en un formato estandar y para usar un mecanismo de entrega que utilice un correo electrónico basado en el estandar X.400. Con la utilización de este estandar se obtienen gran reducción de costos puesto que no se tiene que retectar la información.

En virtud de que EDI es utilizado en diferentes áreas de la industria internacional, algunas compañías requieren que sus proveedores manejen este estandar para lograr con ello que sus tratos sean más expeditos.

EDI también está disponible para negocios no tan grandes, existe software disponible para PC.

Hay también proveedores (terceras partes) de preparación de documentos EDI y servicios de entrega de los mismos.

El desarrollo de estándares ha sido también un factor principal para la utilización de EDI. El trabajo X12 de la American National Standards Institute (ANSI) ha producido estandares EDI para utilizarse en los Estados Unidos y, los estandares X.400, han llegado a estar ampliamente disponibles y relativamente estables para utilizarse como el vehiculo de entrega para documentos EDI.

La existencia de mecanismos de entrega por terceras partes y de X.400 E-mail, aparentemente nos da opciones que pueden ser buenas, pero que

también significan obstáculos cuando una compañía desea usar X.400 y la otra quiere utilizar redes manejadas por terceras partes.

Por otra parte, mientras que X.400 es capaz de realizar transmisiones multimedia y que no deben tener dificultad para llevar información de formas comerciales, el agente usuario de X.400 no es particularmente bien apropiado para formas EDI. El actual trabajo sobre una nueva recomendación CCITT X.435: "Message Handling Systems: EDI Message System", intentará resolver las diferencias entre X.400 y EDI.

Se ha considerado incluir a EDI en el "Government OSI Protocol" (GOSIP) versión 3, ya que el gobierno de los Estados Unidos es un usuario principal de EDI.

Inicialmente EDI no involucrará todas las fases necesarias en la transferencia de este tipo de documentos.

La primer fase involucrará ordenes de compra y facturas, después se abarcará información del status y eventualmente, transferencia de pagos también será parte de EDI. Existen ciertas restricciones en cuanto a seguridad y legalidad tales como firmas electrónicas que son necesarias en estas áreas.

EDI está disponible en dos formas:

- La primera por parte de vendedores específicos (propietario).
- Versiones estandarizadas.

En virtud de esta disponibilidad, es fácil que se confunda a los usuarios cuando ven el acrónimo EDI, ya que no siempre se trata del estandar EDI.

La versión estandarizada EDI es definida por el estandar X12 de la ANSI.

Hay una variación internacional de EDI, la ISO 9735: EDI for Administrative, Commerce, and Transportation (EDIFACT) que es similar al estandar X12 pero que no son lo mismo, aunque se ha establecido el objetivo de que los dos serán compatibles. Aun cuando la palabra "compatible" da lugar a muchas especulaciones, se presume que en este caso significa que se podrá convertir de EDI a EDIFACT y vice versa.

Justo es aclarar que tal conversión también es posible entre las formas de vendedor particulares de EDI y el EDI de la ANSI.

EDI es modelado sobre la estructura de las formas de papel para la información comercial:

Una sola forma es considerada un conjunto de transacciones X12.

Cada línea de una forma dada es expresada como un segmento de datos de X12, los cuales tienen campos para cantidades, unidades de medida, de precio, número de identificación de producto, etc.

Uno o más conjuntos de transacciones, o formas electrónicas, pueden ser puestas en el sobre (envelope) de intercambio de X12. ("header" y "trailer").

El conjunto de transacciones estandarizadas incluye ordenes de compra, facturas, solicitudes de cotizaciones y su respuesta, notas de embarque, lista de precios y cambios a las formas solicitadas.

El sobre de intercambio y su conjunto de transacciones incluidas, son enviados electrónicamente, este envío puede ser por medio de esquemas de red propietarios, pero el estándar X.400 para correo electrónico, se presume que será la vía utilizada en el futuro.

EDIF.- Electronic Design Interchange Format.

Esta especificación tiene que ver con la transferencia de la descripción de productos eléctricos en formato de archivo ASCII entre sistemas eléctricos CAE y "PCBlayout". El objetivo del estándar es proporcionar un formato neutral no ambiguo para compartir datos entre diferentes sistemas eléctricos CAD. El formato EDIF está dividido en diferentes vistas de los datos del producto. Este estándar está siendo desarrollado bajo el patrocinio de la United States Electronics Industry Association (EIA) con aprobación de la ANSI. Es uno de los elementos críticos del trabajo PDES/STEP.

## **PDES/STEP**

PDES.- Product Data Exchange.

STEP.- Standard for the Exchange of Product Data.

Es un esfuerzo de estandarización para proporcionar un método para la transferencia de datos entre sistemas de cómputo heterogéneos y entre gobiernos, industria y organizaciones proveedores. En la ISO esta norma es conocida como STEP.

Estos estándares son complejos ya que requiere referencias a otros estándares en el área de comunicaciones, gráficas, lenguajes de modelados de base de datos, intercambio de datos y otros.

Las aplicaciones y herramientas para PDES/STEP serán implementadas usando una "Standard STEP Data Access Interface specification" (SDAIS), no obstante la tecnología de administración de datos usada para manejar el producto de datos STEP. El componente de sistema de cómputo que proporciona la facilidad para modelado de ambiente de tal forma que pueda ser manejado, es IRDS.

PDES/STEP es derivado del trabajo hecho en 1970 por la United States Air Force. Eventualmente el trabajo surgido como PDES el cual es pensado para manejar todo tipo de productos de datos a través del ciclo de vida de los productos. Actualmente se realiza trabajo paralelo de este tipo en Europa. Los usuarios de MCAD, ECAD, MCAE, ECEA, CAM, CIM y otras aplicaciones de este tipo son la principal audiencia para este estandar.

En la ISO el Comité responsable de STEP es el TC 184. En los Estados Unidos el trabajo PDES/STEP esta siendo hecho por la "IGES/PDES Organization" (IPO) administrada por NIST.

Hay organizaciones similares para actividades STEP en Europa y Japón.

PDES, Inc., creada en enero de 1988 en el "United State Department of Defense" es una asociación de fabricantes y compañías de sistemas de cómputo con un gran interés en este estandar.

### **2.3.5 C.- COMMUNICATIONS INTERFACES (INTERFACES DE COMUNICACION)**

En este elemento se incluyen los componentes de comunicaciones relacionados con redes, conectividad e interoperabilidad.

Las redes pueden ser de área local (LAN) o de área amplia (WAN).

Uno de los deseos más fervientes que en los últimos años se ha tenido por parte de los usuarios, es el poder interconectar e interoperar con redes heterogeneas y es más que el deseo la necesidad, la que ha obligado a organizaciones formales que se dedican a los estándares a trabajar sobre normas en esta área. Han surgido dos trabajos que tienen respaldo por parte de este tipo de organizaciones:

El modelo OSI (Open Systems Interconnection) de la ISO.

Y el Internet Protocol (TCP/IP).

#### **TCP/IP.- "Internet Protocol"**

El "Internet Protocol" surgió de las redes del ambiente militar de los Estados Unidos de Norteamérica como un juego de estándares de facto. Este protocolo proporciona facilidades similares a aquellas especificadas en la capa cuarta del modelo de referencia OSI (capa de transporte), así como también varios servicios empacados de aplicación tales como correo y transferencia de archivos.

El "Internet Protocol", comunmente conocido como TCP/IP, es un juego de protocolos con TCP e IP como los mejores conocidos del juego y está constituidos en capas (como OSI). Algunos de los protocolos, incluyendo IP, TCP y UDP, proporcionan las funciones de bajo nivel necesarias para muchas aplicaciones, los otros protocolos tienen otras tareas específicas.

Las funciones de Internet están agrupadas por capas, hay 4 capas independientes:

- Capa de acceso a la red (Capa de aplicación): FTP, TELNET, SMPT.
- Capa inter-red (red): IP.
- Capa host-a-host (transporte): TCP, UDP.
- Capa de proceso.

Cada capa consiste de un juego de tareas interrelacionadas y manejables que proporcionan servicios a las capas adyacentes.

La capa de acceso a la red se refiere a las tareas que habilitan a los datos para que sean intercambiados entre un host y la red a la cual está conectado, así como entre dos dispositivos de la misma red.

La capa de inter-red realiza tareas que permiten que los datos sean movidos desde una red a otra aún si las redes soportan diferente tecnología y medio de transmisión. Esta capa utiliza el protocolo IP y el protocolo ICMP (Internet Control Message Protocol).

La capa host-a-host ejecuta tareas que compensan la incertidumbre típica de los servicios de la capa inter-red, usando los protocolos TCP (Transmission Control Protocol) y el UDP (User Datagram Protocol).

La capa de proceso permite que los datos sean intercambiados por varias aplicaciones. Los servicios más importantes tradicionales proporcionados por la capa de proceso son:

- Transferencia de archivos utilizando el protocolo FTP (File Transfer Protocol).
- Correo, utilizando el protocolo SMTP (Simple Mail Transfer Protocol).
- Login remoto.

Estos servicios deberán estar presentes en cualquier implementación de TCP/IP aún cuando alguna implementación de PC puede que no sea capaz de soportar correo. Las redes de PC utilizando el modelo cliente/servidor, deberán tener la capacidad de manejar correo, así como otros servicios.

En TCP/IP la información es transferida como una secuencia de datagramas. Un datagrama es una colección de datos que son enviados como una sola unidad. Generalmente pero no siempre, el término datagrama y paquete son indistintos. Los datagramas son enviados a través de la red individualmente. TCP es el responsable de dividir un mensaje en datagramas, reensamblarlo en el punto receptor y re-enviar cualquier parte que se pierda. IP es responsable de rutear datagramas individuales.

TCP/IP está basado sobre el modelo "catenet". Este modelo asume que un gran número de redes independientes están conectadas juntas por medio de compuertas (gateways). Los usuarios tienen acceso a cualquier red. El ruteo de la información de una red a otra es transparente para el usuario. Los usuarios deben conocer solamente una dirección "Internet" o nombre de la red a la cual se desea acceder.

Los protocolos TCP/IP fueron originados como ya dije, en el ambiente militar de los Estados Unidos de Norteamérica pero actualmente son aceptados y estandarizados por la comunidad "Internet".

Internet es una agrupación de redes del gobierno, militares, académicas e Institutos de Investigación que están todos conectados a y pueden comunicarse con, uno con otro. Los documentos del protocolo "Internet" son estándares de grupos de usuarios y éstos han llegado a ser el estándar de facto para las comunicaciones de redes dentro de la industria de la tecnología de la información.

Hoy la red "Internet" consiste de más de 2000 redes WAN, LAN y redes regionales. Es patrocinada por los departamentos del gobierno de los Estados Unidos:

- Department of Defense.
- Defense Advanced Research Projects Agency.
- National Science Foundation.
- Department of Energy.
- National Institute of Health.
- NIST.- National Institute of Standards and Technology.

Para participar en la Internet, una red debe ser patrocinada por uno de los departamentos mencionados.

El "Federal Research Internet Coordinating Committee" compuesto por representantes de cada participante de las agencias de gobierno de los Estados Unidos, es el órgano al cual se dirigen las necesidades de la Internet.

A diferencia del modelo de referencia OSI, el cual define los tipos de tareas que deben ser realizadas en un ambiente de redes de sistemas abiertos, pero no define el protocolo a ser utilizado, Internet primero define los protocolos para su ambiente funcional y la arquitectura viene después.

Los estándares "Internet" son emitidos como un documento RFC (Request for Comments) mantenido por la IAB (Internet Activities Board).

La IAB es responsable de coordinar la evolución de los protocolos "Internet". La mayoría de las actividades para el desarrollo y estandarización de protocolo Internet, toma lugar en grupos de trabajo de la IETF (Internet Engineering Task Force).



Los protocolos que van a llegar a ser estándar en la Internet pasan por una serie de pasos (propuesta del estándar, borrador del estándar, y estándar) involucrando grandes cantidades de escrutinio y prueba experimental. Estos pasos también son llamados "estados" (states).

Es una práctica general de IAB que ningún estándar propuesto puede ser promovido a borrador de estándar sin al menos dos implementaciones independientes. La promoción de borrador de estándar a estándar, generalmente requiere experiencia operacional y de interoperabilidad demostrada de dos o más implementaciones.

La siguiente lista se refiere a los estándares IAB y su número de RFC.

PROTOCOLO	NOMBRE	RFC
ARP	Address Resolution Protocol	826
RARP	A Reverse Address Resolution Protocol	903
IP-ARPA	Internet Protocol on ARPANET	BBN1822
IP-WB	Internet Protocol on Wideband Network	907
IP-X25	Internet Protocol on X.25 Network	877
IP-E	Internet Protocol on Ethernet Networks	894
IP-EE	Internet Protocol on Exp. Ethernet Nets	895
IP-EEE	Internet Protocol on IEEE 802	1042
IP-DC	Internet Protocol on DC Networks	891
IP-HC	Internet Protocol on Hyperchannel	1044
IP-ARC	Internet Protocol on ARCNET	1051
IP-SLIPS	Transmission of IP over Serial Lines	1055
IP-NETBIOS	Transmission of IP over NETBIOS	1088
IP-FDDI	Transmission of IP over FDDI	1103
IP-IPX	Transmission of 802.2 over IPX Networks	1132

Conforme los protocolos OSI se generalicen, la IAB reconoce que habrá una necesidad creciente de soportar la interoperación de ellos con los protocolos TCP/IP. La IETF (Internet Engineering Task Force) está formulando estrategias para tal interoperación.

En un futuro, la IAB espera que una gran parte de la Internet soportará ambos protocolos de red, el OSI y el TCP/IP en paralelo y que será posible correr aplicaciones OSI a través de Internet utilizando la pila completa de protocolos OSI.

**OSI.- Open Systems Interconnection.** (Modelo de referencia de Interconexión de Sistemas Abiertos)

El esfuerzo OSI es un ejemplo de los extensos recursos con los que las organizaciones y gobiernos están dispuestos a comprometerse para el desarrollo internacional de estándares de sistemas abiertos. El trabajo que está llevando a cabo sobre estándares OSI, desarrollo de perfiles y pruebas, indican el manejo acelerado y amplio para el desarrollo real de productos OSI.

La solución OSI está pensada para resolver las dificultades asociadas con la comunicación entre sistemas de cómputo de múltiples vendedores. El modelo consta de siete capas.

En la ISO el estándar del modelo de referencia OSI es el ISO 7498 y en la CCITT es el X.200.

Las siete capas son las siguientes:

<b>CAPA 1</b>	<b>Física.</b>
<b>CAPA 2</b>	<b>Enlace de datos (información de conexión básica).</b>
<b>CAPA 3</b>	<b>De red (ruteo en WAN, conexión de LAN's separadas).</b>
<b>CAPA 4</b>	<b>De transporte (transferencia de datos entre sistemas).</b>
<b>CAPA 5</b>	<b>Sesión (Transferencia de datos entre aplicaciones).</b>
<b>CAPA 6</b>	<b>Presentación (transformaciones tales como ASCII y EBCDIC).</b>
<b>CAPA 7</b>	<b>De aplicación (Servicios de aplicaciones).</b>

Desde la perspectiva del usuario, la capa más inferior del modelo OSI (física), tiene que ver con el tipo de LAN que se utilice:

Ethernet.  
Token Ring.  
FDDI.

Para el usuario, la capa más externa (aplicación), vienen siendo los servicios que en primer término ellos conocen:

Transferencia de archivos (FTAM).  
Correo electrónico (X.400).  
Servicios de directorio (X.500).

Los estándares de OSI a nivel servicios de aplicación, no proporcionan interfaces de programa de aplicación, los proyectos a cargo de TCOS de la IEEE y otros organismos, son los que están haciendo investigaciones acerca de API's para proporcionar estos servicios.

Por años se han hecho esfuerzos internacionales sobre estándares OSI distribuidos a través de las 7 capas de modelo de referencia. Pero aún con este gran número de estándares, será necesario que constantemente surjan nuevos estándares, ya que nuevas aplicaciones y nueva tecnología de comunicaciones los requerirán.

### **HISTORIA Y CONDICION ACTUAL:**

El esfuerzo OSI se inició en 1977-1978 en la ISO para combinar y estandarizar el trabajo que anteriormente se había hecho sobre sistemas abiertos y diseño de computadoras.

A continuación se listan los comités que más se identifican con el trabajo OSI.

Los Subcomités ISO/IEC JTC1 y grupos de trabajo con énfasis particular sobre OSI y los estándares relacionados con el mismo.

(JTC1.- Joint Technical Committee Núm. 1. ISO/IEC dedicado a los estándares de la tecnología de la información.)

**ISO/IEC JCT1/SC 8: Telecomunicaciones e intercambio de información entre sistemas.**

- GRUPO DE TRABAJO 1** Capa enlace de datos.
- GRUPO DE TRABAJO 2** Capa de red.
- GRUPO DE TRABAJO 3** Capa física.
- GRUPO DE TRABAJO 4** Capa de transporte.

**ISO/IEC JTC1/SC 18: Texto y sistemas de oficina.**

- GRUPO DE TRABAJO 1** Requerimientos de usuario y sistemas de oficina (ODA/ODIF).
- GRUPO DE TRABAJO 3** Arquitectura de documento.
- GRUPO DE TRABAJO 4** Procedimientos para intercambio de texto.
- GRUPO DE TRABAJO 5** Arquitectura del contenido de documentos.
- GRUPO DE TRABAJO 8** Descripción de texto y lenguajes de proceso.
- GRUPO DE TRABAJO 9** Interfaces de sistemas de usuario y símbolos.

**ISO/IEC JTC1/SC 21: Recuperación de información, transferencia y administración para interconexión de sistemas abiertos.**

- GRUPO DE TRABAJO 1** Arquitectura OSI.
- GRUPO DE TRABAJO 3** Base de datos.
- GRUPO DE TRABAJO 4** Administración de sistemas OSI.
- GRUPO DE TRABAJO 5** Servicios de aplicación específicos.
- GRUPO DE TRABAJO 6** Sesión OSI, Servicios comunes de aplicación y de presentación.
- GRUPO DE TRABAJO 7** Proceso distribuido abierto

**ISO/IEC JTC1/SC 27: Técnicas comunes de seguridad para aplicaciones IT**  
(Information Technology, término común en el ambiente internacional).

**GRUPO DE TRABAJO 1** Algoritmos de llave secreta y aplicaciones.

**GRUPO DE TRABAJO 2** Sistemas criptográficos de llave pública y modos de uso.

**GRUPO DE TRABAJO 3** Utilización de técnicas de encriptamiento en arquitecturas de comunicación.

**ISO/IEC JTC1/SGFS: Special Group on Functional Standards.e**

La CCITT también está involucrada en el esfuerzo OSI. En las áreas de interés común, las recomendaciones CCITT para telecomunicaciones son frecuentemente idénticas en contenido técnico a los estándares ISO, por otro lado, algunos estándares ISO/OSI tales como X.25 y X.400 están basados sobre recomendaciones CCITT, consecuentemente, ciertos perfiles OSI y estándares ISO incluyen recomendaciones CCITT.

La ISO y la CCITT han tenido una gran relación durante la década pasada, aquí se refleja la relación estrechísima que hay entre las tecnologías de cómputo y de comunicaciones.

#### **ESTANDARES FUNCIONALES: PERFILES ESTANDARIZADOS INTERNACIONALMENTE**

Para que se logre un verdadero sistema abierto, los productos OSI y sistemas de comunicación, deben estar más que de acuerdo a los diversos estándares OSI. Esto es porque, los estándares OSI ofrecen gran flexibilidad y amplia funcionalidad, por ello pueden contener muchas y diversas clases de opciones. El uso de implementaciones de estándares, sin un acuerdo previo en la elección de dichas opciones, puede provocar que los resultados no sean del todo deseables en cuanto rendimiento y que además se incremente el costo de dicha implementación.

Un factor que está dando auge al esfuerzo internacional OSI es la creación del mercado interno europeo el cual requerirá que los estándares usados para proveer compatibilidad, sean los mismos a través de toda Europa. La naturaleza internacional de la tecnología de la información y el creciente intercambio internacional de la información, hacen vital que aquellos estándares sean también estándares del mercado mundial.

Por lo tanto, los fabricantes y los usuarios del mundo que están interesados en productos OSI tienen que juntarse para lograr acuerdos sobre juegos o combinaciones de estándares OSI, con opciones seleccionadas en cada estándar, consideradas necesarias para soportar una función dada. Estas funciones incluyen, por ejemplo, una aplicación dada tal como correo electrónico, o un ambiente de red.

Los documentos que especifican acuerdos similares alcanzados a diferentes niveles (nacional, internacional), se les denomina de varias formas:

- Perfiles.
- Estándares funcionales.
- Acuerdos de implementación.
- Especificaciones de implementadores.

Los documentos que surgen de los acuerdos de la ISO son designados como "International Standardized Profiles" (ISPs). La selección de estándares y opciones, consideradas en un ISP, son desarrolladas para proporcionar las bases para el desarrollo de productos OSI y pruebas de conformidad con el objetivo definitivo de la interoperabilidad de los sistemas.

Los principios básicos para el contenido de perfiles OSI establecen que un perfil OSI especifica la aplicación de uno o más estándares base OSI en soporte de un requerimiento específico para comunicación entre sistemas. Para el propósito de ISPs, los estándares base están restringidos a estándares internacionales o recomendaciones CCITT. Un perfil no requiere de cualquier parte de la estructura definida por el modelo de referencia básico para OSI, ni define la funcionalidad total OSI de un sistema, sino solamente de esa parte relevante para la función que está siendo definida.

Los primeros ISPs publicados por ISO son aquellos sobre FTAM (ISO 10607) y sobre ciertos mecanismos de transporte OSI (ISO 10608 Y 10609).

Después de surgir como ISPs aprobados por ISO, estos documentos estarán disponibles para el uso del público y en particular proporcionarán una base para el desarrollo de sistemas de prueba reconocidos internacionalmente.

### **TRABAJO DE PERFILES RELACIONADOS CON OSI**

Si los perfiles que son implementados en todo el mundo difieren, el resultado producido no será interoperable completamente aún si ellos están conforme a los estándares OSI. Por lo tanto, en adición a los grupos de estándares

formales, hay organizaciones en diversas partes del mundo que desarrollan perfiles funcionales relacionados con OSI.

Estos grupos generalmente son de 3 tipos:

- Los que se dedican a hacer Workshops abiertos de OSI.

La mayoría de los acuerdos se han logrado en este tipo de foros. Los "workshop" sirven como foros públicos para usuarios y proveedores, en este caso de productos OSI.

Algunos de los que organizan "workshops" son:

OIW.- OSI Implementors Workshop.  
EWOS.- European Workshop for Open System.  
AOW.- Asia-Oceania Workshop.

- Consorcios que agrupan a industrias.

Algunos son:

Manufacturing Automation Protocol User Group (MAP).  
Technical and Office Protocols User Group (TOP).  
Standards Promotion and Application Group (SPAG).  
Corporation for Open Systems (COS).  
Promoting Conference for OSI (POSI).  
OSI/Network Management Forum.

- Grupos de organizaciones y gobierno.

Muchos organismos de gobierno en diferentes partes del mundo, han desarrollado políticas acerca de tecnología de información, haciéndolas obligatorias o recomendado el uso eventual de productos OSI en el sector público. El gobierno de Inglaterra ha sido uno de los pioneros en esto. Un documento de este tipo es generalmente llamado un "GOVERNMENT OSI PROFILE" (GOSIP).

## **PRUEBAS: DE CONFORMIDAD Y DE INTEROPERABILIDAD**

### **CONFORMIDAD**

El ISO/IEC JTC1/SC21/MWG1 está desarrollando un estándar internacional integrado de varias partes, el ISO 9646. Este estándar establece una guía para pruebas de conformidad. El trabajo de la CCITT que equivale a este estándar, es la Recomendación X.290.

ISO 9646 "Conformance Methodology and Framework" se encarga del significado de "conformidad", de las especificaciones para juegos de prueba abstractas, del lenguaje de notación de las pruebas y de los métodos de pruebas.

El estándar también define un "Declaración de conformidad de instrumentación de protocolo" (PICS.- Protocol implementation conformance statement) como una declaración del vendedor de las capacidades y opciones las cuales han sido instrumentadas en una implementación o sistema OSI y sobre cualquier característica la cual ha sido omitida. Estas declaraciones serán usadas para pruebas de conformidad de una instrumentación en particular. La "European Commission" ha establecido un juego de pruebas de conformidad para que se utilicen a través de Europa. Se espera tengan acuerdos de prueba con los Estados Unidos.

El ISO 9646 define un sistema o implementación que cumple con la conformidad establecida como:

"Aquel que se demuestra que satisface tanto requerimientos estáticos como dinámicos, consistentes con las capacidades y opciones establecidas en el PICS".

## **INTEROPERABILIDAD**

La condición de "conformidad" es necesaria, pero por si misma no es suficiente para garantizar la capacidad de interoperabilidad en productos o implementaciones OSI. Dos implementaciones que cumplan con "Conformidad", puede que no operen por varias razones:

- Las dos implementaciones han elegido soportar diferentes opciones en el perfil que afecta la interoperabilidad (el perfil está mal diseñado).
- Las pruebas de conformidad no fueron lo suficientemente rigurosas o fueron mal hechas. También puede haber algún aspecto del protocolo o algún aspecto del comportamiento de la implementación que está



siendo probada, que las pruebas de conformidad no usaron pero que afectan la interoperabilidad.

- . El contenido técnico del estándar está mal.
- . Las dos capas subyacentes de las implementaciones no pueden interoperar.

Hay varias redes que proporcionan el servicio de prueba de conformidad a vendedores, estas son.

- . OSInet (anteriormente promovido por NIST y ahora promovido por COS).
- . Eurosinet en Europa.
- . OSicom en Australia.

Recientemente 6 de estas redes fueron combinadas para formar una a nivel mundial de interoperabilidad OSI llamada "OSlone".

El principal objetivo de "OSlone" es el establecimiento de una plataforma común para redes regionales OSI para que faciliten la promoción y el avance de conectividad e interoperabilidad OSI a nivel mundial.

## **ESTANDARES NUEVOS**

El arreglo de estándares ISO que aplica a las 7 capas del modelo de referencia OSI es muy grande y amplio.

Además están surgiendo estándares para ocuparse de nuevas necesidades y nueva tecnología, algunos son los siguientes:

ISO/CCITT X.400 Correo e intercambio de mensajes.

Otra vez, para evitar caos entre diferentes modos de transferir información, en 1984 la CCITT definió una serie de protocolos a los que llamó MHS "Message Handling Service" (Sistemas de tratamiento de mensajes) en sus recomendaciones de la serie X.400. La ISO trató de incorporarlos en la capa de aplicación del modelo OSI con el nombre de MOTIS (ISO 10021) (Sistema de intercambio de textos orientados a mensajes), aunque esta incorporación no es totalmente directa debido a la falta de estructura de la X.400. Sin embargo en 1988 la CCITT modificó el X.400 con objeto de hacerla compatible con MOTIS.

El MOTIS se ocupa de todos los aspectos del sistema de correo electrónico, comenzando desde el instante en el que el extremo de origen decide escribir un mensaje, y terminando en el instante en que el extremo receptor lo desecha.

La recomendación X.400 de CCITT estandariza un protocolo a nivel de aplicación para intercambio de correo, mensajes y documentos entre sistemas locales de correo que cumplen con el X.400. El X.400 puede proporcionar facilidades de distribución para documentos utilizando los formatos ODA e IGES.

#### ISO/CCITT X.500 Servicios de Directorio.

La Recomendación X.500 de la CCITT (ISO 9594-1988) proporciona las capacidades de directorio requeridas por las aplicaciones OSI, los procesos de administración OSI, otras entidades de las capas OSI, y los servicios de telecomunicaciones.

El grupo de trabajo 1003.17 de POSIX está examinando los servicios de nombres de directorio y espacio en un contexto de X.500 como ellos se relacionan a los sistemas distribuidos POSIX.

#### FTAM.- FILE TRANSFER ACCESS AND MANAGEMENT (ISO 8571)

Define una variedad de funciones y opciones que cubren el rango completo de accesos posibles a archivos remotos. El estándar propuesto incluye un modelo que proporciona un marco de referencia para:

- Identificación de interfaces.
- Posición de interfaces (relativas una con otra).
- Identificación de procesos.
- Facilidades y datos necesarios en cada interface.
- Posicionamiento de la interface en términos de un ciclo de vida de los sistemas de información.
- Identificación de ligas alternativas asociadas con cada interface.

Esta área de estandarización también está siendo atendida por el grupo de trabajo 1238 de la IEEE y el 1003.8 de POSIX, ambos están trabajando sobre interfaces del sistema de archivos y semántica para acceso transparente de archivos distribuidos.

**El comité POSIX está examinando mecanismos de acceso remoto de archivos tales como NFS, RFS, AFS y FTAM.**

**Llamada de procedimiento remoto (RPC.- Remote Procedure Calls).**

**RPC proporcionan la transparencia de ubicación y transporte, así como la independencia en una red de múltiples vendedores. El RPC proporciona al desarrollador de aplicaciones los medios de construcción para llamadas de procedimientos generalizados, desde un sistema único, hasta una red de sistemas.**

**RPC es muy apropiado para redes operando en un ambiente cliente/servidor.**

**RPCs pueden ser utilizados para una gran variedad de aplicaciones. Por ejemplo, de Base de Datos, en donde se puede reducir la cantidad de redundancia de base de datos de un sistema a otro, ya que una sola copia de la base de datos en la red puede ser accesada por RPCs.**

**Actualmente no existe un RPC internacional, pero tanto la ANSI como la ECMA (European Computer Manufacturer's Association) tienen propuestas a nivel borrador. La OSF ha adoptado un RPC utilizando un lenguaje de definición de interface (IDL) similar al ANSI C como parte de su "Distributed Computing Environment".**

**ISO/CCITT X.25.- Redes de Area Extendida (WAN).**

**La recomendación X.25 de la CCITT especifica estándares para redes públicas de área extendida. El objetivo de esta norma es evitar que diferentes países lleguen a desarrollar interfaces que no sean compatibles. En 1974 la CCITT propuso una norma internacional para protocolos de acceso a redes para las capas 1, 2 y 3 del modelo OSI. Estos protocolos se revisaron en 1976, 1980 y 1984. Al conjunto de todas estas normas se le conoce como X.25.**

**La recomendación especifica la forma en que los paquetes de datos vienen del usuario hacia la red pública. Define los estándares que gobiernan la relación entre el usuario y la red en tres niveles:**

**NIVEL 1.- El nivel físico, define la parte mecánica, eléctrica, funcional y del procedimiento del enlace físico entre el equipo del usuario y la red pública.**

**NIVEL 2.-** A nivel tramas de datos (frame), define el procedimiento de acceso a la línea para un intercambio de datos sobre la línea de comunicaciones.

**NIVEL 3.-** A nivel de paquete, define el formato del paquete y procedimientos para el intercambio de paquetes conteniendo información de control y datos de usuario.

**LAN.-** Local Area Network (Estándares para redes de área local).

**IEEE 802.3** Tecnología Ethernet.

El estándar 802.3 está escrito para la estandarización de LANs que utilizan el método de acceso conocido como Acceso Múltiple por Detección de Portadora/Con Detección de Colisión (CSMA/CD.- Carrier Sense Múltiple Access with Collision Detection).

Proporciona la definición para una LAN de 10 Mb/s que opera sobre diferentes medios físicos: cable coaxial estándar (thick wire), cable coaxial delgado (Thin Wire), par trenzado blindado (Shielded Twisted-pair), par trenzado sin blindaje (Unshielded Twisted-pair).

Este estándar es una derivación del estándar original Ethernet desarrollado por Digital Equipment Corporation, Xerox Corporation e Intel Corporation, fué introducido en 1980.

**IEEE 802.5.-** Paso de Testigo en Anillo (TOKEN RING).

Este estándar define una LAN basada en un anillo.

Al contrario de la Ethernet, la Token Ring es una LAN en la cual las estaciones están serialmente conectadas para formar un anillo. El método de acceso es controlado por el uso de un patrón de bits especial llamado testigo (token) que circula alrededor del anillo siempre que las estaciones se encuentren inactivas. Cuando una estación quiere transmitir una trama, es necesario capturar el testigo y quitarlo del anillo, antes de efectuar la transmisión. Debido a que solamente hay un testigo (token), una sola estación puede transmitir en un instante dado.

**ANSI X3T9.5.-** Interface de Datos Distribuidos en Fibra Optica (FDDI.- Fiber Distributed Data Interface).

**El FDDI es un nuevo estándar para LANs de múltiples vendedores desarrollado bajo las reglas de la ANSI.**

**El estándar define una LAN FDDI de 100 Mb/s. Esta tecnología mueve grandes cantidades de datos y es de mayor velocidad que la tecnología Ethernet.**

**Es una fibra óptica del tipo de paso de testigo en anillo, es de alto rendimiento, opera a 100 Mbps para cubrir distancias de hasta 200 Km y soporta hasta 1000 estaciones conectadas. Tomando en cuenta su ancho de banda es ideal para conectar redes tipo LAN de cobre.**

**Todas las recomendaciones FDDI X3T9.5 son enviadas a las organizaciones ISO correspondientes para su inclusión en el estándar OSI.**

**Aún cuando no están incluidos en el modelo MUSIC, existen otros 3 elementos muy importantes dentro de los sistemas abiertos:**

- Seguridad.**
- Internacionalización.**
- Desarrollo de Software.**

## 2.4 SEGURIDAD

El control de la seguridad tiene gran impacto en la totalidad de un sistema en lo que se refiere a:

Rendimiento.

Facilidad de utilización.

Interoperabilidad de las aplicaciones.

Administración del sistema (la red, el procesador, la configuración y los datos mismos).

Es por eso muy difícil concebir un sistema abierto con gran seguridad, debido a que la interoperabilidad y la portabilidad de las aplicaciones crean potencial vulnerabilidad.

Lo anterior se denota cuando vemos que se están desarrollando mecanismos y estándares para seguridad que se enfocan a aspectos independientes de un sistema, tales como seguridad en las redes o seguridad en base de datos.

Actualmente no hay estándares que toquen el tema de seguridad en una forma total e integral.

Los estándares de seguridad de hoy se pueden categorizar en dos: de comunicaciones y los relacionados con los sistemas operativos.

### SEGURIDAD EN COMUNICACIONES

ISO 7498-2.- Se denomina Arquitectura de Seguridad OSI, nos da una base conceptual para seguridad OSI. Describe servicios tales como validación, control de acceso, confidencialidad, integridad (detección de modificación). Este estándar está en proceso de depuración.

CITT X.500.- (Recomendaciones para directorio), contiene un marco de referencia para validación. El X.509 se refiere a validación criptográfica.

ANSI X3.92-1981 y FIPS-46.- Están siendo usados en comunicaciones y en aplicaciones financieras, se refieren a algoritmos de encriptación. El estándar ANSI X12.42 es utilizado en validación del EDI.

## SEGURIDAD EN SISTEMAS OPERATIVOS

Algunos estándares que actualmente están a la mano son:

**POSIX 1003.8** Este grupo está trabajando sobre una interface de seguridad para su ambiente de sistemas abiertos.

El objetivo es especificar funciones adicionales al sistema y comandos para seguridad en el POSIX a fin de promover la portabilidad de las aplicaciones.

Este trabajo define interfaces de seguridad para:

Control discrecional de acceso.  
Control obligatorio de acceso.  
Etiquetado de la información.

Hay otro estándar que no es un estándar formal, está descrito en un documento conocido como el "Libro Anaranjado", fue publicado por "Trusted Computer System Evaluation Criteria" (TCSEC) del Departamento de Defensa de los Estados Unidos; dicho documento identifica las siguientes áreas de requerimientos básicos:

Políticas de seguridad.  
Marcado de la información.  
Identificación.  
Contabilidad.  
Protección continua.

Identifica clases de evaluación contra las cuales pueden ser fijados los mecanismos de seguridad.

Estas clases son 7 niveles de seguridad en un sistema operativo:

D Protección mínima.  
C1 Control de acceso discrecional.  
C2 Acceso controlado con validación y auditoría.  
B1 Control de acceso obligatorio, salida etiquetada.  
B2 Documentación, control de configuración. canales encubiertos de estructura interna, administración de la facilidad de confidencialidad.  
B3 Listas de control de acceso, estructura interna.  
A1 Pruebas formales y diseño del sistema verificado.

## **2.5 INTERNACIONALIZACION**

Un sistema abierto debe ser adaptable a un ambiente internacional para que éste sea internacionalmente exitoso.

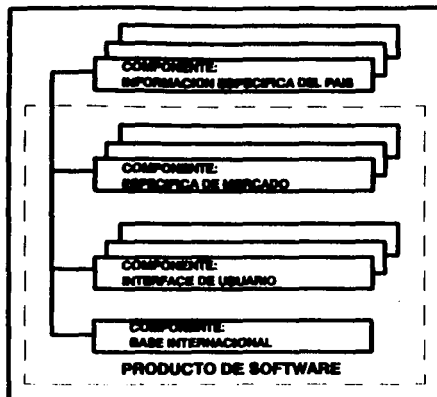
Hay requerimientos y convenciones que difieren de un lugar a otro (país, región, etc.). Cualquier estándar internacional debe darle al usuario el medio para definir y cambiar aspectos del ambiente y adaptarlos a su entorno local.

Algunos aspectos que deben ser considerados en la internacionalización son:

- Lenguajes de programación.
- Formato de números.
- Juego de caracteres.
- Archivos de mensajes.
- Formas y Menús.
- Textos.
- Estructura y formato de datos.
- Mapas de teclado.
- Controles de impresión.
- Otros.

En la siguiente figura aparece el modelo de un producto internacional y sus componentes:





**FIGURA NUM. 21.- MODELO DE PRODUCTO INTERNACIONAL**

**COMPONENTE: BASE INTERNACIONAL.-** Es la parte del producto que se vende por todo el mundo y no se modifica, pero puede incluir algunas variantes que son seleccionables por parte del usuario.

**COMPONENTE INTERFACE DE USUARIO.-** Incluye el lenguaje y elementos de proceso de texto. Esta componente es específica para el lenguaje local y debe satisfacer las necesidades lingüísticas y culturales de un grupo específico de usuarios.

**COMPONENTE: ESPECIFICA DE MERCADO.-** Esta componente tiene como objetivo satisfacer requerimientos de ciertas regiones que comparten algunas convenciones culturales y un lenguaje, como por ejemplo: Región de hispanoamérica.

**COMPONENTE: INFORMACION ESPECIFICA DEL PAIS.** Incluyen un juego de documentación requerida producida para satisfacer todas las regulaciones para venta de producto en un país específico.

En algunos casos la internacionalización es manejada en algún estándar específico, por ejemplo:

**Estándar C.-** Define un juego de interfaces para soporte a la internacionalización.

**POSIX 1003.1.-** Proporciona una interface programable que soporta selección de programa de ambientes. También especifica que otras funciones deben ajustar su comportamiento basados en el ambiente seleccionado.

El estandar propuesto POSIX 1003.2 proporciona un mecanismo para que los usuarios definan y especifiquen el ambiente.

La mayoría de los grupos de la industria que ya han entrado al mercado de los sistemas abiertos han incorporado las especificaciones de la internacionalización en sus ambientes de sistemas abiertos.

X/Open, el Consorcio X y los grupos de trabajo POSIX están trabajando de forma cooperativa en el nivel ISO sobre elementos de estandarización para ambientes distribuidos.

## **2.6 AMBIENTE DE DESARROLLO DE SOFTWARE (SDE.- Software Development Environment)**

El SDE es un área de aplicación particular que afecta todos los intereses en la industria, especialmente a aquellos usuarios involucrados en el desarrollo de grandes y complejos proyectos, así como de proveedores de software.

El desarrollo de software tiene que ver mucho con la portabilidad pero también para requerimientos particulares de integración e interoperabilidad.

En este ambiente están considerados:

Una base central de datos (repository) para guardar información acerca de especificaciones, diseños, código, planes de configuración, programas y pruebas.

Herramientas.

Una interface entre el repositorio y las herramientas para asegurar la portabilidad de las mismas.

### **Entorno de Herramienta Común Portable (PCTE.- Portable Common Tool Environment).**

Es una interface de herramienta pública para un grupo de servicios para construir Ambientes de Soporte a Proyectos Integrados (IPSE.- Integrated Project Support Environments) y para integración de herramientas en tal ambiente.

La base de PCTE es un sistema de administración de objetos. Incluye una metabase, y servicios para distribución, acceso al sistema de archivos, ejecución de procesos e imágenes, comunicación de interprocesos, notificación de eventos, transacciones, respuestas, conexión a red, seguridad y contabilidad.

PCTE es el resultado de un proyecto de investigación iniciado en 1984 en la Comunidad Europea.

Este estandar fué adoptado por la European Computer Manufacturers Association (ECMA) en 1990.

**Estandar de Integración de Herramientas (ATIS.- Tools Integration Standard).**

ATIS es un estandar propuesto tanto por la ANSI (X3H4) como por la ISO (JTC1/SC 21/AWG3). Incluye un juego común de elementos de los modelos de datos ANSI y ISO-IRDS. El objetivo de este estandar es proporcionar servicios que puedan soportar el ciclo de vida total del desarrollo de aplicaciones. Su alcance incluye repositorios tradicionales CASE y diccionarios de datos. Su repositorio orientado a objetos proporciona un mecanismo estándar tanto para independencia de datos como para compartir datos.

El trabajo original para ATIS fué hecho en ambiente CAD. Desde entonces ha sido extendido para eliminar las dependencias de sistema operativo, mejora para soporte relacional, modularización y facilidades generales, así como adición de conceptos de diccionario.

**SERVICIOS DE INTEGRACION CASE (CIS.- CASE Integration Services)**

El comité CIS es un grupo de organizaciones interesadas en la integración de herramientas de ingeniería de software en ambientes CASE. Este comité incluye vendedores de sistemas, de software, usuarios de herramientas CASE y organismos de investigación.

Este grupo ha dividido el dominio de integración en:

- Intercambio de datos.
- Integración de datos.
- Integración de control.
- Integración de presentación.

Está enfocado en las áreas de integración de información y control, tomando un enfoque orientado a objetos.

## **2.7 PERFILES DE AMBIENTES DE APLICACION (AEP.- *Application Environment Profile*)**

De los estándares podemos decir que:

- Son muchos los que se nos presentan en el panorama.
- No todos aplican a todas las situaciones.
- Para que sean exitosos, deberán agruparse y trabajar juntos.
- Se deben seleccionar y especificar aquellos apropiados para ciertas necesidades y ambientes particulares.

Un perfil de ambiente de aplicación es un grupo de estándares que incluyen opciones que son necesarias para soportar los requerimientos funcionales de un dominio específico de aplicaciones, como por ejemplo aplicaciones bancarias. Son un medio muy eficaz para lograr una selección adecuada de estándares simplificando la tarea de identificar estándares relevantes y opciones aplicables.

Un AEP identifica:

- Las necesidades del área de aplicación.
- Los estándares disponibles para satisfacer dichas necesidades.
- Diferencias entre capacidades necesarias y estándares existentes.
- Carencias entre los estándares disponibles.

### **CARACTERISTICAS DE LOS PERFILES.**

Entre las características de los perfiles tenemos:

- Proporcionan un mapa para combinar el arreglo de estándares en una forma coherente.
- Deben estar completos respecto a su ambiente.
- Deben ser coherentes, es decir, estar lógicamente conectados.
- Se requiere al menos un lenguaje.

El concepto de Perfil de Ambiente de Aplicación derivó del concepto de perfiles OSI el cual fué desarrollado para el manejo de varios cientos de estándares OSI. De este gran grupo de estándares es esencial que los usuarios seleccionen un juego de estándares y opciones que no solamente

se interconecten exitosamente, sino que también satisfagan completamente las necesidades funcionales específicas que se requieran.

Tal vez el perfil OSI mejor conocido es el GOSIP (GOVERNMENT OSI PROFILE), especificado por NIST del gobierno de los Estados Unidos. Hay otros, tales como el UK GOSIP (De Inglaterra).

El modelo MUSIC proporciona un marco de referencia para el desarrollo de un perfil. Para la mayoría de las áreas de aplicación, se especificarán estándares relevante en las áreas cubiertas por todos los elementos del modelo MUSIC. La siguiente figura muestra una modificación del modelo MUSIC en términos de la funcionalidad manejada por cada uno de los elementos de MUSIC. De esta representación es patente que la mayoría de los perfiles de sistemas abiertos especificarán estándares que se refieran a uno o más componentes de cada elemento.

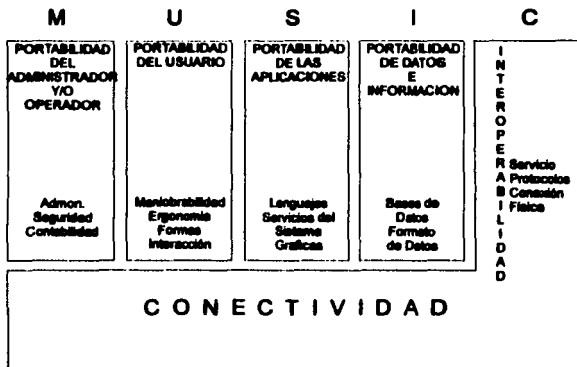


FIGURA NUM. 22.- COMPONENTES DEL PERFIL DE SISTEMAS ABIERTOS.

Debido a la asociación común de ciertos lenguajes con dominios específicos de aplicación, la selección del lenguaje puede ser muy evidente, es decir, no será difícil identificar el lenguaje adecuado.

Cuando se requiere más de un lenguaje para un perfil dado, el perfil necesita enfocarse a la relación entre los lenguajes, por ejemplo, cuando se requiere codificación de datos comunes, se espera que se tengan archivos compartidos.

Los perfiles tienen tres funciones esenciales y relacionadas.

**PRIMERO.-** Proporcionan una base clara para comunicaciones acerca de las características necesarias de sistemas abiertos requeridas por un usuario o una aplicación.

Un perfil puede ser usado como modelo para la obtención de documentos que pueden servir de base para la evaluación de productos de vendedores considerando los requerimientos tanto de sistemas como de aplicaciones.

**SEGUNDO.-** Un perfil proporciona una guía para aplicaciones y sistemas abiertos, de tal forma que define cuando está apropiadamente implementado una aplicación o un sistema. Una aplicación que se adapta a un perfil dado de estándares es adecuadamente implementada para tal perfil.

El perfil proporciona agrupaciones lógicas de estándares que los desarrolladores pueden usar para simplificar la tarea de obtener aplicaciones para ambientes específicos. La aceptación de perfiles por parte de la industria ampliará el campo de acción para los desarrolladores.

Para los implementadores de sistemas, los perfiles ofrecen un beneficio similar proporcionando los fundamentos para enfocarse sobre segmentos de mercado específicos. Un vendedor puede seleccionar algunos perfiles y optimizar sus sistemas para aquellos ambientes. Otros pueden producir sistemas generales que satisfagan las necesidades de muchos perfiles.

Algunos perfiles pueden ser mutuamente exclusivos. Tal hecho ayuda a los vendedores a asegurar que ellos satisfacen las necesidades de un cierto mercado.

**TERCERO.-** Las pruebas de calidad de los sistemas o aplicaciones pueden ser realizadas en términos de perfiles. Los perfiles eliminarán la ambigüedad que existe en las actuales pruebas donde cada estándar es tratado separadamente y no se toma en cuenta nada acerca de las opciones o relaciones entre estándares.

Los perfiles también puede influenciar la forma en que los estándares son desarrollados dado que ellos identifican diferencias en el trabajo existente acerca de estándares. Si los grupos desarrolladores de perfiles identifican requerimientos para alguna funcionalidad adicional, los estándares para tal funcionalidad pueden ser desarrollados en una forma en que sean consistentes con ella y para el trabajo existente.

Si dos grupos de perfiles requieren funcionalidades mutuamente exclusivas de un solo estándar, deberán conseguir la documentación de capacidades opcionales de dicho estándar y cada uno deberá usar la opción apropiada a su perfil.

Los perfiles pueden ser desarrollados a diferentes niveles:

- En una compañía pueden ser desarrollados para satisfacer ciertas necesidades departamentales o corporativas.
- Por área de aplicación específica o por industria única o particular.
- Entre compañías para trabajar cooperativamente en el desarrollo de AEP'S que soporten aplicaciones específicas para un mercado común de su interés.

Conforme progresa el trabajo de los perfiles, pueden ser desarrollados perfiles que incluyan otros perfiles. Por ejemplo, en el campo bancario, los servicios que necesite el departamento de crédito son diferentes a los que necesite el departamento de sistema de transferencia electrónica de fondos. Pero los sistemas que dan servicio a ambas áreas pueden necesitar interactuar con uno u otro tipo de sistemas distribuidos. Dado que los perfiles pueden ser estandarizados, un perfil para un gran dominio de aplicación puede ser construido a partir de perfiles estandarizados con un enfoque específico.



Los usuarios pueden considerar los siguientes pasos para maximizar la efectividad de su trabajo acerca de perfiles.

*. Aislar áreas lógicamente funcionales para perfilarlas, particularmente las áreas que pueden compartir estándares comunes a través de una organización o grupo de organizaciones.*

*. Identificar estándares, opciones y parámetros a ser incluidos en los perfiles. Documentar áreas donde sean necesarios estándares adicionales e identificar las características que pueden ser importantes en el desarrollo de estándares para llenar las diferencias.*

*. Armonizar los perfiles incluyendo estándares, opciones y parámetros adicionales con perfiles que se relacionen mucho.*

*. Publicar los perfiles y/o moverlos a través de canales formales para su aceptación. Cuando sea apropiado, trabajar con grupos industriales.*

*. Mantener documentados los estándares y motivar a los organismos de estándares para que llenen las diferencias entre estándares.*

## **2.8 DESARROLLO DE PERFILES EN ORGANIZACIONES DE ESTANDARES**

El desarrollo de perfiles por parte de organismos acreditados tales como ISO, IEEE y EWOS está todavía incompleto. Por ejemplo hay grupos activos de perfiles en el esfuerzo POSIX de la IEEE trabajando sobre supercómputo, proceso transaccional, cómputo de tiempo real y multiprocesamiento. El trabajo de perfiles es una actividad principal en EWOS. El resultado del trabajo hecho por cada grupo deberá moverse hacia el comité ISO sobre estándares funcionales para armonización internacional y reconocimiento de ISO como perfiles estandarizados internacionales (ISPs.-International Standardized Profiles).

Los perfiles preparados por las organizaciones de estándares, difieren de los AEPs en que ellos están compuestos solamente de estándares propuestos o formalizados y generalmente se enfocan sobre funcionalidades muy amplias en lugar de un ambiente específico de cómputo.

Por ejemplo, POSIX 1003.13 está enfocado sobre un perfil para capacidades de cómputo en tiempo real. No está enfocado a una aplicación específica de cómputo en tiempo real tal como obtención de datos o rastreo de inventario.

Un ISP tiene la misma condición en la comunidad internacional como lo tienen los estándares internacionales.

El ISP está definido en el documento ISO/IEC TR 10000, "Marco taxonómico y directorio de perfiles", como "un acuerdo internacional para documentos armonizados los cuales identifican un grupo de estándares, junto con opciones y parámetros, necesarios para realizar una función o grupo de funciones".

Los ISPs definen las combinaciones de estándares base con el propósito de:

- . Identificar los estándares base, junto con clases apropiadas, subjuergos, opciones y parámetros necesarios para realizar funciones identificadas para propósitos tales como INTEROPERABILIDAD.
- . Proporcionar un sistema para referenciar los diversos usos de estándares base los cuales son significativos para los usuarios y proveedores semejantes.

. Proporcionar un medio para mejorar la disponibilidad de implementaciones consistentes de grupos definidos funcionalmente de estándares base que son esperados para ser los componentes principales de sistemas de aplicación real.

. Promover uniformidad en el desarrollo de pruebas para sistemas que implementen las funciones asociadas con los perfiles.

La taxonomía de perfil incluida en la parte II de el documento TR 10000 es la estructura y clasificación en la cual caerán los IPSs. Esta estructura da una especificación de primer nivel de perfiles, incluyendo cualquier restricción técnica. Las clasifica y también especifica un número de relaciones entre ellos.

La armonización de perfiles significa que diferentes perfiles usan las mismas palabras para referenciar el mismo estandar cuando ellos están buscando la misma funcionalidad. El beneficio principal de la armonización es asegurar que los perfiles especificados por diferentes grupos tengan tanto en común uno con otro como sea posible.

Como sucede en el trabajo de los estándares, el trabajo correspondiente a los perfiles formales toma tiempo y consume recursos. Dado que este trabajo específicamente está encaminado a las necesidades del usuario, esta es un área muy valiosa para que los usuarios se involucren en el proceso de los estándares. Como una ayuda a los usuarios interesados en este trabajo, el grupo de trabajo del POSIX 1003.0 de la IEEE ha desarrollado un documento para desarrollo de perfiles llamado "Guía para el Ambiente de Sistemas Abiertos POSIX" (POSIX Open System Environment). Esta guía también está siendo usada como documento de trabajo por parte de EWOS.

El trabajo acerca de perfiles de organizaciones formales de estándares, se extiende fuertemente sobre el trabajo interno de grupos de usuarios, agencias de gobierno y de asociaciones industriales sobre los ambientes de sistemas abiertos (OSEs.-Open Systems Environments). El trabajo de OSE de estos grupos no es acerca de AEPs ni de perfiles estandarizados, más bien, ellos generalmente describen un ambiente completo para operación de sistemas abiertos.

El OSE de cada organización refleja el enfoque, necesidades y experiencia de sus respectivos miembros.

Por ejemplo, NIST ha desarrollado el "Applications Portability Profile" (APP) el cual referencia estándares ANSI e ISO. Identifica tres áreas funcionales que necesitan ser direccionadas por trabajo adicional de estándares: administración del sistema, acceso transparente de archivos y ventanas. Estas son áreas de interés para la comunidad usuaria en el gobierno federal de los Estados Unidos.

Similarmemente, X/Open ha desarrollado un "Common Applications Environment" (CAE) que incluye estándares y otras especificaciones.

La Open Software Foundation cuenta con su "Applications Environment Specification" (AES) el cual abarca un rango muy amplio de estándares ISO más extensiones con las que la OSF ha contribuido como recomendaciones para estándares.

EWOS ha establecido el Expert Group on Common Applications Environment el cual ha comenzado a desarrollar una taxonomía de perfiles para sus OSE. En esta taxonomía, los perfiles están divididos en:

- . Perfiles de ambientes de sistemas abiertos.
- . Perfiles para plataformas de ambientes de sistemas abiertos.
- . Perfiles para los componentes de ambientes de sistemas abiertos.

Los perfiles de los componentes se subdividen adicionalmente en perfiles que se relacionan fuertemente con los elementos del modelo MUSIC.

La siguiente figura muestra las ventajas y limitaciones de los perfiles creados en varios niveles.

NIVEL DE PERFIL	VENTAJAS	LIMITACIONES
ISP.- International Standardized Profiles	Aceptación a nivel mundial	Sólo contiene estándares internacionales
Perfiles nacionales	Aumenta el poder de compra del sector público	Puede no tener compatibilidad y consistencia cultural internacional
Perfiles de la Industria de la Tecnología de la Información	Control sobre especificaciones y su evolución  Disponibilidad de instrumentación muy amplia  Mayor compromiso con proveedores a través del uso de perfiles que son comunes en la industria	Pueden incluir funciones no -estándar  Puede no tener control sobre especificaciones y su evolución  Posibles alternativas sin restricción
Perfiles de la industria vertical	Alienta el desarrollo de aplicaciones de proveedores de software independientes  Guía clara para todos los jugadores de la industria	Puede estar aislado de las tendencias de la industria de la tecnología de la información
Perfiles de la empresa	Alientan el uso común de la empresa.  Las habilidades que se sostengan en el software son reutilizables	Puede limitar el acceso de aplicaciones externamente desarrolladas  Pueden ser aplicaciones dependientes de la implementación
Perfiles de Departamento	Reflejan las aplicaciones y el hardware real en uso	Pueden estar aislados de las habilidades y aplicaciones de la industria, base del futuro.

FIG. NUM. 23.- NIVELES EN LOS QUE EXISTEN LOS PERFILES.

Conforme se lleguen a formalizar más perfiles y éstos sean más ampliamente conocidos, llegarán a ser más aceptados.

Los perfiles formales como ISPs también ofrecen la gran oportunidad para la portabilidad e interoperabilidad pero su aplicabilidad funcional puede ser limitada.

En otra parte del espectro, un perfil de ambiente de aplicación desarrollado para una organización específica ofrece gran aplicabilidad funcional y lo más adecuado para la organización pero su gran especificidad puede limitar opciones de implementación y apertura evolutiva.

Conforme la industria acepta las definiciones de perfiles, los usuarios encontrarán disponibles más implementaciones de sistemas abiertos y aplicaciones disponibles.

## **2.9 AMBIENTES DE SISTEMAS ABIERTOS (OSE.- Open Systems Environments)**

Hay varias organizaciones que están desarrollando y especificando ambientes de sistemas abiertos, a continuación se mencionan algunos y éstos son presentados utilizando los elementos del modelo MUSIC.

### **TRABAJO DE ESTANDARES DE LA IEEE (TCOS):**

Aquí podemos ver el trabajo de el "Computer Society's Technical Committee on Operating Systems Standards Subcommittee" de la IEEE. Esta es la sede de los estándares POSIX así como del trabajo que se viene realizando acerca de ventanas y de algunas interfaces de programación de aplicaciones (API) para el área de comunicaciones.

Se podrá observar que una gran parte de este trabajo se traslapa con la labor que otros grupos están llevando a cabo.

La Guía para el ambiente de sistemas abiertos POSIX que está siendo ensamblada por el grupo de trabajo POSIX 1003.0 describe el OSE POSIX y sirve de guía para los desarrolladores y para las organizaciones involucradas en el trabajo de perfiles.

M	U	S	I	C
P1003-7 Administración del Sistema  P1003.-15 Service Batch  P1003-8 Seguridad	P1003-2 DUPE  <hr/> Mantenibilidad P1201.2 UMBB(*)  (P1201.3) TestG P1201.1	C P1003.18 Fortran P1003.9 Ada P1003.5  <hr/> POSIX.2 P1003.2 (DP945-2) POSIX.1 P1003.1 (IS 9945-1) Tiempo Real, etc P1003.4	ISAM (*) (Suguido)	X.400 API P1224 FTAM API P1238 TFA P1003.8 Protocolo Independiente P103.12 NS/D8 P1003.17

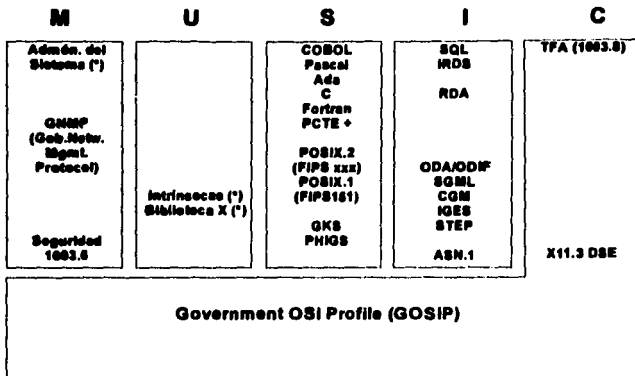
(\*) Actualmente no está dentro de un proyecto de estándares

FIG. 24.- TCOS DE IEEE

**PERFIL PARA PORTABILIDAD DE APLICACIONES DE NIST. (APP.-APPLICATIONS PORTABILITY PROFILE)**

NIST ha estado desarrollando un Perfil para Portabilidad de Aplicaciones, enfocándose sobre el uso de estándares ISO y ANSI como requerimientos principales. En este perfil, se han identificado 3 áreas que necesitan estandarización: administración del sistema, acceso transparente de archivos y ventanas. El trabajo de POSIX de la IEEE en sus comités 1003.7 y 1003.8, como ya vimos anteriormente, direccionan las 2 primeras áreas. NIST ha propuesto a FIPS basado sobre el sistema "X Window" versión 11.3 incluyendo la XLibrary y los "X Intrinsics".





(\*) No está actualmente en un proyecto de estándares

**FIG. 25.- NIST APPLICATION PORTABILITY PROFILE (APP)**

**AMBIENTE COMUN DE APLICACIONES DE LA X/OPEN. (CAE.- COMMON APPLICATIONS ENVIRONMENT)**

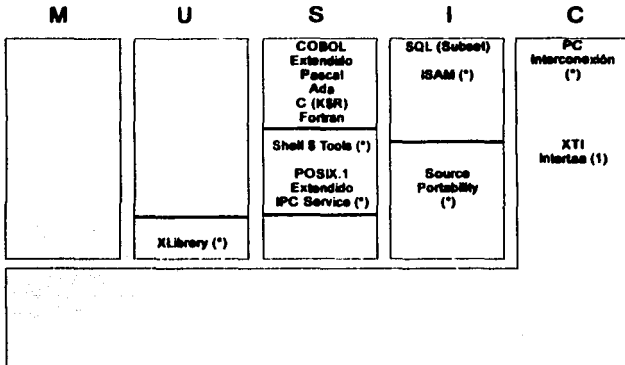
Este trabajo está basado en el CAE, está considerado en el X/Open Portability Guide (Guía núm.3). De acuerdo a este documento, el CAE está basado sobre las interfaces especificadas en el estándar POSIX 1003.1-1988 de la IEEE, extendido para cubrir otros aspectos que se requieran de un OSE.

El CAE incluye COBOL, POSIX 1003.1, SQL que son un subconjunto de ANSI y los estándares ISO, así como una especificación de la Xlib que es un subconjunto de FIPS de la NIST para el sistema "Xindow". COBOL y PL/I son los únicos dos lenguajes que incluyen interfaces ISAM.

El trabajo de portabilidad de código fuente de X/Open incluye especificaciones de los medios que exceden el alcance de las interfaces de software.

X/Open considera la interconexión de PC's con una solución de producto específico basado sobre un producto de vendedor específico y clones de tal producto.

Como se puede observar, la X/Open no está incluyendo en su CAE ningún estándar para cubrir el componente de administración.



(\*) No está incluido actualmente en ningún proyecto de normas  
 (1) X/Open Transport Interface

FIG. 26.- X/OPEN COMMON APPLICATION ENVIRONMENT (CAE)

**ESPECIFICACION DE AMBIENTE DE APLICACIONES DE LA OPEN SOFTWARE FOUNDATION. (AES.- APPLICATIONS ENVIRONMENT SPECIFICATION)**

La OSF ha tomado de varias fuentes los elementos que ha incluido en su AES, van desde POSIX de la IEEE, los estándares ISO y gran parte del CAE de la X/Open.

Una inclusión principal en el área de comunicaciones es el Internet Protocol (TCP/IP) que tiene la intención de moverse hacia estándares OSI conforme los productos vayan estando disponibles. En el área de ventanas, la OSF ha desarrollado el "Motif Toolkit".

También se incluye una guía de estilo la cual se direcciona a la maniobrabilidad.

Aquí tampoco se considera ningún estándar dentro de la componente de administración.

M	U	S	I	C
	Style Guide UI-UserInterface Language	COBOL Pascal Ada C Fortran LISP BASIC	SQL Level II	SFTP (*) FTP (*) VT (*)
	Toolkit (Motif) Intrinsecos (*) Biblioteca X (*)	POSIX.2 POSIX.1		DCE (*) TCP/IP
		GKS PHIGS		X11.3DSE

(\*) No está actualmente en un proyecto de estándares

**FIG. 27.- OPEN SOFTWARE FOUNDATION APPLICATION ENVIRONMENT SPECIFICATION (AES)**

## 2.10 CONCLUSIONES DEL CAPITULO

Cuando se empezaba a hablar de sistemas abiertos era muy común asociar el tema inmediatamente con UNIX y una de las razones es que los vendedores así lo enfocaban, después de una revisión de todos los conceptos vistos en este capítulo, podemos percatarnos de que "SISTEMAS ABIERTOS" no solo se refiere a un sistema operativo.

Como se pudo observar, el foco central de este capítulo fueron los **ESTANDARES**, que vienen siendo el ingrediente clave de esta mezcla de conceptos. John S. Quarterman y Susanne Wilhelm, en su libro "UNIX, POSIX and Open Systems", hacen una analogía muy interesante, comparan a los estándares con piezas de un rompecabezas. Con dichas piezas se pueden formar "n" rompecabezas, de diferente tamaño y con muchas diferentes combinaciones. También considera que un rompecabezas puede ser considerado como un perfil de ambiente de aplicación (AEP.- Application Environment Profile).

Así lo podemos ver, dependiendo de nuestro entorno, nuestro objetivo y la capacidad de recursos a invertir en un proyecto, podemos ensamblar las diferentes piezas de nuestro rompecabezas particular (nuestro ambiente de sistemas abiertos) y aplicarlo adecuadamente a nuestro negocio.

**CAPITULO 3**  
**TECNOLOGIA ACTUAL**

## **CAPITULO 3**

### **TECNOLOGIA ACTUAL**

El mundo de los sistemas abiertos como ya vimos en el capítulo anterior, tiene que ver con gran cantidad de estándares, y en determinado momento corremos el riesgo de perdernos. La forma en que el modelo MUSIC enmarca a los estándares, las funciones, los productos y las características de los sistemas abiertos, me parece muy clara y congruente, así que es ese mismo marco con el que me permitiré abordar el tema de la tecnología actual.

El término "TECNOLOGIA ACTUAL" en esta tesis se refiere a aquella tecnología que se considera así misma como perteneciente al mundo de los sistemas abiertos.

### **3. TECNOLOGIA ACTUAL**

Una vez que hemos revisado algunos ambientes de sistemas abiertos, enmarcados éstos en el modelo MUSIC, será fácil revisar lo que nos ofrece la tecnología actual que se dice perteneciente a los sistemas abiertos.

M.- Management (Administración).

#### **SOFTWARE.**

Sería ideal que se contara con un producto que nos permitiera administrar los recursos totales de nuestro ambiente, pero la realidad nos demuestra que se administran los elementos por separado, por un lado, tenemos productos para administrar redes y por otro los que administran las bases de datos (DBMS).

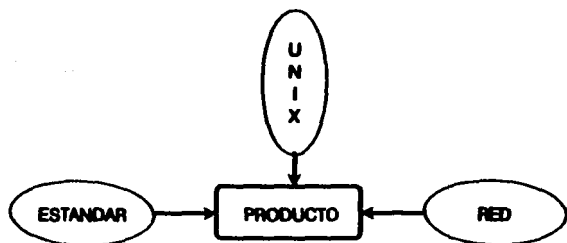
Por lo que respecta a los DBMS los mencionaré más adelante, aquí solo me enfocaré a la administración de redes y específicamente a los de ambiente UNIX.

Para administrar redes existen muy variados productos, pero desafortunadamente la mayoría son propietarios, es decir, cada uno está basado en su propio juego de estándares, reconocidos y garantizados por su vendedor.

Ejemplo de propietarios tenemos:

- Netview de IBM para ambientes SNA y UNIX.
- Grace LAN Network Manager de Technology Works.
- Radar de Sonic Systems.
- Open View de HP.
- SuNet Manager de SUN.

Dentro de los productos debemos considerar aquellos que soportan algún estandar reconocido, así que tenemos:



El protocolo TCP/IP es nativo de los ambientes UNIX, con lo que se tiene la ventaja de que en diversos UNIX contamos con SNMP.

Aún cuando los productos de administración que se encuentran actualmente en el mercado son propietarios, algunos ofrecen soporte para el estándar SNMP.

SNMP como ya se vió en el capítulo 2, es un protocolo que forma parte del grupo que integra la pila TCP/IP.

Aunque no es el mejor, existe un producto que me pareció adecuado para este análisis, su nombre es SunNet Manager y su fabricante es SUN, a continuación veremos lo más importante de este producto.

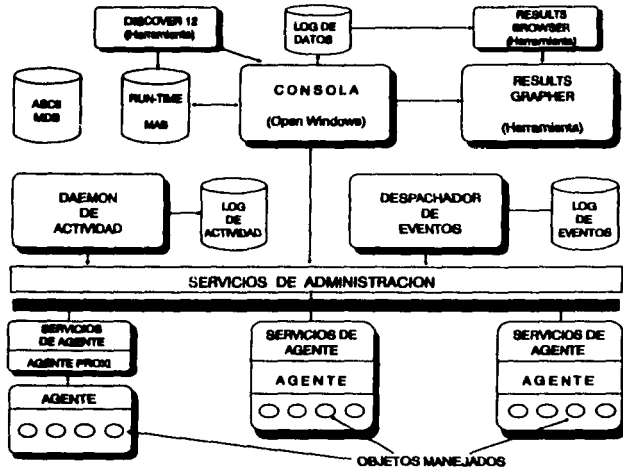
SunNet Manager es una plataforma para la administración de redes. Como plataforma de administración, este producto se basa en el modelo "ADMINISTRADOR-AGENTE" descrito en el modelo OSI, se basa en el TCP/IP y la interface de usuario que presenta, está basada en el X11, específicamente en "Open Look".

El producto tiene dos protagonistas principales: El administrador y el agente.

El administrador es un proceso iniciado por el usuario. El agente es un proceso que accesa los objetos manejados y recoge datos que serán enviados al administrador. Los objetos manejados son las entidades que están siendo administradas.

La estructura general de SunNet Manager se aprecia en la siguiente figura:





MDB = Management Data Base

FIGURA NUM. 28.- ESTRUCTURA DE SUNNET MANAGER.

#### HARDWARE.

Por lo que respecta al hardware, en el mercado existen ruteadores y concentradores, que tienen soporte para el SNMP, es por eso que un producto que administre redes que soporte el SNMP puede tener control de este tipo de dispositivos.

U.- USER INTERFACE (Interace de usuario).

X WINDOWS.

## HARDWARE.

Los sistemas X WINDOW pueden ejecutarse sobre distintos dispositivos físicos:

- La unidad de salida es una pantalla gráfica cuya resolución al menos debe ser VGA.
- Los dispositivos de entrada más comunes son el teclado y el ratón.

Otras opciones de entrada son:

Bola de tracción.  
Tableta digitalizadora.  
Pantalla táctil.

- Opciones de procesamiento:

- . Estaciones de trabajo.- La marca más popular en el mercado es Sun Microsystems.
- . Terminales X.- Son terminales con capacidad gráfica y sin disco. Este hardware es mucho más barato que las estaciones de trabajo.
- . Microcomputadoras.- Si se utiliza una microcomputadora, ésta deberá tener como requisito primordial una alta resolución gráfica, de preferencia Super VGA, el color es opcional.

El procesador mínimo es un 80386 o su equivalente (por ejemplo un 68030).

Es absolutamente necesario contar con un disco duro de gran capacidad y muy rápido. El sistema X Window no es como el DOS que requiere de un mínimo de espacio en disco, el sistema X requiere al menos de 110 MB para él y su entorno, UNIX incluido.

Como se vió en el capítulo 2, el sistema X Window incluye dos tareas: Ejecutar la aplicación de usuario y manejar gráficos. Así de que a menos de que se disponga de un auténtico sistema operativo multitarea, una microcomputadora no podrá atender a la aplicación de usuario y a los gráficos. Una microcomputadora aislada con sistema operativo DOS tan solo puede gestionar una de estas dos tareas, pero no ambas.

## SOFTWARE

UNIX y otros sistemas operativos multitarea con frecuencia controlan grandes microcomputadoras. Como ejemplo está el Open Desktop de Santa Cruz Operation, que se ha convertido en el sistema operativo UNIX estándar de las microcomputadoras de gama alta. Este paquete proporciona por completo todas las capacidades X Window.

Como vimos en el capítulo 2, existen 2 interfaces gráficas basadas en el sistema X Window: OSF/Motif y Open Look. Para programar aplicaciones gráficas utilizando el sistema X Window, es necesario hacer uso de 3 productos interrelacionados: Xlib, Xt Intrinsic y la interface de usuario seleccionada (puede ser OSF/Motif u Open Look, por ejemplo).

Los fabricantes y compañías que soportan Motif y Open Look se pueden apreciar en la siguiente tabla:

Fabricante	Motif	Open Look
IBM	V	T
DEC	V	T
HP	V	T
UNISYS	V	
Sun Microsystems	T	V
Solbourne	V	V
Compaq	V	
Dell	V	
Prime	V	
Data General	V	
Silicon Graphics	V	
MIPS	V	
NCR	V	
AT&T	V	V
Wang	V	
Hitachi	V	
Comodore		V

V - Soportado por el fabricante.

T - Soportado por otras compañías.

Fuente: A Mike Bugard, "Who's Winnig the GUI Race?" UNIXWORLD, agosto de 1991.

Un ejemplo de lo que hay hoy en el mercado, es el SOLARIS 2.3 y en este paquete se incluye, como parte básica del mismo, el "OPEN WINDOWS" Versión 3.3, que es un ambiente gráfico basado en el estándar del MIT, el X11 R5.

Otras implementaciones.

Existen implementaciones de interfaces gráficas que no se basan en X Window, sino en sus estandares propietarios, se mencionan aquí solo porque algunas de ellas son muy populares en el mercado, pero no hay que olvidar que no pertenecen al mundo de los sistemas abiertos:

Macintosh Finder.- El Apple Macintosh es una de las computadoras desarrolladas desde un principio con una interfaz gráfica de usuario. Como la filosofía GUI fue parte de su diseño original, mucha gente encontró que estas computadoras eran más fáciles de usar que las compatibles IBM. Sin embargo, el Macintosh Finder no ofrece la potencia de gestión de ventanas ni la flexibilidad de los sistemas X Window.

Microsoft Windows.- Como mencioné antes, éste se ha convertido en el estándar GUI de los sistemas compatibles IBM.

OS/2 Presentation Manager.- En abril de 1987 IBM lanzó la serie de microcomputadoras PS/2, muchas de las cuales corren bajo el sistema operativo OS/2. Presentation Manager (PM) fue el GUI estándar de las primeras versiones del OS/2. Su apariencia es bastante similar al MS-Windows.

El sistema DESQview.- Estrictamente hablando, el DESview no es un GUI ya que no se ejecuta en modo gráfico. Sin embargo, proporciona una alternativa a MS-Windows tanto para las microcomputadoras del tipo XT como las superiores. DESQview admite como unidad de entrada tanto el ratón como el teclado, proporciona tanto ventanas como multitarea, e incluso ejecuta MS-Windows. Se anuncia la versión DESQview/X que promete ejecutar X Window en computadoras compatibles IBM.

S.- SERVICES INTERFACES FOR PROGRAMS (Interfaces de servicios para programas).

Como ya se vió, este elemento incluye programas que son típicamente parte del sistema operativo:

- Interfaces para aplicaciones.

- Programas del sistema.
- Servicios de la plataforma de aplicaciones.
- Especificaciones de los lenguajes.
- Ligas a los lenguajes.
- API's para gráficos y otros servicios.

También vimos que debido a que diferentes fabricantes han producido diversos sistemas operativos UNIX, se inició el trabajo POSIX que como objetivo principal tiene el permitir la portabilidad de las aplicaciones.

En el mercado encontré que en el SOLARIS 2.3 que es un ambiente SunSoft (Sistema Operativo UNIX), ofrece soportar el estandar POSIX 1003.1.

También encontré que la especificación POSIX ha sido prometida para algunos otros sistemas operativos propietarios, incluyendo el HP MPE/IX, DEC VMS y OS/2.

Debido a la inmadurez que existe respecto a trabajar con este estandar, no encontré ningún ejemplo que nos indique de las bondades o limitaciones de este estándar, porque como ya vimos, en teoría ofrece bastante.

#### **I.- INFORMATION AND DATA FORMATS (Información y formato de datos)**

Dentro del elemento "I" del modelo MUSIC mencionaré a las bases de datos y sus manejadores porque tienen que ver mucho con la portabilidad.

Específicamente los lenguajes de bases de datos tienen como objetivo:

- Definición de bases de datos.
- Programas de aplicación de base de datos.

Entre los lenguajes de base de datos el más conocido es el SQL, también llamado lenguaje relacional SQL.

En diferentes sistemas de bases de datos relacionales es posible encontrar este estandar, por ejemplo:

- ORACLE.
- SYBASE.
- DB2.
- IDMS (última versión).
- INFORMIX

En el mercado existen diversos productos para generar aplicaciones en un ambiente de base de datos, así que si por una parte se definen los datos con SQL y por otra, los programas de aplicación también los tenemos en SQL, teóricamente podremos obtener portabilidad, al menos, a nivel fuente.

En función de la teoría anterior, si contamos con una aplicación en ORACLE, podremos portarla a SYBASE en formato fuente. Aquí hay que tomar en cuenta las características propias de cada manejador, para ver si conserva la funcionalidad la aplicación portada.

De los productos en el mercado que más se promueven como abiertos son ORACLE y SYBASE, pues aparte de que dan soporte para SQL, pueden trabajar en diferentes plataformas, desde PC hasta mainframe.

En particular el producto ORACLE, tiene su versión para diferentes plataformas y una aplicación hecha en esta herramienta, puede ser portada no solo a nivel fuente, en este caso, ORACLE cuenta con los mecanismos necesarios para recibir una aplicación en objeto y en cualquier plataforma, generar el ejecutable.

Por lo que se refiere a la información de los diccionarios de datos, aún no se cuenta con ningún estándar al respecto, por lo que cada producto tiene sus características propias.

En cuanto a las formas, cada manejador cuenta con herramientas propias para la generación de formas y reportes, por lo que no encontré a ninguno que soporte algún estándar. Con lo antes dicho no podremos tener portabilidad en formas y reportes, a menos que lo que genere sea código SQL, en cuyo caso lo que se puede hacer es adecuar dicha codificación a la herramienta a donde se porte.

#### C.- COMMUNICATION INTERFACES (Interfaces de comunicación)

La importancia que actualmente tiene este componente es ya demasiado visible, en vista de que en las organizaciones se cuenta con diferentes plataformas y es necesario interconectarlas, estamos obligados a conseguir productos que realmente permitan lograr la ya tan mencionada interoperabilidad de los sistemas.

Las redes con las que comúnmente cuentan las empresas trabajan con diferentes protocolos y para manejarlos recientemente se han presentado en

el mercado los "sistemas operativos de red", aunque no están sujetos a ningún estándar reconocido formalmente, considero importante mencionarlos:

Novell Netware.  
Microsoft LAN Manager.  
Banyan VINES.  
AppleShare Server.  
SCO LAN Manager.

Por otro lado, se pueden manejar en una sola estación de trabajo (DOS) más de un protocolo lo cual habilita al usuario a acceder diferentes tipos de recursos en la red. Si se tienen cargados tanto el protocolo SPX/IPX (protocolo propietario) como TCP/IP (estándar formal), se puede acceder el servidor de archivos NetWare y simultáneamente correr una sesión de terminal sobre un host UNIX. Esta flexibilidad es crítica para muchas organizaciones que cuentan con ambos tipos de servidores en la red, pero la realidad es que aquí tenemos conectividad.

En términos de comunicaciones de red, el sistema operativo de red por lo tanto, deberá ser capaz de soportar concurrentemente múltiples pilas de protocolos sobre el servidor. Esta capacidad es necesaria debido al rango de protocolos, hoy en uso, entre los cuales podemos mencionar:

- SPX/IPX (Sequenced Packet Exchange/Internetwork Packet Exchange) de Novell.
- TCP/IP.
- SNA de IBM.

Un producto que podemos mencionar, aunque es propietario, es el Netware de Novell, con el que, en una red Ethernet pueden estar viajando paquetes IP y paquetes IPX.

#### UNIX-SNA

Uno de los retos más difíciles que ha tenido la industria, ha sido el de integrar los mundos de redes UNIX y SNA. ¿Por qué menciono estos dos ambientes?, la respuesta es fácil, si recordamos que IBM ha sido por mucho tiempo el que ha dominado el mercado y UNIX es la alternativa más actual y viable, como consecuencia tenemos el deseado paso de redes SNA hacia redes UNIX.

Para interconectar redes SNA con UNIX, en el mercado se pueden encontrar diferentes "cajas" que, mediante diversos métodos, integran, en cierta medida, estos dos mundos. En general las funciones que permiten son las siguientes:

- Desde una terminal SNA de la familia 3270, acceder un servidor UNIX, abriendo una sesión en el host UNIX permitiendo trabajar como si se estuviera en una red UNIX.
- Desde una terminal UNIX acceder los recursos del mainframe en una red SNA.
- Transferir archivos desde un servidor UNIX a mainframe y viceversa.

Estos sistemas, compuestos de hardware y su software correspondiente, no integran totalmente los mundos UNIX y SNA, pero sí permiten, entre otras cosas:

- Aprovechar la inversión que en redes SNA se tenga.
- Permitir que mientras se lleva a cabo la migración de sistemas de SNA a UNIX se siga brindando el servicio a los usuarios desde cualquiera de sus terminales, ya sea de UNIX o SNA.
- Utilizar los grandes mainframes como medio de comunicación entre diversos puntos geográficos.

Los productos que están disponibles en el mercado nacional son los siguientes:

- Hypercom.- Hypercom Network Systems Inc.

Este hardware es un conmutador híbrido que combina las funcionalidades de un puente (bridge), un ruteador (router) y un conmutador (switcher). Puede recibir tráfico de LAN o SNA y transmitirlo por los enlaces WAN, como si cada uno de ellos tuviera el enlace y el ancho de banda disponibles para sí. Esto significa que las compañías que tengan plataformas LAN y SNA, podrán conducir el tráfico producido por ambas, sobre una sola infraestructura WAN. Lo anterior es posible porque en una misma "caja" se alojan tarjetas para cada tipo de protocolo.

- 3030 de Open Connect. Esta solución está integrada por un procesador UNIX conectado directamente a canal de un mainframe. El software que



permite conectar SNA con UNIX reside, tanto en el procesador UNIX como en el mainframe.

Este tipo de soluciones no es único, en el mercado están surgiendo más "cajas", pero como podemos ver la integración de estos tipos de redes es limitada, solo tenemos conectividad, no tenemos interoperabilidad.

## **CONECTIVIDAD EN HP**

### **BASICA**

Las soluciones de conectividad en la HP 3000 se basan en estándares como el TCP/IP, OSI y el propietario de IBM SNA. Los enlaces de la HP 3000 se agrupan en dos: aquellos que conectan a LAN'S y los que se conectan a WAN'S.

El soporte que ofrece para los protocolos de LAN:

802.3/Ethernet y 802.5 Token Ring. Por lo que respecta al protocolo FDDI está disponible junto con el sistema operativo MPE/iX versión 5.0

El soporte que ofrece para los protocolos de WAN's:

X.25 y SNA/SDLC (propietario de IBM).

## **INTEGRACION DE SISTEMAS EN AMBIENTES MULTIPROVEDOR**

Para lograr comunicación entre HP 3000 y HP 3000, Hewlett-Packard ofrece un conjunto de servicios para conexión en redes, conocido como HP NS (propietario de HP).

Para lograr comunicación entre HP 3000 y HP 9000, así como otras comunicaciones multiproveedor, se deberá utilizar el protocolo FTP para transferencia de archivos y para comunicación programa-a-programa Berkley Sockets. Si se requiere de funciones de terminal virtual en una red multiproveedor, se puede utilizar Telnet o NS VT de HP cuando se trate de HP's 3000 y 9000.

Los usuarios de HP que deseen implementar servicios OSI, deberán usar el FTAM para realizar transferencias de archivos y X.400 para llevar a cabo funciones de correo.

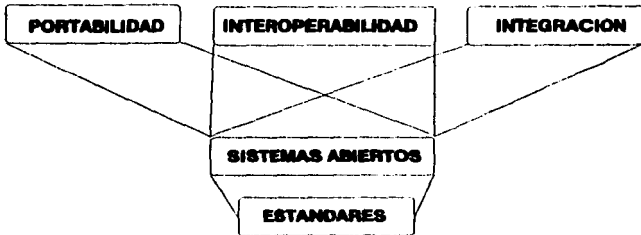
## **CAPITULO 4**

# **ANALISIS DE LA SITUACION Y PLANTEAMIENTO DE ALTERNATIVAS**

## CAPITULO 4

### **ANALISIS DE LA SITUACION Y PLANTEAMIENTO DE ALTERNATIVAS**

Como ya vimos, las promesas que han establecido los sistemas abiertos se resumen a:



Para cumplir con lo antes dicho, también se estableció que deberán sujetarse a estándares reconocidos formal e internacionalmente.

Después de haber visto lo que tenemos en el mercado, podemos decir que no hay mucho de donde elegir, también nos percatamos que los productos que se anuncian como abiertos no lo son tanto.

## **4.1 PORTABILIDAD**

Primero analizaremos que pasa con la PORTABILIDAD tan anunciada, y lo veremos en estos tres elementos:

### **DATOS-APLICACIONES-PERSONAL**

#### **4.1.1 DATOS-APLICACIONES**

Al investigar el mercado acerca de los manejadores de base de datos que se dicen "Abiertos", encontré que el estándar con el que se puede contar es con el SQL. Teóricamente si definimos los datos y las aplicaciones con SQL, obtendremos portabilidad, pero en la práctica es probable que existan aplicación que aparte de ser definidas con SQL, sea necesario hacer algunas rutinas específicas para lo cual deberá utilizarse ya sea un lenguaje propietario del manejador o incluso hacerlas en algún lenguaje de alto nivel que ofrezca más versatilidad.

Pongamos como ejemplo INFORMIX. Al hacer una aplicación en este manejador de base de datos, se pueden definir datos y aplicación utilizando SQL, pero en la práctica no hacemos totalmente la aplicación utilizando solo este estándar, para ello debemos hacer uso del 4GL de INFORMIX e incluso hacer rutinas en algún lenguaje como por ejemplo C. En estas condiciones, si por ejemplo deseo portar mi aplicación de un manejador a otro, podré llevarme el código fuente de lo que esté codificado en SQL puro. Digo SQL puro porque en el caso de ORACLE aparte de soportar el estándar SQL viene con el propio que le adiciona algunas otras características al producto, pero la parte que se genere con el SQL propietario ya no será portable porque tiene características propietarias de ORACLE.

Y a propósito de ORACLE, en este manejador encontramos que se dice "abierto" porque han fabricado su producto para que corra en diferentes plataformas y esto es bien cierto, pero, si trato de portar una aplicación generada en ORACLE hacia otro manejador, me encontraré con la misma limitación que en el caso anterior.

## **DATOS:**

Los datos en este caso, están en un formato propietario dependiendo el manejador que se utilice, así que si se desea portar datos de un ambiente a otro, éstos tendrán que ser convertidos a ASCII para poder transferirlos de un lugar a otro. Las diferentes herramientas cuentan con mecanismos para hacer lo anterior.

## **REALIDAD:**

El porcentaje de portabilidad de una aplicación a otra varía, dependiendo con qué código fue generada.

En algunos casos sólo se obtendrá portabilidad a nivel fuente.

No obtendremos portabilidad en formas o reportes generados con herramientas propietarias.

El único código que podrá ser portado es el SQL.

Los datos deberán ser convertirse a ASCII para que puedan ser portados.

## **ALTERNATIVA:**

- Para generación de aplicaciones deberá elegirse un manejador de base de datos relacional.
- El generador que se elija deberá soportar al menos el estandar que está disponible: SQL.
- Para rutinas especiales utilizar un lenguaje de alto nivel que esté estandarizado (COBOL, C, etc).

### **4.1.2 PERSONAL**

La portabilidad del personal cae dentro del componente U.- USER INTERFACE del modelo MUSIC y tenemos lo siguiente:

Las interfaces gráficas de usuario (GUI) deberían proporcionar las siguientes posibilidades:

- Uso extensivo de elementos de control visual estándares, como los iconos (icons) y las barras de desplazamiento (scroll bars). Los iconos son representaciones gráficas de objetos, por ejemplo, un cubo de basura puede significar borrado de un archivo. Las barras de desplazamiento permiten al usuario desplazar los datos que aparecen en una ventana.
- Manipulación directa de los elementos que hay en la pantalla. Por ejemplo, presionar un botón del ratón para modificar el margen de un documento hasta obtener el ancho deseado, en lugar de introducir un valor previamente calculado.
- Consistencia a través de aplicaciones y plataformas. Los usuarios aprenden una sola vez la forma de imprimir, borrar, copiar, etc., así que cuando conocen como llevar a cabo estas operaciones, podrán hacerlo en cualquier aplicación, independientemente cual sea ésta o en qué computadora esté trabajando.
- Aplicaciones múltiples simultáneas. Por ejemplo, un usuario puede editar un documento y al mismo tiempo volver a calcular una hoja de cálculo. Conforme la computadora recalcula la hoja de cálculo, la versión actualizada aparece automáticamente en el documento.
- Presentar un sistema atractivo y fácil de usar. Un usuario contento es más productivo. Idealmente los usuarios deberían poder personalizar su GUI para adaptarla mejor a su estilo de trabajo y no al revés.

Los sistemas X Window y las interfaces gráficas de usuario (GUI's) para microcomputadoras, tienen el mismo objetivo básico: facilitar el uso de la computadora. Todos proporcionan elementos gráficos para comunicarse con el usuario pero existen importantes diferencias entre los sistemas X Window y las GUI's basados en microcomputadoras.

#### REALIDAD:

Ninguna de las interfaces gráficas de usuario existentes actualmente cumplen todos los objetivos indicados anteriormente. Sin embargo algunas de las GUI's más usuales cumplen muchas de ellas.

#### ALTERNATIVA:

**Aún cuando la aplicación esté destinada a trabajar en una microcomputadora, se deberá elegir el soporte del Sistema X Window pues es el que actualmente ofrece ser un estándar "de jure" y que por ello garantiza más características de apertura que cualquiera otra. Ya sea que se elija Open Look o bien OSF/Motif, la decisión puede considerarse buena.**

## **4.2 INTEROPERABILIDAD**

Este elemento es uno de los más maduros, actualmente podemos encontrar conectividad e interoperabilidad entre diferentes plataformas. Lo importante es adquirir sistemas y equipos que soporten los estándares formales.

Un ejemplo de interoperabilidad es el que se refiere a que diferentes productos de correo electrónico que soporte estándares tales como MHS o SNADS, puedan interoperar de una forma armónica y flexible. No importa la plataforma ni el proveedor, tal es el caso de los siguientes productos:

<b>PRODUCTO</b>	<b>PROVEEDOR</b>	<b>PLATAFORMA.</b>
<b>MS-MAIL</b>	<b>MICROSOFT</b>	Red local Novell.
<b>MEMO-LAN</b>	<b>VERIMATION</b>	Red local Novell. LAN Manager.
<b>MEMO-HOST</b>	<b>VERIMATION</b>	Mainframe.

Por lo que respecta a la interoperabilidad todavía falta mucho pero es el elemento con el que ya podemos contar.



### 4.3 LOS ESTANDARES

Una de las causas por las que no todos los productos actuales pueden considerarse abiertos, es por no estar soportados sobre estándares formales y esto se debe a que no hay estándares que se ocupen de algunos temas, los que existen tocan algunos temas pero no de manera integral ni total y finalmente proliferan los productos con estándares propietarios.

Hay algunos productos que aún cuando se sujetan a estándares, no brindan una portabilidad completa.

La tecnología actual que aún cuando ya está en camino a pertenecer a los sistemas abiertos, todavía es muy inmadura.

Después de haber revisado los capítulos anteriores, es ya demasiado insistir en que los estándares son los que juegan el papel más importante en el campo de los sistemas abiertos, sin ellos no se tendría la menor posibilidad de llegar a cumplir las promesas que han establecido.

Conforme los componentes que ya hemos analizado con el modelo MUSIC se adhieran a los estándares, se estará en el camino correcto para alcanzar los objetivos de los sistemas abiertos.

La realidad es que para los proveedores el adoptar los estándares muchas veces significa potenciales limitaciones al producto por lo que solo adoptan los estándares comercialmente viables y que no necesariamente van a ser aquellos formulados por organismos internacionales dedicados a ellos.

**REALIDAD:** No todos los proveedores se sujetan a los estándares.

**ALTERNATIVA:** Adquirir productos que soporten los estándares formales que satisfagan el área que necesitemos.

#### 4.4 LOS MAINFRAMES

Los mainframe son un elemento clave en la tecnología que desde ya hace varios años ha venido y todavía viene produciendo los servicios informáticos (En empresas grandes, por supuesto). El grado en que el mainframe dé dichos servicios, está en función de la magnitud y tipo de empresa. Considero que el mainframe es todavía una pieza que no podemos desechar tan fácilmente de nuestros activos de información y no podía pasar por alto su situación ante el tema que nos ocupa.

##### REALIDAD:

Las empresas no pueden desechar la tecnología con la que hoy cuentan, más bien, deberán sacarle el mayor provecho posible. Deberá buscarse la manera de adaptarse a los cambios provocados por los sistemas abiertos sin perder la inversión hecha anteriormente y sobra decir que si se cambia, el cambio deberá ser para mejorar.

Al plantear lo que son los sistemas abiertos y ver lo que ofrece hoy el mercado, debemos hacernos las siguientes preguntas:

¿Me va a beneficiar el cambio?

¿Mejor ni le movemos?

¿Espero a que la tecnología madure?

¿Lo que me va a costar vale la pena?

El papel que ahora desempeñarán los mainframes podrá ser de grandes servidores de archivos o de comunicaciones. Proveedores como IBM aún cuando se alinean a diversos estándares formales, todavía no han ofrecido al cliente una solución abierta completa, sucede lo mismo con todos, solo algunas partes de sus productos reúnen características de apertura pero todavía siguen manteniendo sus estándares propietarios.

##### ALTERNATIVA:

Los sistemas que hoy están en el mainframe y que sean candidatos a correr en red, podrán ser implementados en esta nueva plataforma, la decisión deberá basarse en el tamaño, complejidad y cobertura de servicio.

Hay aplicaciones que por sus características tendrán que permanecer en el mainframe, ahí se quedarán.

Con esta estrategia sobra decir que de momento no será posible sacar al mainframe de su lugar, pues las características de algunos negocios requieren de este tipo de computadoras todavía.

Si se desea encaminarse hacia los sistemas abiertos, las aplicaciones tendrán que rehacerse pues las que actualmente están en el mainframe fueron generadas en herramientas que no se sujetan a estándares formales, así que en la mayoría de los casos no podemos hablar de "migración" hacia sistemas abiertos, aquí la alternativa única hasta hoy es hacer una generación completa de las aplicaciones.

#### **4.5 COMPUTADORAS PERSONALES**

Tanto el mainframe como las computadoras personales con MS-DOS, son parte de la realidad de nuestros días y, hoy por hoy, ambos elementos aun cuando no cumplen con los requisitos para formar parte de los sistemas abiertos, son activos imprescindibles de las empresas.

##### **REALIDAD:**

Aquí nos encontramos ante una realidad tajante:

- El sistema operativo DOS puede correr en infinidad de máquinas provenientes de diferentes fabricantes, con ésto nosotros podemos tener independencia de proveedor.
- Las aplicaciones que corren bajo DOS, ya sea WINDOWS, DBIII, EXCEL, MS-MAIL, etc., pueden correr en cualquier computadora personal. Aquí tenemos portabilidad de las aplicaciones.
- El personal que trabaja en una computadora personal determinada, podrá sin ningún esfuerzo, trabajar con otra de un proveedor diferente, aquí tenemos portabilidad del personal.
- La interoperabilidad también se da entre PC's al poder conectarse entre ellas por medio de diferentes sistemas operativos de red: Netware, Lan Manager, Etc.

El requisito que no cumple el ambiente de computadoras personales con sistema operativo DOS es que no se sujeta a estándares propuestos por los consorcios internacionales que se dedican a ello.

##### **REALIDAD:**

Actualmente los usuarios consideran a las computadoras personales como las más estándares y son las que prefieren, pero de acuerdo a la teoría que vimos en el capítulo 2, el ambiente MS-DOS no es considerado abierto.

Las computadoras personales tampoco podrán desecharse tan fácilmente, por fortuna el sistema operativo considerado abierto, que es el UNIX, también lo podemos encontrar en computadoras personales, aunque hoy profiera el MS-DOS, a medida que los productos de sistemas abiertos maduren, se espera que también se encuentren en muchas modalidades y estén ampliamente disponibles.

**ALTERNATIVA:**

Las computadoras personales con MS-DOS seguirán en nuestros escritorios y tendremos que esperar a que productos tan útiles y atractivos al usuario como el WINDOWS, sean fabricados bajo estándares formales o que en UNIX tengamos herramientas tan atractivas como WINDOWS..

#### **4.6 PROVEEDORES**

Por muchos años hemos tenido que sufrir con el mismo proveedor. Si tenemos un mainframe IBM, solo con ellos podremos ver todo lo concierne a nuestros sistemas: Hardware, software, aplicaciones, mantenimiento, soporte, capacitación, etc.

Si se trata de UNIX vemos que hay dependencia del hardware, Ejemplo:

<b>EQUIPO</b>	<b>UNIX.</b>
SUN	SOLARIS.
RS/6000 IBM	AIX
SILICON GRAPHICS	IRIX.
S/390 IBM	AIX/ESA

Aunque UNIX teóricamente es independiente del proveedor, algunos servicios y códigos dependen directamente del hardware, luego entonces, también del proveedor.

Las aplicaciones implementadas en un UNIX específico, tienen dependencias del hardware.

#### **REALIDAD:**

El UNIX todavía no es completamente abierto y todavía depende del proveedor.

#### **ALTERNATIVA:**

Aunque la realidad nos indica que el UNIX todavía no cumple con las promesas de un sistema abierto, es el único que tiene características de llegar a serlo, lo que debemos hacer es adquirir este sistema y esperar a que maduren los estándares formales que permitan al usuario no depender de un solo proveedor.

#### **4.7 ESTRUCTURA ORGANIZACIONAL PARA LOS SISTEMAS ABIERTOS**

El día de hoy estamos viendo que con el auge de las redes y la facilidad que hasta cierto punto algunos productos brindan al usuario para manipular de manera independiente sus sistemas, el esquema del gran centro de cómputo proveedor de todos los servicios informáticos está compartiendo el lugar que hoy tienen ya los sistemas en el escritorio del propio usuario.

La estructura que ahora tendrán las organizaciones tendrá mucho que ver con la tecnología pues es con ella con la que se puede ser más productivo y competente en los diversos mercados.

El papel que desempeñan actualmente los elementos que integran el Departamento de Sistemas tiene que ver también con la tecnología, antes se podía encontrar a 100 programadores de Cobol en un Departamento, pero ahora tendrán que:

- Vencer la resistencia al cambio y adentrarse en la nueva tecnología.
- Capacitarse en la tecnología que ofrecen los actuales sistemas abiertos.
- Aceptar el nuevo papel que deberán desempeñar dentro de la estructura organizacional actual.

Decir y escribir lo anterior es muy fácil, pero nos encontramos ante un gran reto por lo que respecta a la parte humana, para que los sistemas abiertos sean exitosos, se debe contar con el personal adecuado y convencido de la nueva tecnología.

Esta tesis no tiene como objetivo adentrarse en terrenos pertenecientes los especialistas en la conducta humana o relaciones laborales, lo que si vale la pena mencionar, es que para que los sistemas abiertos sean herramientas de provecho para las empresas, éstas deberán cambiar su estructura organizacional para sacarles el mayor jugo y si ésto no convence, solo tienen que revisar la evolución de la estructura organizacional del área de sistemas que se plantea en el primer capítulo para darse cuenta de que estoy en lo cierto.

En el capítulo de metodología aplicada a un ejemplo práctico, plantearemos una posible estructura organizacional y será interesante ver los cambios a que debemos enfrentarnos.

#### **4.7 ESTRUCTURA ORGANIZACIONAL PARA LOS SISTEMAS ABIERTOS**

El día de hoy estamos viendo que con el auge de las redes y la facilidad que hasta cierto punto algunos productos brindan al usuario para manipular de manera independiente sus sistemas, el esquema del gran centro de cómputo proveedor de todos los servicios informáticos está compartiendo el lugar que hoy tienen ya los sistemas en el escritorio del propio usuario.

La estructura que ahora tendrán las organizaciones tendrá mucho que ver con la tecnología pues es con ella con la que se puede ser más productivo y competente en los diversos mercados.

El papel que desempeñan actualmente los elementos que integran el Departamento de Sistemas tiene que ver también con la tecnología, antes se podía encontrar a 100 programadores de Cobol en un Departamento, pero ahora tendrán que:

- Vencer la resistencia al cambio y adentrarse en la nueva tecnología.
- Capacitarse en la tecnología que ofrecen los actuales sistemas abiertos.
- Aceptar el nuevo papel que deberán desempeñar dentro de la estructura organizacional actual.

Decir y escribir lo anterior es muy fácil, pero nos encontramos ante un gran reto por lo que respecta a la parte humana, para que los sistemas abiertos sean exitosos, se debe contar con el personal adecuado y convencido de la nueva tecnología.

Esta tesis no tiene como objetivo adentrarse en terrenos pertenecientes los especialistas en la conducta humana o relaciones laborales, lo que si vale la pena mencionar, es que para que los sistemas abiertos sean herramientas de provecho para las empresas, éstas deberán cambiar su estructura organizacional para sacarles el mayor jugo y si esto no convence, solo tienen que revisar la evolución de la estructura organizacional del área de sistemas que se plantea en el primer capítulo para darse cuenta de que estoy en lo cierto.

En el capítulo de metodología aplicada a un ejemplo práctico, plantearemos una posible estructura organizacional y será interesante ver los cambios a que debemos enfrentarnos.



#### **4.3 CONCLUSION**

Ya vimos que los elementos que intervienen en los sistemas abiertos no cumplen en su mayoría con las expectativas que el común de la gente tiene al querer lograr un ambiente de este tipo. La teoría nos demostró que no es nada trivial hablar de las características que deben reunir dichos elementos para conformar este ambiente tan deseado y también nos dimos cuenta que en el mercado no existen productos que realmente formen parte de este mundo. Por lo que concluimos que actualmente no existen los sistemas abiertos.

Lo que sí es cierto, es que la mayoría de los proveedores ya se dieron cuenta que existe una gran competencia por ser el que brinde las características de apertura más aceptables y podemos decir que con esto, nos encontramos en el punto inicial de un camino muy largo que se debe de recorrer para llegar a esa meta final: LOS SISTEMAS ABIERTOS.

Aquí lo importante es que al menos, ya encontramos el camino. Por lo que respecta a nosotros los usuarios, la alternativa que tenemos ante esta realidad es:

- Investigar a fondo todas las características de los productos antes de decidir que comprar.
- Ver diferentes alternativas tanto de equipo como de proveedor.
- Adquirir productos que manejen estándares formales.
- Esperar a que la tecnología madure.

## **CAPITULO 5**

# **METODOLOGIA APLICADA A UN EJEMPLO PRACTICO**

## **CAPITULO 5**

### ***METODOLOGIA APLICADA A UN EJEMPLO PRACTICO***

Al momento de querer desarrollar un sistema abierto específico para nuestra empresa, nos encontramos con que no es nada sencilla esta tarea, así que representa un gran reto para el que quiere llevar a cabo este proyecto.

Ya vimos qué conceptos tienen que ver con los sistemas abiertos y al menos, por esa parte, ya sabemos exactamente de qué magnitud será nuestra responsabilidad. También analizamos la situación y nos dimos cuenta de las realidades de los sistemas abiertos y de las alternativas que tenemos.

El ambiente de cómputo actual, como lo comenté al principio de este documento, presenta un panorama bastante heterogéneo y difícil de analizar, pues tendremos que tratar con:

- Diferentes plataformas de cómputo (Diferente hardware y sistemas operativos).
- Tratar con bases de datos diferentes e independientes.
- Contar con datos en diferentes formatos ubicados en diferentes plataformas, así como en diferente lugar geográfico.
- El personal informático actual, está familiarizado con conocimientos solamente de alguna parte de este ambiente.

En la parte de definición de requerimientos de los sistemas, así como en las especificaciones, los desarrolladores deben identificar:

- Los componentes que serán incluidos en el ambiente de sistemas abiertos.
- Componentes que operarán de modo aislado.
- Recursos existentes que serán reemplazados o reescritos.

También habrá factores críticos que influyen fuertemente en la elección, como por ejemplo:

- Tener que incluir hardware y software existente (inversión anterior).

- Considerar aplicaciones críticas.
- Necesidades y demandas específicas de los usuarios.

En varias fuentes (incluyendo muchas revistas) he visto que se plantean diferentes formas en las que se puede abordar el problema, la mayoría coinciden en algunos puntos, pero encontré una metodología en el libro *Open Systems Handbook: A Guide to Building Open Systems*<sup>1</sup>, que me pareció muy acertada, pues ésta se enfoca precisamente al desarrollo de los sistemas abiertos, así que me basaré en los pasos ahí señalados para aplicarlos a un ejemplo práctico, que servirá de ejercicio.

---

<sup>1</sup> OPEN SYSTEMS.- Gary J. Nutt.- University of Colorado, PTR Prentice Hall, Englewood Cliffs New Jersey, 1992.

## **5.1 METODOLOGIA**

A continuación presento los puntos que deberán considerarse en esta metodología:

### **PRIMERO.- CONOCER LOS OBJETIVOS Y PLANES DEL NEGOCIO**

Es de vital importancia conocer el OBJETIVO y los planes de la compañía y sobre decir que los responsables de la administración de la tecnología de la información, tienen doble obligación de conocerlos.

Si se desea implantar sistemas abiertos en una empresa, lo primero que deben tener claro los responsables de esta tarea, es conocer plenamente el objetivo del negocio.

Para ser más competitivos en los negocios han surgido herramientas muy importantes, entre las cuales está la tecnología de la información, pero para que ésta sea utilizada eficientemente y se le saque provecho, deberá ser enfocada a la consecución de los objetivos del negocio.

Así que como primera tarea debemos conocer a fondo los objetivos del negocio de nuestra compañía, debemos tener conocimiento pleno del área a la que se desea favorecer (si es solamente un área del negocio) o si se desea trabajar con la totalidad del negocio.

Es probable que solo se desee aplicar los sistemas abiertos a un área de interés, así que deberá ser esa la que debemos analizar y comprender.

Como en el manejo de cualquier otro proyecto, factores como restricciones y criterios de éxito también deberán considerarse, y éstos son muy particulares de cada ambiente, por lo que deberá implementarse una administración de proyecto muy estricta.

Es necesario también conocer los planes del negocio, no se pueden tomar decisiones a la ligera si no se conocen los planes o tendencias que se tengan a nivel negocio, es probable que si estamos parcialmente enterados de este punto, no tomemos las decisiones correctas al poner herramientas inadecuadas en las manos de los demás.

## SEGUNDO.- ANALISIS DEL AMBIENTE ACTUAL

En este paso es de gran importancia saber qué tenemos, no podemos partir hacia ningún punto, si no sabemos en qué lugar nos encontramos.

En estos tiempos, toda empresa cuenta ya con activos informáticos, es decir:

- Equipo de Cómputo.- Diferentes plataformas y marcas.
- Sistemas y aplicaciones.
- Datos en diferentes formatos.
- Recursos humanos con experiencia en la tecnología que tiene.

De por sí un ejemplo como el que se plantea en este trabajo puede sonar muy teórico, sería mucho más hipotético dar un ejemplo partiendo de una empresa en la que todo se comprará y hará nuevo.

Aquí debemos dejar plasmado todo lo relativo a los recursos con los que se cuenta, para ello debemos hacer una investigación exhaustiva no solo en el área de sistemas, debemos analizar todas las áreas de la organización ya que gracias a la computación personal, el día de hoy existen como pequeñas islas, diferentes aplicaciones generadas en diversas herramientas. Como producto final de este paso, debemos obtener un documento que contenga toda esta información, pues continuamente haremos referencia a ella.

Dicho documento debe tener información respecto a:

- Las aplicaciones que existen en toda la organización, de que tipo son, en que plataforma se encuentran y si presentan alguna problemática específica o requerimiento especial.
- Las aplicaciones que se deben modificar.
- Las aplicaciones que deben crearse por nuevas necesidades o necesidades no atendidas.
- Las aplicaciones que deben rehacerse o convertirse a otra herramienta o plataforma.
- Las aplicaciones que deben eliminarse.
- La compatibilidad o escalabilidad del hardware de las diferentes plataformas.

### **TERCERO.- IDENTIFICAR LOS REQUERIMIENTOS DE LAS APLICACIONES**

Dependiendo de la naturaleza de la aplicación habrán diferentes requerimientos para cada una, en algunos casos éstos serán comunes pero en general, cada una tiene algunos que son específicos.

Algunos elementos que son generales y que siempre son considerados son: desempeño, seguridad, capacidad de recuperación, etc.

Algunos elementos que sí tienen que ver con sistemas abiertos son:

- Los que afectan la portabilidad de las aplicaciones.
- Los que tienen que ver con la interacción humana.
- El acceso a la información.
- Interoperabilidad.

Los requisitos de las aplicaciones en este sentido, deben ser considerados y especificados en términos de estándares para preservar, reforzar y, de alguna manera, obligar a que se ajusten al concepto de sistemas abiertos.

### **CUARTO.- PREPARAR UN PERFIL**

Este paso es necesario para que se identifique el juego de capacidades necesarias que deberán reunir las aplicaciones.

Un perfil de aplicación describe las facilidades requeridas para que una aplicación sea ejecutada. El perfil deberá identificar características tanto estándares como no estándares de tal forma que todos los requerimientos sean considerados en la selección de la plataforma y en el desarrollo o la compra del software, según sea el caso.

Un perfil para ambiente de escritorio, probablemente abarque una gran variedad de aplicaciones. Con un perfil común amplio, el esfuerzo invertido una vez para desarrollar tal perfil, puede producir valor creciente conforme se seleccionen aplicaciones múltiples.

También enfocándose sobre aplicaciones coexistentes en un perfil común puede lograrse que el trabajo básico mejore la portabilidad e interoperabilidad.

Si tenemos la necesidad o deseo de contar con un ambiente de sistemas abiertos, nosotros estamos hablando de que queremos lograr objetivos tales como portabilidad, interoperabilidad y consistencia con la interface de usuario. Pero con estos objetivos no se puede asumir que sean los únicos en los cuales estamos interesados, ni podemos asumir que ellos sean completamente alcanzables dado que pueden entrar en conflicto con otros objetivos que también queremos alcanzar, tales como un cierto nivel de rendimiento o seguridad.

Para desarrollar un perfil, hay cientos de estándares base y especificaciones abiertas de las cuales debemos seleccionar. En la mayoría de los casos, estándares individuales se enfocan a un área particular de alcance limitado ya sea en portabilidad o interoperabilidad. Muchos de los documentos base desarrollados en años son limitados en su alcance y, al mismo tiempo, ampliamente aplicables, por ejemplo los estándares para los lenguajes.

En muchos casos, existe una relación entre la selección de los estándares y la aplicación, por ejemplo si es una del ambiente de finanzas, el lenguaje apropiado estándar es el COBOL, mientras que el FORTRAN o C pueden ser especificados para aplicaciones científicas o de ingeniería (En la realidad, existen muchas aplicaciones administrativas que ya utilizan el lenguaje C como su lenguaje estándar).

En cualquiera de los casos, el lenguaje estándar deberá ser seleccionado como parte de su perfil, debido a la relación histórica que existe entre la aplicación y el lenguaje.

Dado que en general las aplicaciones financieras están escritas en COBOL, puede ser deseable utilizar el COBOL para mantener la continuidad histórica dado que existe gran experiencia, software y plataformas para soportar esta elección. Sin embargo, otros lenguajes estándar pueden ser elegidos para desarrollar nuevas aplicaciones.

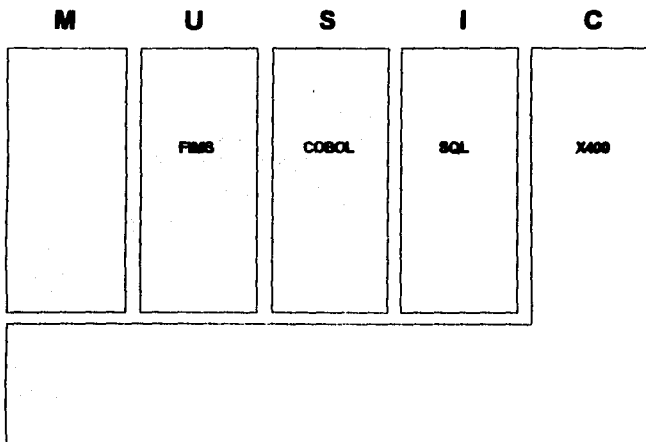
El perfil que se prepare nos brindará una descripción estructurada y específica de la aplicación deseada, así como un enfoque de lo que se puede hacer para implementarla.



Los perfiles ayudan tanto a usuarios como proveedores para concordar las aplicaciones con las necesidades del usuario. La existencia de un perfil da al proveedor clara idea acerca de lo que debe construir en sus productos para satisfacer los requerimientos del usuario. Así mismo, permite a los usuarios seleccionar soluciones para las aplicaciones que van a estar de acuerdo con sus especificaciones y que por lo tanto, deberán satisfacer sus objetivos.

Aquí están varios ejemplos simplificados de perfiles de aplicación los cuales pueden ser generados con normas especificadas. En estos ejemplos, los estándares están clasificados de acuerdo a los elementos del tan mencionado modelo MUSIC.

**Ejemplo 1.-**

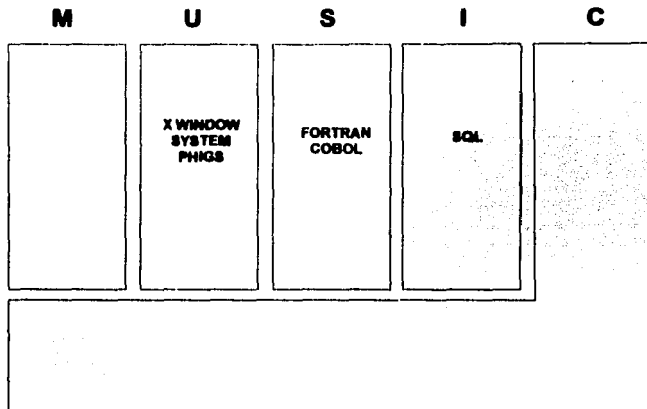


**FIGURA NUM. 29.- APLICACIÓN DE TIPO CONTABLE**

Para esta aplicación, el perfil puede incluir los siguientes estándares:

Form Integration Management System (FIMS) como estándar del elemento U para formato, despliegue e intercambio de formas; COBOL como estándar del elemento S. COBOL es elegido como lenguaje estándar debido a las ventajas mencionadas anteriormente. SQL se incluye como estándar del elemento I para acceso a base de datos y, finalmente, a X.400 como estándar de mensajes en el elemento C para comunicaciones.

**Ejemplo 2.-**



**FIGURA NUM. 30.- APLICACIÓN INGENIERIL CAD/CAM**

En esta aplicación el perfil puede incluir al Sistema X Window (el cual no es un estándar formal pero es el que actualmente se acepta ampliamente como estándar de la industria) para el elemento U, Fortran y COBOL como estándares del elemento S, PHIGS para gráficas tridimensionales como un estándar del elemento S y SQL para acceso a base de dato como el estándar dentro del elemento I.

## **QUINTO.- ADQUISICION O DESARROLLO DE APLICACIONES**

Se adquirirá o fabricará el software que se adapte a su perfil. Comprar o desarrollar sistemas o componentes que cumplan con los estándares y servicios requeridos por los perfiles, es una parte del proceso, necesaria pero no suficiente.

Las aplicaciones deben ser construidas tomando como referencia las capacidades perfiladas en el tercer paso porque si cuentan con características diferentes, la portabilidad y la interoperabilidad para el futuro en el mejor de los casos es limitada y, en el peor, se pierde.

Utilizando estándares en el proceso de obtención de software, ayudará a calificar productos existentes y a clarificar las ideas de tener que hacer o tener que comprar. Para maximizar las características de un sistema abierto, las especificaciones abiertas y estándares formales deberán siempre ser considerados primero. Como ya vimos anteriormente, si ellos no proporcionan las capacidades necesarias requeridas, entonces se pueden incluir funciones no estandarizadas.

Un punto importante es el tipo de código ya que está directamente vinculado con la portabilidad de las aplicaciones, así que hay que decidir qué tipo se necesita.

Aún cuando una aplicación garantice el uso de capacidades de un vendedor específico, también es importante considerar el buen soporte e independencia del proveedor ya que mejorará la evolución futura del sistema y la posible migración de aplicaciones.

## **SEXTO.- VERIFICAR LAS CARACTERISTICAS DE SISTEMAS ABIERTOS DE LA APLICACION**

Hay dos puntos importantes al momento de verificar una aplicación de sistemas abiertos: La primera es la documentación que describe el ambiente requerido y la segunda las características de interoperabilidad de la misma.

La verificación deberá llevarse a cabo haciendo una comparación con el perfil predefinido.

Una parte esencial de la verificación es la ejecución de la aplicación sobre un sistema cuya implementación ha sido conforme a un perfil. En algunos casos puede ser útil probar la aplicación sobre 2 diferentes sistemas configurados

con algunas variantes para probar la portabilidad y corregir cualquier desviación en la ejecución mientras se tenga al experto a la mano (En caso de que sea software comprado).

**Prueba de conformidad de las aplicaciones.-** Desde el punto de vista del usuario, la verdadera prueba de conformidad con estándares de una aplicación es aquella en la que la aplicación hace lo que se supone debe hacer en un ambiente específico. Pero dado que las organizaciones tendrán diferentes ambientes, las pruebas de conformidad deben estar basadas en un criterio más generalizado.

Con una aplicación que esté conforme a estándares que están estrictamente definidos por POSIX u otra organización formal de estándares, la prueba de conformidad de la aplicación es posible.

#### **SEPTIMO.- VERIFICACION DE OBJETIVOS**

Tal como en el primer paso se identificaron los objetivos del negocio, para cerrar el ciclo apropiadamente, el último paso deberá ser la confirmación de que esos objetivos han sido alcanzados.

Hasta aquí todo parece indicar que hemos terminado con los pasos necesarios para implementar un ambiente de sistemas abiertos, pero la realidad es diferente. En el ambiente de los negocios así como en la tecnología, todo es cambiante y dinámico, así que es muy probable que cuando se llegue a este punto, ya hayan surgido nuevas necesidades, nuevos retos y que en el mercado ya tengamos disponible nueva tecnología que nos haga más competitivos, es por eso que existe otro punto:

#### **OCTAVO.- REPETIR LA SECUENCIA**

Es necesario mantener un ciclo como el antes descrito para tener a la tecnología de la información como una herramienta útil para la consecución de los objetivos del negocio.

También hay que pensar que en este punto ya se experimentó con este tipo de implementaciones, así que se tendrá una experiencia diferente de la cual deberemos disponer para hacer este trabajo mejor.

Conforme se repitan los pasos en el proceso de desarrollo, el perfil existente aclarará como puede ser aplicada la tecnología nueva o mejorada y surgirá un entendimiento claro de las habilidades que se necesitan para adaptar las aplicaciones.

## 5.2 FACTOR HUMANO

Hay una consideración muy importante que debemos hacer al momento de desarrollar nuestras aplicaciones y es el factor humano.

Los pasos aquí esbozados, describen un proceso para identificar y obtener aplicaciones y sistemas abiertos que proporcionarán la funcionalidad que se necesita. En cualquier empresa que se aplique, forzosamente se tiene que hablar del factor humano que ésto implica:

- Identificar áreas de responsabilidad.
- Superar territorialidad.
- Inercia y resistencia al cambio.
- Desarrollo de espíritu de equipo.
- Impartir capacitación en la nueva tecnología.
- Cambiar la estructura organizacional.

Sólo por mencionar algunos.

Claramente la tecnología, aún con la interface de usuario consistente e íntegra que prometen los sistemas abiertos, es poco para encaminar a la gente en los temas que son parte de cualquier cambio. El éxito de un ambiente de sistemas abiertos dependerá mucho de como se dirijan estos temas, así como de la forma en que se implemente la tecnología elegida.

Lo que si es obligado y deberá hacerse en forma paralela a los pasos antes señalados, es la modificación paulatina de la estructura organizacional del área informática. Dicha modificación deberá obedecer a la forma en que se vaya dando el servicio con la nueva tecnología.

Ya no podrá trabajarse con estructuras jerárquicas tan rígidas, lo más indicado será que el trabajo se lleve a cabo en diferentes grupos y que éstos no estén inmersos en diferentes niveles jerárquicos, más bien se deberá optar por una estructura plana de pequeños grupos que a su vez se relacionen entre sí al mismo nivel para que de ese modo, se aproveche óptimamente la tecnología adquirida.

### **5.3 EJEMPLO**

#### **5.3.1.- OBJETIVO**

Sea un Banco mediano que proporciona servicios financieros diversos, pero que el objetivo principal del mismo es el manejo de créditos.

#### **5.3.2.- ANALISIS DEL ENTORNO**

Los servicios que brinda este banco tienen cobertura nacional, es decir, tiene sucursales en los diferentes Estados de la República.

Es una empresa en la que se ha adquirido tecnología en diversas épocas y por lo tanto cuenta con diferentes equipos de cómputo. Dichos equipos los agruparemos en: Sistemas mayores, medianos y sistemas personales.

#### **SISTEMAS MAYORES.**

Equipo:

En esta plataforma el servicio informático está centralizado en un sitio en la ciudad de México, D.F., es un clásico Centro de Cómputo y está integrado de:

- Un mainframe marca IBM modelo 9121 y sus correspondientes periféricos.
- Dos Sistemas Operativos VSE/ESA (la cpu está lógicamente partida en dos).
- Red SNA a nivel República Mexicana, con enlaces por línea privada.
- Manejador de Base de Datos IDMS tipo red (De Computer Associates).
- Generador de aplicaciones con manejo de archivos convencionales CSP (Cross System Product) que trabaja bajo CICS.

Aplicaciones:

- . Sistema de Créditos a la industria. En IDMS.
- . Sistema de Créditos al transporte. En IDMS.
- . Sistema Integral de Administración de Personal. En IDMS.
- . Sistema de Contabilidad. En CSP.
- . Sistema de Presupuesto y Control. En IDMS.
- . Sistema de Mobiliario y Equipo. En IDMS.

Cobertura: A nivel nacional vía red SNA.

**Recursos Humanos:**

Para dar soporte a esta plataforma, se cuenta con un área que dentro de la estructura organizacional se encuentra a nivel Gerencia, los elementos que la componen son:

- . 20 programadores de Cobol e IDMS.
- . 10 Analistas de Sistemas.
- . 3 Subgerentes (De Desarrollo, de Soporte Técnico y de Operación).
- . 8 Operadores del mainframe.
- . 3 Supervisores de Cintoteca y Control de la Producción.

**Problemática:**

- Costos de operación muy caros.
- En esta plataforma no se cuenta con herramientas amigables al programador para el desarrollo rápido de aplicaciones.
- Todas las aplicaciones se encuentran en un ambiente propietario.
- Se depende de un solo proveedor: IBM.
- Los datos se encuentran en diferente formato dependiendo el generador de la aplicación.
- Tiempo excesivo para el desarrollo de aplicaciones.
- Si se quiere migrar sólo se podrá hacer a un sistema operativo propietario de IBM.

**SISTEMAS MEDIANOS.**

En este rango se encuentran servidores diseminados en diferentes oficinas del área metropolitana y en las sucursales de la República Mexicana.

**Equipo:**

- Servidores marca ALTOS 1000
- Sistema Operativo UNIX System V.
- Manejador de base de datos Informix.



- El software de red de algunas redes Ethernet es PC CONNECTION KIT (de ALTOS). Son redes LAN en 31 Estados y 3 Dptos. en oficinas centrales.
- Software de red para enlace al mainframe es SNA/3270 (de ALTOS).

**Aplicaciones:**

- . Sistema de Créditos a la Vivienda.
- . Sistema Mesa de Dinero.
- . Sistema de Contabilidad local en sucursales foráneas.

Cobertura: Departamental.

**Recursos Humanos:**

En oficinas centrales se cuenta con un grupo de 4 programadores que desarrollaron los sistemas anteriormente señalados. En cada Estado se encuentra un especialista en informática que da soporte a la red local. Un subgerente del área.

**Problemática:**

- El equipo es obsoleto.
- No se cuenta con soporte del proveedor porque éste ya desapareció del mercado.
- Para esta plataforma no hay en el mercado productos compatibles que hagan más amigable el desarrollo de sistemas.

**SISTEMAS PERSONALES.**

En este grupo hablamos de las computadoras personales que no están en red.

**Equipo:**

- Computadoras personales de diferentes marcas.
- Sistema Operativo DOS.
- Herramientas de escritorio de diferentes marcas, entre las que se encuentran procesadores de palabra, hojas de cálculo, manejadores de bases de datos o tablas, para presentaciones, etc.

**Aplicaciones:**

Son diversas y están en su mayoría elaboradas en DBIII o en hojas de cálculo.

**Cobertura:** Por usuario y por Departamento (hoy no están en red).

**Recursos Humanos:**

Cada usuario es el administrador y operador de su computadora personal. Existen aproximadamente 1500 PC's en toda la organización.

**Problemática:**

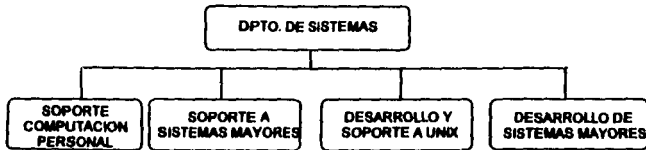
Sólo las computadoras en sucursales foráneas están en red con el producto de Altos denominado PC Connection Kit en la PC y Atlantes en el servidor, el cual es un producto propietario de Altos. No obstante estar en red, no existe aplicación alguna que funcione en red.

En el siguiente cuadro se resume el ambiente actual:

APLICACIONES	EQUIPO	UBICACION	SIST.OP.	GENERADO EN	COBERTURA
Créditos a la Industria	IBM-9121	México, D.F.	VSE/esa	IDMS, COBOL	Nacional
Administración de Personal	IBM-9121	México, D.F.	VSE/esa	IDMS, COBOL	Nacional
Contabilidad	IBM-9121	México, D.F.	VSE/esa	CSP, COBOL	Nacional
Presupuesto y Control	IBM-9121	México, D.F.	VSE/esa	IDMS, COBOL	Nacional
Mobiliario y Equipo	IBM-9121	México, D.F.	VSE/esa	IDMS, COBOL	Nacional
Créditos a la vivienda	Serv ALT OS-1000	Todos los Estados	UNIX S.V.	Informix, C	Departamental
Mesa de dinero	Serv ALT OS-1000	México, D.F.	UNIX S.V.	Informix, C	Departamental
Contabilidad Estatal	Serv ALT OS-1000	México, D.F.	UNIX S.V.	FoxBase, C	Departamental
Diversas	PC'S	Toda la Organiz.	MS-DOS	Excel, DBIII, etc.	Por usuario y Departamental
Créditos al transporte	IBM-9121	México, D.F.	VSE/ESA	IDMS, COBOL	Nacional

**FIG.NUM. 31.- AMBIENTE ACTUAL**

Por lo que el organigrama del área informática es el siguiente:



**Funciones generales:**

**SOPORTE A MICROS.**

- Control e inventario de equipo.
- Instalación de hardware y software (sistema operativo y paquetería).
- Soporte al software y a los usuarios.

**SOPORTE A SISTEMAS MAYORES.**

- Administración de sistema operativo, software base y de comunicaciones.
- Monitoreo y mantenimiento a red SNA.
- Soporte a usuarios de red SNA.

**DESARROLLO EN SISTEMAS MAYORES.**

- Análisis, desarrollo, mantenimiento y soporte a las aplicaciones en sistemas mayores.

**DESARROLLO Y SOPORTE EN UNIX.**

- Análisis, desarrollo, mantenimiento y soporte a las aplicaciones en UNIX.

### **5.3.3.- IDENTIFICAR LOS REQUERIMIENTOS DE LAS APLICACIONES**

De acuerdo a los objetivos de esta organización, los créditos son el elemento medular de la operación, por lo que en nuestro ejemplo trabajaremos sobre el sistema de créditos.

El sistema de créditos como se puede observar, está diseminado en diferentes maquinass:

**Sistema de Créditos a la Industria.-** Está en la partición 1 del procesador IBM 9121 en IDMS.

**Sistema de Créditos al Transporte.-** Está en la partición 2 del procesador IBM 9121 en IDMS.

**Sistema de Créditos a la Vivienda.-** En Informix en un servidor ALTOS en México y en cada Sucursal foránea.

En particular la problemática de este sistema es que por un lado es necesario que se descentralice la operación de los créditos ya que el volumen alto de los mismos se concentra en el sistema de créditos a la Industria y por lo tanto, en un solo procesador lógico (demasiada carga), pero por el otro, es necesario consolidar diariamente en la ciudad de México, las operaciones de todas las plazas.

Tomando en consideración lo anterior, los requerimientos de esta aplicación son los siguientes:

- El sistema de créditos deberá ser único, es decir, deberá manejar cualquier tipo de crédito.
- Durante el día, los datos podrán ser manejados en forma independiente y distribuidos en las diferentes plazas para una operación normal, pero en el momento de requerir autorización de desembolso o de que se haga un pago en una plaza diferente a aquella en donde no se haya contratado originalmente el crédito, será necesario hacer una consulta a datos centralizados.
- Al final del día todos los datos deben estar concentrados en la ciudad de México con objeto de control y consolidación.
- Desde cualquier plaza y en horas hábiles, deberá tenerse acceso a la información centralizada.

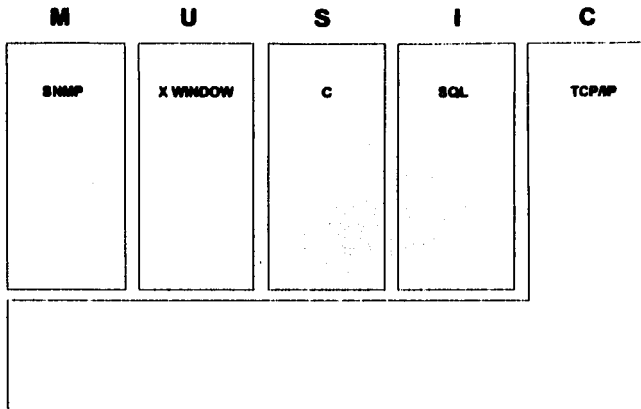
- La interface con el usuario deberá ser amigable y única en cualquier plataforma, con objeto de que la operación sea ágil y libre de errores.
- El desempeño y la seguridad adecuados son elementos que, como cualquier otro sistema, deberá tener la aplicación con el fin de garantizar un buen servicio.
- Deberá contar con mecanismos eficientes de respaldo y recuperación.
- La aplicación deberá estar generada en una herramienta que pueda trabajar en los diferentes ambientes con que se cuenta con el fin de aprovechar la inversión que se ha hecho en equipo anteriormente, es decir, evitar al máximo tener que gastar en nuevo equipo.
- Si se adquiere nuevo equipo, éste deberá ser lo suficientemente compatible con los que ya se tienen.
- La herramienta para generar, operar la aplicación y manipular los datos, deberá soportar estándares formales tales como:

SQL, TCP/IP, Lenguajes estándar, Etc.

#### **5.3.4.- PREPARAR UN PERFIL**

Para facilitar la tarea, nos alinearemos al modelo MUSIC para la elaboración del perfil de esta aplicación.

Considerando los requerimientos de la aplicación, los elementos estandarizados y los no estandarizados disponibles en el mercado manejaremos la siguiente opción:



Si estamos pensando en un sistema que deberá estar disponible a nivel nacional, estamos hablando de una red y, para administrar la misma, es necesario que empecemos a trabajar con elementos estandarizados, es por eso que, cualesquiera que sea la marca del hardware que se compre (ruteadores, puentes, compuertas, etc.), éste deberá soportar el estándar SNMP para lograr una buena administración. SunNet Manager o Netview por ejemplo, son administradores que brindan soporte SNMP. Se optará por Netview, ya que éste tiene la capacidad de administrar tanto redes TCP/IP como SNA.

La interface de usuario que elegiremos, soportará el estándar XWindow, porque brinda los mecanismos con los cuales se pueden construir una gran variedad de interfaces. Para ello adoptaremos la plataforma Sun que incluye Open Windows que es un ambiente gráfico basado en el estándar X11.

Aquí hay que hacer notar que, por la parte de desarrollo de formas, no encontré en nuestro mercado una herramienta de desarrollo que incluya generación de formas y pantallas para el usuario y que éste soporte el estándar FIMS. En el caso de los manejadores de base de datos, cada uno tiene su propia herramienta para generar formas y ninguna maneja estándares formalmente reconocidos. En el caso de ORACLE, éste cuenta con su herramienta "FORMS".

En este punto no podremos apegarnos a un estándar y tendremos que comprar lo que haya en el mercado.

El lenguaje elegido es "C" debido a que es el lenguaje nativo del ambiente UNIX.

El manejador de base de datos elegido es ORACLE, debido a que corre en diversas plataformas y soporta el estándar SQL.

Finalmente la aplicación pide que se tenga acceso a los datos desde cualquier plaza, hemos visto que el protocolo TCP/IP es un estándar disponible en diferentes plataformas, desde mainframe hasta PC, así que es el más adecuado para esta selección y, sobre todo, tomando en cuenta que ya se puede encontrar en diferentes plataformas y que las puede interconectar.

Todos los estándares anteriores se encuentra solamente en un ambiente UNIX, así que deberá ser éste el sistema operativo que se elija como base para nuestra aplicación.

En este punto nuestro ejercicio se complica, como dice el dicho popular "No tenemos mucha tela de donde cortar", ya que con base en lo anterior nos damos cuenta que para lograr lo que deseamos, no podemos aprovechar muchos de los recursos ya existentes, por lo que para esta aplicación se tendrá que adquirir hardware y las herramientas de software necesarias para generarla.

### **5.3.5.- ADQUISICION, DESARROLLO O MIGRACION DE APLICACIONES**

En la aplicación que nos ocupa sería muy remoto pensar en encontrar en el mercado un sistema que reúna todas las características que requiere la misma, así que en el caso de esta aplicación, no procede la adquisición. Tal vez cuando se analice el sistema de nómina o contabilidad, podremos optar por adquisición, pues en el mercado es común encontrar este tipo de sistemas.

Migración suena atractivo, pero ya vimos que actualmente esta aplicación se encuentra:

En mainframe dentro de un sistema propietario (la mayoría de los créditos se manejan aquí) en un manejador de base de datos tipo red.

En un servidor Altos. Aquí solo se manejan créditos a la vivienda y está en un manejador de base de datos tipo relacional.

De aquí lo único que podemos migrar serán los datos. Los datos deberán pasarse a un formato ASCII y ser transferidos a la plataforma que se seleccione. La herramienta para generar la aplicación, deberá tener los mecanismos necesarios para formatear los datos ASCII a su ambiente.

Como la aplicación va a generarse completamente, puesto que los requerimientos así nos lo obligan, se adquirirá:

- Equipo de interconectividad: Compuertas, concentradores, ruteadores (Diferentes marcas pero que soporten agentes SNMP).
- Servidores y estaciones de trabajo SUN.
- Manejador de base de datos ORACLE.
- Herramienta para desarrollo de aplicaciones: CASE de ORACLE.
- Sistema operativo de la SUN: Solaris.

Este ambiente se interconectará a los sistemas mayores para:

- . Aprovechar la infraestructura de telecomunicaciones actual.
- . Aprovechar la base instalada de terminales 3270.
- . Integrar la red TCP/IP a la red SNA.

### **5.3.6.- VERIFICAR LAS CARACTERISTICAS DE SISTEMAS ABIERTOS DE LA APLICACION**

En este punto haremos lo siguiente:

- a) Recurrir a la documentación de definición de requerimientos para verificar que éstos sean satisfechos.



- b) Revisar que el sistema tenga la capacidad de interoperar entre las diferentes plataformas: mainframe y redes locales.
- c) Para probar la portabilidad es necesario que se tome algunos de los módulos de la aplicación en una plataforma, y probarlo en otra plataforma, aquí se comprobará a que nivel de portabilidad se encuentra nuestra aplicación: a nivel fuente, objeto o ejecutable.

#### **5.3.7.- VERIFICAR LOS OBJETIVOS**

En el caso del ejemplo comparamos los resultados con los objetivos planteados.

#### **5.3.8.- SE REPETIRÁ LA SECUENCIA**

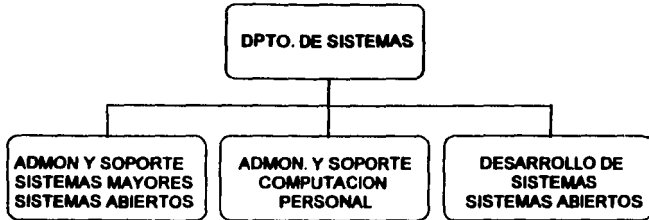
#### **5.4- CONCLUSION DEL EJERCICIO**

En estas condiciones tenemos que admitir una realidad tajante: Para esta aplicación no será posible utilizar ninguna de las plataformas que ya se tienen porque:

- La plataforma mainframe ya maneja TCP/IP, pero una característica que en este caso no maneja es la interface de usuario. Todos conocemos la forma tan rudimentaria y tan poco amigable que presentan las terminales de la familia 3270. Seguramente IBM tendrá equipo que maneje caracteres gráficos más eficientemente pero aquí estamos hablando de la inversión hecha con anterioridad pues se cuenta en el área metropolitana con aproximadamente 128 terminales 3270 y en algunos casos, computadoras personales con programa emulador de terminal 3270. En el caso de sucursales foráneas, los servidores ALTOS emulan un controlador de terminales IBM 3274 con terminales de esta misma familia, que aunque son PC's, al momento de emular una 3270 se tiene una presentación básica, no podemos manejar caracteres gráficos.
- En el mainframe se tiene un ambiente propietario con el sistema operativo VSE/ESA, que tiene además arquitectura SNA para manejo de red, por lo que la única opción es integrar el mundo SNA con el mundo UNIX a través de un gateway, con lo que se tendrá las ventajas de que por ese medio se tenga acceso desde terminales 3270 al mundo UNIX y viceversa. Lo anterior con objeto de migrar datos y seguir dando el servicio mientras se termina la generación de la aplicación en la nueva tecnología.
- La plataforma de servidores ALTOS. Aquí se tiene la ventaja de que ya es UNIX, de que Informix es un manejador de base de datos relacional y que brinda soporte SQL, pero tenemos la gran desventaja de que este equipo ya es obsoleto, no se tiene soporte para el mismo y de que el software actual no es compatible con ese UNIX instalado.

Se deberá comprar equipo.

En estas condiciones la estructura organizacional deberá cambiarse y podría ser como sigue:



**Funciones generales de cada área:**

**ADMINISTRACION. Y SOPORTE A SISTEMAS MAYORES Y SISTEMAS ABIERTOS.**

- Administración y soporte a sistemas mayores, sistemas abiertos, red SNA, red TCP/IP.
- Administración, control de la producción y operación de sistemas mayores.

**ADMINISTRACION. Y SOPORTE A COMPUTACION PERSONAL.**

- Control e inventario de equipo.
- Instalación de hardware y software (sistema operativo y paquetería).
- Soporte al software y usuarios.

**DESARROLLO DE SISTEMAS ABIERTOS.**

- Análisis, diseño y desarrollo de sistemas en nueva tecnología.
- Mantenimiento a las aplicaciones de sistemas mayores (sólo mantenimiento).

**Con la estructura planteada se pretende:**

- **Integrar las funciones de desarrollo de sistemas.**
- **Integrar la administración y soporte en las diferentes plataformas.**
- **Integrar al personal de sistemas mayores a la nueva tecnología.**

**CAPITULO 6**  
**CONCLUSION**

## **CAPITULO 6**

### **Conclusiones.**

Después del análisis realizado en esta tesis, lo primero que nos dimos cuenta es que el concepto "SISTEMAS ABIERTOS" no tiene el significado trivial que el común de la gente le atribuye, ni tampoco aquel que con tanta insistencia pregonan los vendedores.

Los Sistemas abiertos son aquellos que:

- Instrumenten suficientes especificaciones abiertas para interfaces y servicios. Los estándares son la piedra filosofal de los sistemas abiertos.
- Soporten formatos que permitan que aplicaciones de software correctamente definidas, sean portadas a través de un amplio rango de ambientes con cambios mínimos. (PORTABILIDAD).
- Que interoperen con otras aplicaciones en un ambiente remoto o local, con el mismo o diferente tipo de plataforma. (INTEROPERABILIDAD)
- Que interactúen con los usuarios en un estilo que facilite la portabilidad de los mismos. (INTEGRACION).

Otro punto interesante fue el relativo a las promesas y realidades de los sistemas abiertos.

Promesas tales como:

Portabilidad.- De datos es factible con algunas consideraciones.

De aplicaciones. Aquí solo podemos decir que se está en un punto inicial de partida. Falta mucho por hacer.

Hoy no es realidad.

De personal. Todavía no podemos decir que en este momento la tecnología nos brinde muchas ventajas. No tenemos interfaces consistentes. Se está trabajando pero muy lentamente.

Hoy no es realidad.

**Interoperabilidad.-** Con el modelo cliente-servidor y gracias a que en este elemento se tiene más madurez tanto en hardware como software, ya podemos decir que estamos en la línea correcta para poder cumplir con esta promesa.

**Independencia de hardware y software.-** Hoy no es realidad.

**Independencia de proveedor.-** Hoy no es realidad.

**Estándares.-** Es difícil que los fabricantes los acepten e impulsen. La realidad exige que los fabricantes acepten el riesgo que implica soportar estándares.

En suma podemos decir que actualmente la tecnología de la información dentro del marco de sistemas abiertos es incipiente y muy inmadura.

Aún cuando los sistemas abiertos no cumplen con las promesas establecidas, hay que reconocer que son muchos los beneficios que podrían aportar a los diferentes grupos. Por lo que es necesario establecer que los diferentes sectores involucrados deberán seguir trabajando para finalmente poder cumplir las promesas establecidas.

A las organizaciones si les conviene marchar por esta línea, no hay otra opción más adecuada que optar por iniciar un cambio hacia esta que podríamos llamar "Filosofía" de la tecnología.

Si las organizaciones deciden iniciar este cambio, serán ellas las que obliguen a los fabricantes a observar estándares formales en la fabricación de sus productos.

Las organizaciones deberán tener cuidado al querer implementar sistemas abiertos puesto que cualquier cambio implica muchos riesgos. Se deberá:

- Cumplir con los objetivos planteados.
- La aplicación no deberá perder funcionalidad.
- Obtener beneficios a la empresa.
- Crear un ambiente productivo.
- No afectar el rendimiento de una aplicación o sistema.

Hay que considerar que la metodología aquí planteada, es un punto de vista muy general que deberá ser ponderado en cada organización y en cada uno de sus pasos para poder lograr buenos resultados. En el ejemplo dado pocos fueron los productos que sujetándose a estándares formales, proporcionaron la funcionalidad requerida.

Un factor de suma importancia que no deberá pasarse por alto, es el referente a los recursos humanos, un proceso de reestructuración organizacional deberá ir aparejado con los cambios tecnológicos para que dichos cambios brinden los resultados deseados. No se podrá seguir operando con una estructura que fué adecuada para un concepto cerrado, jerárquico, centralizado e inflexible. Los tiempos modernos requieren que la parte humana también se modernice y esta modernización deberá ser la adecuada para funcionar en los esquemas actuales. No hablo solamente de la entidad "Dpto. Sistemas de Información" que hay en toda empresa, me refiero a algo más general, ya que deberá involucrar a los usuarios que se verán muy influidos en la forma en que ahora deberán trabajar.

Lo anterior es hasta cierto punto difícil de aceptar, pero de manera natural el papel de cada miembro del área de sistemas está cambiando.

También dentro del factor humanos está la capacitación adecuada. La tecnología de sistemas abiertos requiere de gente especializada que muchas veces es difícil de encontrar, por lo que una manera de resolver este problema será con un buen programa de capacitación para el personal con que se cuenta.

El día de hoy la mayoría de las piezas que integran un auto, están fabricadas siguiendo estándares internacionalmente aceptados, en la estructura de los aparatos eléctricos y electrónicos y en muchos otros dispositivos, se contempla el uso de estándares, lo anterior hace posible que dichos artículos puedan ser utilizados en diferentes puntos geográficos, las fronteras entre países no cuentan al momento de utilizar un artículo que ha sido fabricado conforme a estándares.

En la industria de la tecnología de la información deberá suceder lo mismo.

Con la información que hemos recibido aquí, sabemos que el día de hoy solo tenemos promesas de los sistemas abiertos, nos dimos cuenta también de las pocas realidades.



**Pero toca a cada elemento involucrado en este negocio, armar su propio rompecabezas a partir de todas esas piezas que ya cuentan con respaldo de estándares.**

**Lo que se dice hoy "Sistemas Abiertos" son solamente incipientes esbozos de lo que muchos quisieran ya tener.**

**APENDICE**

**A - B - C**

**APENDICE A.- INDICE DE FIGURAS**

<b>FIGURA NUM. 1</b>	<b>INFRAESTRUCTURA CLASICA</b>
<b>FIGURA NUM. 2</b>	<b>POSICION DEL DEPTO. DE S.I. EN LOS 80'S</b>
<b>FIGURA NUM. 3</b>	<b>POSICION DEL DEPTO. DE S.I. EN LOS 70'S</b>
<b>FIGURA NUM. 4</b>	<b>POSICION DEL DEPTO. DE S.I. EN LOS 80'S</b>
<b>FIGURA NUM. 5</b>	<b>POSICION DEL DEPTO. DE S.I. EN LOS 90'S</b>
<b>FIGURA NUM. 6</b>	<b>SISTEMAS ABIERTOS.- PERSPECTIVAS</b>
<b>FIGURA NUM. 7</b>	<b>PORTABILIDAD DEL SOFTWARE</b>
<b>FIGURA NUM. 8</b>	<b>INTEROPERABILIDAD</b>
<b>FIGURA NUM. 9</b>	<b>INTEGRACION</b>
<b>FIGURA NUM. 10</b>	<b>ARQUITECTURA DE SISTEMAS</b>
<b>FIGURA NUM. 11</b>	<b>DIVISION VERTICAL</b>
<b>FIGURA NUM. 12.</b>	<b>DIVISION HORIZONTAL</b>
<b>FIGURA NUM. 13</b>	<b>MAQUINA ABSTRACTA</b>
<b>FIGURA NUM. 14</b>	<b>PLATAFORMAS DE COMPUTO</b>
<b>FIGURA NUM. 15</b>	<b>MAQUINA MULTINIVEL</b>
<b>FIGURA NUM. 16</b>	<b>MODELO "DISTRIBUTED MANAGEMENT ENVIROMENT"</b>
<b>FIGURA NUM. 17</b>	<b>EL "X WINDOW SYSTEM"</b>

<b>FIGURA NUM. 18</b>	<b>COMPONENTES DEL ESTANDAR POSIX 1003.2</b>
<b>FIGURA NUM. 19</b>	<b>LENGUAJES ESTANDARIZADOS INTERNACIONALMENTE</b>
<b>FIGURA NUM. 20</b>	<b>ESTANDARES RELACIONADOS A LENGUAJES</b>
<b>FIGURA NUM. 21</b>	<b>MODELO DE PRODUCTO INTERNACIONAL</b>
<b>FIGURA NUM. 22</b>	<b>COMPONENTES DEL PERFIL DE SISTEMAS ABIERTOS</b>
<b>FIGURA NUM. 23</b>	<b>NIVELES EN LOS QUE EXISTEN LOS PERFILES</b>
<b>FIGURA NUM. 24</b>	<b>TCOS DE IEEE</b>
<b>FIGURA NUM. 25</b>	<b>NIST APPLICATION PORTABILITY PROFILE (APP)</b>
<b>FIGURA NUM. 26</b>	<b>X/OPEN COMMON APPLICATION ENVIRONMENT (CAE)</b>
<b>FIGURA NUM. 27</b>	<b>OPEN SOFTWARE FOUNDATION APPLICATION ENVIRONMENT SPECIFICATION (AES)</b>
<b>FIGURA NUM. 28</b>	<b>ESTRUCTURA DE SUNNET MANAGER</b>
<b>FIGURA NUM. 29</b>	<b>APLICACION DE TIPO CONTABLE</b>
<b>FIGURA NUM. 30</b>	<b>APLICACION INGENIERIL CAD/CAM</b>
<b>FIGURA NUM. 31</b>	<b>AMBIENTE ACTUAL</b>

## APENDICE B.- GLOSARIO

### ANSI

American National Standards Institute.- La organización de estándares en los Estados Unidos.

### API

Application Program Interface.- Un paquete de programas y funciones que dan al programador de aplicaciones la habilidad de comunicarse con otro ambiente.

Una interface que permite escribir programas de aplicación sin tener que saber detalles internos del ambiente para el cual se está escribiendo.

### APP

Application Portability Profile.- Un perfil que tiene como objetivo promover la portabilidad de las aplicaciones.

### BITMAP

Mapa de bits.- Un mapa de puntos en el que cada punto está representado por un único bit.

### CAE

Common Application Environment.- Un ambiente para una gran variedad de facilidades e interfaces. Parecido a un OSE pero no especificado por un organismo formal de estándares. Usado por X/Open y EWOS.

### CCITT

International Consultative Committee on Telegraphy and Telephony.- Una organización internacional que desarrolla recomendaciones para telecomunicaciones, interconexión e interoperabilidad. Sus recomendaciones son frecuentemente adoptadas por ISO como estándares.

## **CLIENTE**

En un ambiente de red TCP/IP, un proceso que emplea (o consume) recursos proporcionados por un servidor. El proceso cliente es iniciado por un usuario cuando éste emite un comando de red. El proceso cliente envía una solicitud de servicio a un proceso servidor que reside en un host remoto. Si la solicitud es atendida, se establece una conexión entre el proceso cliente local y el proceso servidor remoto.

## **CMIP**

Common Management Information Protocol.- El protocolo de administración de red OSI.

## **ESPECIFICACIONES**

Una especificación es un documento que dice lo que hace una pieza de hardware o software y como ésta deberá ser usada. Tal especificación puede ser la documentación del vendedor para su producto. Puede también incluir documentos respecto al diseño que fueron utilizados al producir tal software o hardware; tales documentos de diseño pueden ser modelos en el sentido general, destinados éstos a mostrar como se ensamblan las diferentes piezas de un producto. Cada vendedor tendrá diferentes versiones de dichos documentos propietarios, así que dichos documentos no son muy útiles en la selección de productos entre diferentes vendedores. En su lugar, especificaciones escritas por terceras partes (Organizaciones Formales de Estándares) son frecuentemente más útiles. Tales especificaciones necesitan ser evolucionadas y mantenidas mediante un proceso abierto que permite que todas las partes involucradas intervengan, dichas especificaciones son llamadas **ESPECIFICACIONES ABIERTAS**.

## **ESTANDAR BASE**

Un estándar que no requiere de otros estándares. Formalmente, un Estándar Internacional, un Reporte Técnico, o una recomendación CCITT usada en la especificación de un perfil.

## **ESTANDAR DE FACTO**

Un estándar que ha sido aceptado ampliamente pero no es reconocido oficialmente por las organizaciones formales que se dedican a la creación, evolución y mantenimiento de estándares.

### **ESTANDAR DE JURE**

Un estándar que ha sido oficialmente reconocido por una organización formal de estándares.

### **ESTANDAR FORMAL**

Otro nombre para un estándar, algunas veces usado para distinguirlo de una especificación. Un estándar formal es un estándar "De Jure".

### **ETHERNET**

Una topología en redes LAN en la cual los dispositivos son conectados a un cable o bus común, utilizando el protocolo CSMA/CD.

### **EWOS**

Expert Group on the Common Application Environment.- Un forum que proporciona y coordina la información de Europa para la estandarización internacional de perfiles.

### **FILE SERVER**

Servidor de archivos.- Una computadora de alta velocidad que funciona como repositorio central de datos y/o programas de aplicación para la red. El servidor de archivos administra la red centralmente, procesa las comunicaciones, controla el acceso de los usuarios y comparte impresoras.

### **FTAM**

File Transfer, Access, and Manipulation.- El protocolo OSI para transferencia de archivos.

### **GOSIP**

Government OSI Profile.- El perfil OSI del gobierno de los Estados Unidos especificado por NIST. FIPS 146-1.

### **GUI(Graphical User Interface)**

Interfaz Gráfica de Usuario.- Elemento intermedio entre la computadora y el usuario que se ejecuta en modo gráfico. Es el sistema con el que interactúa el usuario y que emplea iconos e imágenes gráficas para representar funciones.

**Ejemplo:** Open Look que distribuye AT&T y Sun Microsystems, así como Motif de OSF.

### **MLLAPI (High-Level Language Application Program Interface)**

Una interface de programación de IBM que habilita a que aplicaciones en PC se comuniquen con aplicaciones de mainframe.

### **ICON**

Icono.- Representación simbólica de un objeto, por ejemplo un cubo de basura puede representar el borrado de archivos.

### **IEEE**

Institute of Electrical and Electronics Engineers.- Una organización profesional internacional. Crea sus propios estándares y también emite estándares para ANSI.

### **IMPLEMENTACION**

Un paquete de software que fué creado conome un estándar o una especificación.

### **INTRINSICS**

El nivel de programación de X Window en el que se definen las funciones y tipos de datos estándar utilizados en la creación de WIDGETS e implementación de éstos en las aplicaciones de usuario.

### **IP**

Internet Protocol.- El protocolo clave en el juego de protocolo Internet, el cual también es conocido como "Protocolos TCP/IP". IP proporciona un ruteo y espacio de dirección comunes sobre redes diversas o en protocolos de capa de enlace.

### **ISO**

International Organization for Standarization.- Organismo internacional que trabaja los estándares formales.



## **IT**

Information Technology.- Tecnología de la Información.- Término general para referenciar tópicos relacionados a computación y comunicación de datos.

## **MAINFRAME**

Un sistema de cómputo grande, ejemplos: la serie de 3090 de IBM, cuyas características generales son: Multiproceso, multiusuario, ambiente transaccional eficiente, sistema de seguridad maduro y manejo de procesos por lotes.

## **MHS**

Message Handling System.- El protocolo de correo y mensajes en OSI.

## **MODELO CLIENTE/SERVIDOR**

Este modelo es el resultado de una mayor sofisticación y confiabilidad de las redes de área local (LAN). Uno de los principales objetivos de este concepto es descargar el trabajo de los servidores pues gran parte del proceso se efectúa en la computadora denominada **CLIENTE**, que por lo general es una PC, mientras que el **SERVIDOR** se dedica exclusivamente a responder a las peticiones de información del cliente. La idea principal de este concepto es utilizar al máximo lo mejor de cada uno de los dos equipos, por una parte en el servidor se aprovecha la características de multiproceso, seguridad y rapidez. En el cliente se aprovecha su ambiente gráfico que permite una fácil interacción entre las aplicaciones y los usuarios además de que la mayoría de la gente no tiene problemas en el manejo de computadoras personales (Ver **CLIENTE** y **SERVIDOR**).

En el contexto de X Window el modelo tiene estas dos componentes principales:

Clientes.- Son los programas de aplicación en este caso.

Servidores.- Residen en las unidades de visualización.

## **MODELO OSI**

Open Systems Interconnection.- Interconexión de Sistemas Abiertos.- Modelo de referencia desarrollado por la ISO (International Organization for Standardization) Organización Internacional de Normas, cuyo objeto es la

estandarización de varios protocolos destinados a la conexión de sistemas heterogéneos.

Este modelo consta de 7 capas, cada capa define sus protocolos, la pila de protocolos OSI que comunmente se utiliza es la siguiente:

<b>NUM. CAPA</b>	<b>PROTOCOLOS</b>	
7	Aplicación	VTM, FTAM, MHS
6	Presentación	IS 8822/x.216, IS 8823/X.226
5	Sesión	IS 8226&X.215, IS 8327/X.225
4	Transporte	IS 8072/X.214 (TP0-TP4)
3	Red	X.25
2	Enlace de datos	HDLC
1	Física	X.21

Muchas de las ideas del modelo OSI se basan en otros juegos de protocolos y modelos, tales como DECNET de DIGITAL, SNA de IBM y especialmente en el TCP/IP.

#### **MOTIF**

Una GUI favorecida por OSF.

#### **NFS**

Network File System.- Un sistema de archivos de red desarrollado por Sun Microsystems, Inc. y ahora ampliamente distribuido por otros vendedores.

#### **NIST**

National Institute of Standards and Technology.- Una agencia federal de los Estados Unidos dedicada a producir estándares para el gobierno.

#### **OPENLOOK**

Una GUI favorecida por UNIX International

#### **OSE**

Open System Environment.- Un ambiente especificado por un juego de estándares y perfiles para interfaces, servicios y formatos para obtener

interoperabilidad y portabilidad de aplicaciones, datos y gente. La guía POSIX IEEE P1003.0 especifica un OSE.

## **OSF**

Open software Foundation.- Un consorcio que produce especificaciones tales como el AES y productos de software, tales como OSF/1. OSF produce la arquitectura OSF DCE.

## **OSF/DCE**

OSF Distributed Computing Environment.- El modelo de OSF para cómputo distribuido.

## **OUTSOURCING**

Modalidad en la que las funciones que realiza el área de sistemas son delegadas a terceras partes en lugar de realizarlas en la propia corporación. Esta delegación puede incluir actividades de proceso central de datos, administración de red y desarrollo de software.

## **pcode (pseudocode)**

Un lenguaje intermedio entre el código fuente que escribe un programador y el formato binario que ejecuta una computadora. Desarrollado en el UCSD Pascal para promover portabilidad de aplicaciones.

## **POSIX Guide (IEEE P1003.0)**

Documento que especifica el POSIX OSE y sirve como un modelo de referencia para estándares e implementaciones POSIX. Producido por el IEEE P1003.0 o POSIX.0.

## **PERFIL**

Un documento que indica varios estándares base con parámetros, opciones, clases o subjuegos para construir un sistema de cómputo completo o para llevar a cabo una función específica. El término es definido por TR10000 (SGFS) y IEEE P1003.0.

## **PORTABILIDAD DE APLICACIONES**

Habilidad para mover software entre computadoras sin tener que reescribirlo.

## **POSIX**

Portable Operating System Interface.- Un juego de estándares que definirá una interface portable para sistemas abiertos.

## **PROCESO PARALELO**

Un tipo de proceso de cómputo que utiliza cientos o aún miles de microprocesadores, compartiendo la tarea de proceso entre muchas unidades pequeñas.

## **PROTOCOLOS**

El protocolo de red es una especificación que indica como enviar, recibir e interpretar información a través de una red de computadoras. Los protocolos son usualmente definidos en capas y, dentro de este marco, cada protocolo en cada capa se define para que únicamente se comunique con la capa superior y la inferior. En este sentido, los protocolos de red no son más que especificaciones de una interface. Sin embargo, eventualmente la información pasada a protocolos de red viaja sobre la red para alcanzar otros procesos o computadoras.

## **RPC**

Remote Procedure Call.- Un API para ejecución remota de funciones detalladas.

## **SERVER**

Servidor.- En un ambiente de red TCP/IP, un proceso que proporciona recursos a una red. El servidor es un proceso que reside en un host remoto que atiende requerimientos hechos por un cliente. El servidor es un proceso que corre en background y que está atento a solicitudes de servicio. Cuando un servidor recibe un requerimiento, establece una conexión con el cliente solicitante, crea un subproceso y nuevamente regresa a su estado de espera de requerimientos.

## **SNA**

Systems Network Architecture.- Modelo de red propietario establecido por IBM. Describe la estructura lógica, formatos, protocolos, secuencias

operacionales de las unidades de transferencia de información a través de una red, así como la forma en que se controla y opera la misma.

### **SNMP**

Simple Network Management Protocol.- El estándar para administración de red usado con TCP/IP.

### **SOCKETS (Berkeley sockets)**

Una interface para servicios generales de red .

### **SVID**

System V Interface Definition.- La descripción de UNIX System V producida por USL (UNIX System Laboratories).

### **TCOS**

Technical Committee on Operating Systems and Application Environments.- El Comité Técnico de Sociedad de Cómputo de la IEEE que es responsable de los sistemas abiertos.

### **TCP**

Transmission Control Protocol. El protocolo de transporte Internet.

### **TCP/IP**

Grupo de protocolos diseñados para interconectar redes heterogeneas, entre los protocolos del modelo Internet (Así conocido) están:

<b>CAPA</b>	<b>PROTOCOLOS</b>
Proceso/Aplicación	TELNET, FTP, SMTP
Transporte	TCP
Internet	IP
Red	X.25, Ethernet, FDDI
Física	Varios.

## **TELNET**

- 1.- Acrónimo de TELETYPE NETWORK.
- 2.- Un protocolo TCP/IP usado para acceso a consola de sistemas remotos.

## **UNIFORM**

UniForum (International Association of Open Systems Professionals).- Grupo de usuarios con más influencia en estándares.

## **WORKSTATIONS**

Estaciones de trabajo.- Computadoras de sobremesa cuyas capacidades computacionales y gráficas son mayores que las de las microcomputadoras de gama alta.

## **WINDOW MANAGER**

Gestor de ventanas.- Un cliente especial responsable de la gestión de ventanas en la pantalla.

## **XLib**

Una parte estándar del sistema X Window que le permite a los programadores crear y usar bloques estándar para elaborar presentaciones sobre la pantalla llamados widgets. También se le conoce como X Toolkit.

## **X/Open (X/Open Company, Ltd)**

Un consorcio que produce especificaciones, tales como el XPG y juegos de prueba tales como elV SX para resolver incompatibilidades entre sistemas de cómputo y componentes de software.

## **X PROTOCOL**

Protocolo X.- El protocolo de red que define los bytes que se necesitan exactamente para realizar todas las operaciones X Window.

## **X TERMINAL**

Terminal X.- Una terminal gráfica sin disco sobre la que se ejecuta un único programa, el programa servidor X Window.

## **X TOOLKIT INTRINSICS**

Una parte estándar del sistema X Window que le permite a los programadores crear y usar bloques estándar para elaborar presentaciones sobre la pantalla llamados widgets. También se le conoce como Xt Intrinsic.

## **WIDGETS**

Bloques estándar para elaborar presentaciones sobre la pantalla como por ejemplo menús, barras de desplazamiento, botones y cuadros de diálogo.

## **X WINDOW**

Sistema "X Window".- Herramienta de software para el desarrollo de interfaces gráficas de usuario (GUI'S) sobre estaciones de trabajo. A veces erróneamente llamado "X Windows".

El sistema X Window brinda independencia de dispositivo, pues las interfaces gráficas de usuario desarrolladas sobre una computadora, se pueden ejecutar en distintas computadoras, ya que proporcionan una salida común para estaciones de trabajo, mainframes, minicomputadoras y microcomputadoras conectados en red.

## APENDICE C.- BIBLIOGRAFÍA

McNurlin, Bárbara C. , R. H. Sprague Jr., *Information Systems Management in Practice.*, 2a. Edición, Englewood Cliffs, Prentice Hall, New Jersey, 1989.

Day, Michael, L. Budnick, E.c Harper, B. Homer, J. Jerney, J. Krochmal, R. Lewis, D. Williams, *Las Operating systems*, New Riders Publishing, Carmel, Indiana, 1993.

Nutt, Gary J., *Open Systems*, University of Colorado, PTR Prentice Hall, Englewood Cliffs, New Jersey, 1992.

Apuntes del curso impartido por HP titulado, *IT Standards for Open Systems*, 1992.

Tanenbaum, Andrew S., *Redes de ordenadores*, Prentice Hall Hispanoamericana, S.A., 2a. Edición, 1991.

Tanenbaum, Andrew S, *Organización de Computadoras, Un Enfoque Estructurado*, 3a. Edición.

Pressman, Roger S, *Ingeniería de Software, un enfoque práctico*, 3a. Edición.

Reiss Levi, Joseph Rodin, Aplique X Window, *Serie Mc GrawHill de Informática*, 1993.

Date C.J., Introducción a los sistemas de base de datos. 5a. Edic., Vol. 1, Addison Wesley Iberoamericana, U.S.A. 1993,

*Revista Red. Año IV, Núm. 47*, Agosto de 1994.

*SunNet Manager 20 Reference*, Junio 1992

Márquez García Francisco Manuel, *UNIX Programación Avanzada*, Addison Wesley Iberoamericana, USA, 1993

Perspectivas, Hewlett Packard No. 0, Vol. 1



Quarterman John S., Susanne Wilhelm, ***UNIX, POSIX and Open Systems, The Open Standards Puzzle***, Addison Wesley Publishing Company, USA, 1993.

Digital Equipment Corporation, Open Systems Handbook, ***A Guide to Building Open Systems***, USA, 1991.