Original

Universidad Nacional Autónoma de México

Facultad de Ciencias

EL PROCESO DE LA EVALUACION, PLANEACION Y CONTROL DE LA CAPACIDAD DE UN COMPUTADOR

T E S I S

Que para obtener el título de:

A C T U A R I O

P r e s e n t a n :

JORGE CONTRERAS IRIGOYEN

Y

JAIME A. VILLANUEVA GONZALEZ

Cd. Universitaria, 1981

I N D I C E

Capítulo IV HERRAMIENTAS DE MEDICION Y PREDICCION DEL RENDIMIENTO

- Objetivos de la medición
- Requerimientos de información
- Métodos de medición y tipo de herramientas
- Métodos de proyección de cargas futuras
  - . BENCH-MARK
  - . Simulación
  - . Modelos matemáticos o analíticos

Capítulo V IMPLEMENTACION

- Situación típica de las instalaciones actuales
- Fases de implementación y actividades por fase
- Requerimientos de personal

Capítulo VI EJEMPLO

- Antecedentes
- Metodología
- Terminología
- Fases del estudio
- Actividades realizadas y resultados obtenidos

P R E F A C I O

La creciente demanda de equipos de procesamiento de datos ha - propiciado tanto la multiplicación de instalaciones de cómputo electrónico en el sector público y en el privado como la sofisticación- y crecimiento de aquéllos que ya existían en la década de 1960. Sin embargo, los centros de enseñanza del país no han producido los recursos humanos necesarios para la administración y desarrollo de -- los centros de informática.

Esto ha dado lugar a que, salvo en muy contados casos, las organizaciones no cuenten con el personal adecuado para el desarrollo de aplicaciones y carezcan de políticas y metodologías de trabajo - que permitan analizar al centro de informática como a cualquier --- otro centro de producción. Es fundamental implementar las funciones de evaluación, planeación y control (EPC), para que los directivos de las instituciones estén sistemáticamente informados sobre la situación del centro de informática en sus diferentes aspectos (producción, servicio a los usuarios, relación de costo/beneficio, etc.) de tal manera que puedan reaccionar con suficiente oportunidad ante el incremento de la demanda, el ofrecimiento de nuevos productos, - la implantación de nuevas aplicaciones, etc.

El objetivo de este estudio es el planteamiento de los alcan--

ces del proceso de EPC, así como el análisis de los aspectos que afectan su implantación, las herramientas de medición que contribuyen a una mejor utilización de los recursos de cómputo; los sistemas administrativos que impactan en la productividad del centro de informática; los aspectos del sistema operativo que deben ser analizados con mayor detenimiento para que se cumpla con los objetivos de la instalación; la planeación del personal necesario para la implantación de esta metodología; las metodologías existentes para la proyección de cargas de trabajo, etc.

Ya que cada uno de estos aspectos es de por sí motivo de un análisis profundo, cuya exposición requeriría de un trabajo específico con consideraciones diversas como tipo de computador, sistema operativo, medio ambiente de la instalación, objetivos de la institución, etc., en este trabajo no profundizaremos en su análisis, sino que exclusivamente señalaremos la importancia de su estudio en el proceso de EPC.

Consideramos conveniente incluir un caso práctico de la evaluación de la capacidad de un centro de cómputo para ejemplificar algunos tópicos que se describen a propósito del proceso.

En la exposición de este caso práctico se analizan los requerimientos de cambio del equipo de cómputo. Sin embargo, queremos hacer resaltar el hecho de que esta tesis de ninguna manera pretende analizar la problemática que representa el cambio de equipo, de por sí interesante, pues el trabajo presentado por la matemática Guadalupe Quijano León y el actuario Sergio Gilberto Barmet: Evaluación de equipos de cómputo para la UNAM, y el del M. en C. Eduardo Ojeda

Trejo: Una metodología para la selección de equipos de cómputo, analizan este problema de una manera clara y didáctica. Por otro lado, debido a que consideramos que otro de los objetivos del proceso EPC es recopilar toda la información necesaria para el funcionamiento del centro de cómputo, incluida la proyección de cargas de trabajo para poder determinar situaciones críticas a mediano o largo plazo para en su momento desarrollar un estudio de viabilidad para la ampliación o cambio del equipo instalado; en este sentido, el presente trabajo puede considerarse como complemento de otros estudios sobre los procesos de evaluación y selección de equipos de cómputo.

Otro aspecto que debemos apuntar se relaciona con la experiencia de los autores, quienes se han desarrollado profesionalmente en el campo de la informática fundamentalmente con equipo IBM. Apesar de que para el desarrollo de este trabajo se consultaron obras de diferentes autores y se trató de hacer una exposición general de los conceptos relacionados con el proceso EPC, es común encontrar términos y conceptos usados en la jerga de IBM, lo cual nos obliga a aclarar que algunos podrían tener diferente connotación en la jerga de algún otro proveedor.

Es importante subrayar que en el desarrollo de esta tesis se utilizará gran cantidad de anglicismos, pues son más precisos, o más conocidos que los términos en español, cuando los hay. Tal es el caso del término BIT, cuyo significado en español es "pedacito", "pizca", "un poquito", etc., y que en la jerga de la computación se conoce como la unidad mínima de información con características dicótomas de encendido o apagado, sí o no, etc.

También emplearemos vocablos de uso común en materia de computación como SOFTWARE o HARDWARE, cuya traducción al español, según algunos autores, es "programalógica" o "programática" y "mecatrónica" o "mecamática", respectivamente, pero que a la fecha no ha sido aceptada en el medio mexicano.

Finalmente diremos que tratamos de utilizar preferentemente términos en español ("rendimiento" por PERFORMANCE; "E/S" por I/O, etc.) siempre que consideremos que la traducción conserva el sentido de la palabra original, o que la traducción sea ampliamente usada, ya que el objetivo de esta tesis es proporcionar al lector una guía adecuada para el desarrollo del proceso EPC de acuerdo con la terminología actualmente en uso en el mercado mexicano del procesamiento de datos.

C A P I T U L O I

INTRODUCCION

- El centro de cómputo como centro de producción
- Definición de EPC
- Motivación e importancia

EL CENTRO DE COMPUTO COMO UN CENTRO DE PRODUCCION

La electrónica nació en 1889 cuando JHON AMBROSE FLEMING, profesor de la facultad de Ingeniería de la Universidad de Londres, -- encontró que al modificar una lámpara como las que había inventado Edison, ésta podía utilizarse para detectar corrientes alternas de alta frecuencia. Sin embargo, no fue sino hasta la Segunda Guerra Mundial cuando el invento pudo ser utilizado para construir la primera computadora electrónica, que utilizaba miles de bulbos que funcionaban como interruptores. (Nonotza 01)

En esa época una computadora que podía multiplicar dos cifras de 23 dígitos en seis segundos pesaba cinco toneladas, era cara y poco confiable.

La aparición del transistor en 1947, inventado por los laboratorios Bell, fue el primer paso hacia la desaparición de la computadora gigantesca, no en capacidad de cómputo, sino en dimensiones. A finales de la década de 1950 hicieron su aparición otras computadoras más sofisticadas, mucho más confiables y con mucho mejor relación precio / rendimiento.

En ellas se empleaban transistores para las operaciones aritméticas, núcleos de ferrita para la memoria y discos o cintas magnéticas para el almacenamiento. Podían multiplicar dos cifras de 10 dí

gitos en 1/100,000 de segundo. Los avances tecnológicos están in-- fluyendo para que la habilidad de almacenar, procesar y comunicar - información tienda a infinito, mientras el costo tiende a cero. -- Por ejemplo, el costo de 100,000 multiplicaciones en 1952 era de -- aproximadamente 30 pesos; actualmemte cuesta menos de 15 centavos.

Esto permite que la computadora sea ya una herramienta al al-- cance de la mayoría de las instituciones.

Sin embargo, conforme la computadora va siendo cada vez más po pular como herramienta, se observa que el aspecto administrativo de los centros de cómputo no ha tenido un desarrollo paralelo al desa- rrollo tecnológico.

En los primeros años de la computación electrónica en los nego cios, mucha gente pensaba que el proceso de datos era muy complica- do en su administración; el computador y las personas relacionadas- con éste (programadores, analistas, operadores, etc.), eran conside rados en el resto de la institución casi como un mal necesario.

Esto puede atribuirse a la mística que rodeaba en ese entonces el proceso de datos, debido principalmente a que:

- La computación electrónica era un campo completamente nuevo y poca gente no relacionada directamente con la computadora (usuarios potenciales) conocía sus posibilidades.
- La función de procesamiento de datos se encargaba normalmen- te a un especialista técnico con pocos conocimientos de admi nistración.
- El resto del personal encargado de esta función era muy esca so y estaba mal preparado. Además, utilizaba un lenguaje --

tan técnico y sofisticado que dificultaba la comunicación -- con los usuarios.

Gradualmente, conforme las operaciones de proceso de datos se popularizan y representan un mayor porcentaje de los recursos financieros de las instituciones, los altos funcionarios se preocupan cada vez más porque estas operaciones tengan un buen retorno a la inversión y presionan a los responsables para que administren más eficientemente su operación.

Los usuarios conocen cada vez más la naturaleza del proceso de datos, y los centros de cómputo son dirigidos por personas con experiencia en administración.

Sin embargo, debido al crecimiento desproporcionado en esta -- área y a la sofisticación cada vez mayor de los computadores, los - problemas de administración continúan, y en muchos casos parecen -- agudizarse aún más. Se gasta dinero en el desarrollo e implanta--- ción de aplicaciones que no producen claros beneficios a las instituciones. Algunas aplicaciones no se implantan en las fechas establecidas de antemano y muy a menudo sobrepasan el presupuesto estimado, los niveles de servicio prometidos no se cumplen, lo cual influye en que el usuario se retrase significativamente en sus funciones.

Algunos centros de cómputo tienen cada vez menos capacidad para satisfacer los nuevos requerimientos del usuario; no se tiene una idea clara de la razón de esta falta de capacidad de producción ... todo esto influye para que el centro de proceso de datos no demuestre que contribuye al éxito de la institución en la misma pro-

porción en que contribuye a los gastos.

Gran parte de estos problemas se solucionan cuando el proceso-de datos se administra según los mismos principios que durante mu-- chos años se han utilizado con éxito en centros de producción.

En un centro de cómputo la materia prima son los datos que se procesan y manipulan de acuerdo con ciertas especificaciones y prio ridades definidas por el propio usuario de la información, lo cual- da como resultado una serie de datos útiles y oportunos necesarios- para planear y controlar las actividades de una institución.

Esto significa que es necesario establecer un sistema administrativo que permita controlar adecuadamente las operaciones del cen tro de proceso de datos, que establezca un mecanismo de comunicación efectiva entre todas las partes relacionadas con esta función y que fomente una actitud de negocios en todos sus integrantes, siempre - pensando en los tres objetivos básicos de cualquier sistema de producción:

EFECTIVIDAD: Solamente puede lograrse cuando los objetivos y- planes están de acuerdo con los objetivos y planes de la institu--- ción, para lo cual se requiere de la participación activa tanto de- los altos funcionarios, como de los usuarios correspondientes.

CONTROL: No se puede conseguir a menos que el personal se sujete a la misma disciplina y métodos de trabajo que el resto de los miembros de la institución. Deberá establecerse un mecanismo que - permita verificar que los objetivos, fechas, costo y niveles de servi cio se cumplan.

EFICIENCIA: Se consigue cuando los técnicos en proceso de da-

tos toman una actitud de negocios. Esto significa que se evaluarán permanentemente los niveles de servicio y rendimiento, siempre tratando de encontrar mejores métodos para incrementar la productividad y mejorar el costo/rendimiento del sistema.

DEFINICION DE EVALUACION Y PLANEACION DE LA CAPACIDAD DE UN COMPUTADOR

En el punto anterior vimos la necesidad de administrar el centro de proceso de datos como si fuera un centro de producción basándose en los mismos principios de administración para que los objetivos estén de acuerdo con los objetivos de la institución y se alcancen en las fechas previstas, utilizando siempre el método más productivo y rentable.

En esta perspectiva la función de planeación y pronóstico llega a ser de gran importancia.

El objetivo de la planeación sería, entonces, definir la capacidad requerida del computador, con base en los pronósticos de cargas de trabajo, para alcanzar los objetivos de servicio requeridos por el usuario mediante la administración eficiente de los recursos (personal, HARDWARE, SOFTWARE).

Los conceptos en los que se apoyan la planeación y la evaluación de la capacidad (EPC) de un computador no son nuevos. El objetivo de este estudio es analizar las consideraciones necesarias que permitan al administrador de un centro de cómputo conocer:

. Los parámetros que definen las características de la carga de trabajo de una instalación.

- Las características de los componentes de SOFTWARE y HARDWARE que impactan en el rendimiento de un sistema.
- Las consideraciones para pronosticar cargas y rendimiento futuro.
- Las herramientas necesarias para reunir, analizar y reportar los datos antes mencionados.

Como se observa, mediante el proceso de EPC se monitorea y analiza la carga, la utilización y el tiempo de respuesta para cada recurso del sistema según las cargas de trabajo actuales y futuras para poder proyectar y determinar la capacidad requerida y poder cumplir y satisfacer al usuario y a los funcionarios de una institución.

MOTIVACION E IMPORTANCIA

Los primeros computadores estaban diseñados para ser operados por el programador de aplicaciones. El programa diseñado por él, era el que se encargaba de administrar todos los componentes del sistema; los sistemas operativos (compiladores, métodos de acceso, manejo de colas de entrada/salida) eran prácticamente inexistentes. En estas circustancias la función EPC era relativamente simple, pues estaba basada en parámetros muy específicos como ciclos de CPU, tiempo de ejecución de las instrucciones básicas (sumas, restas, multiplicaciones, movimiento de datos, etc.), número y tipo de instrucciones que debían ejecutar los programas.

Al aparecer memorias mayores y más baratas, periféricos más sofisticados y más rápidos (discos, cintas, etc.), empezaron a aparecer ayudas de SOFTWARE como compiladores y ensambladores, y el concepto de sistema operativo, que no es más que una serie de programas responsables de la administración y control de los componentes de un computador, todo lo cual liberó al programador de aplicaciones de responsabilidades que anteriormente soportaba.

Toda esta serie de ayudas de SOFTWARE que venían junto con el computador mejoraron enormemente la productividad del sistema y de los programadores de aplicaciones. Sin embargo, se complicó la fun

ción de EPC porque ya no dependía solamente de los parámetros antes citados, sino que empezaron a intervenir nuevos elementos como el - diseño del supervisor y la velocidad de compilación. La situación- se complicó definitivamente cuando apareció en los años 60 el con-- cepto de multiprogramación, que tiene como objetivo aprovechar la - mayor velocidad de ejecución de la CPU y la memoria con respecto de los periféricos (ver capítulo Los sistemas operativos y su evolu--- ción) para mantener en ejecución concurrente varios programas que - comparten los recursos del computador (memoria principal y auxiliar, periféricos, etc.) La pregunta clásica ¿cuánto tiempo me llevará - ejecutar mi trabajo? ya no podía ser respondida fácilmente, pues el tiempo de ejecución de un mismo trabajo dependía mucho de las situaciones bajo las cuales se había ejecutado: solo en el computador, - con una mezcla de trabajos orientados a operaciones de entrada/salida o por el contrario, orientados a operaciones de CPU.

Esto obligó a la función responsable de EPC a aprender más sobre operaciones internas del SOFTWARE, algoritmos de programación - de trabajos, políticas de distribución de recursos, relaciones entre los diferentes componentes del computador, afinación, etc., lo que - influyó para que muchos funcionarios, viendo la complejidad de la - función EPC no la implantaran en sus centros de cómputo.

Sin embargo, el responsable de proceso de datos se dió cuenta- de que para mantener su credibilidad debía responder a preguntas -- como:

¿Cuánto gastaré en proceso de datos en los próximos 3 años?

¿Qué le pasará al servicio de mi sistema en línea cuando le agregue

40 terminales?

¿Podré instalar este paquete de teleproceso y mantener el rendimiento y servicio actual?

¿Cuánto necesito incrementar el poder de la CPU en los próximos 2 - años?

¿Cuánta memoria necesito para tener proceso compartido?

¿Qué nivel de rendimiento tengo de mi computador durante los diferentes periodos del día?

¿Existen cuellos de botella y cuáles son?

¿Qué porcentaje de la capacidad de cómputo utiliza la aplicación de administración de inventarios? ...

Para responder adecuadamente a este tipo de preguntas se requiere de una metodología, lo más simple y eficaz, que permita a la función de EPC realizar su trabajo.

C A P I T U L O II

LOS SISTEMAS OPERATIVOS Y SU EVOLUCION

- Concepto de sistema operativo
- Sistemas de cómputo sin sistema operativo
- Monoprogramación
- Multiprogramación
- Memoria virtual
- Tipos de proceso
  - . Proceso en lote
  - . Proceso en línea
  - . Proceso compartido

LOS SISTEMAS OPERATIVOS Y SU EVOLUCION

En este capítulo haremos una breve descripción de la evolución de los sistemas operativos con el único propósito de familiarizar al lector con algunos de los tópicos más importantes dentro del proceso EPC, de tal suerte que en la lectura de los siguientes capítulos tenga una idea clara de conceptos como paginación, DISPATCHER, SCHEDULER, SPOOLING, multiprogramación, tiempo compartido, memoria virtual, etc.

Por otro lado, es necesario aclarar que esta tesis se limita a analizar cierto tipo de sistemas de cómputo, pues actualmente en el mercado hay muy diversas clasificaciones (microprocesadores, miniprocesadores, etc.), y a pesar de las diferencias en el poder de cómputo (velocidad del procesador central), de la sofisticación de los sistemas operativos y de la multitud de posibilidades de configuración, no es posible una clasificación clara entre estos sistemas.

Por eso establecemos que en este trabajo se analizarán sistemas mayores, es decir, sistemas de propósito general con características de multiprogramación, manejo de comunicaciones y usuarios de tiempo compartido.

CONCEPTO DE SISTEMA OPERATIVO

El sistema operativo es un conjunto de programas diseñados para el manejo y control de todos los recursos disponibles en un sistema de cómputo, como son la unidad central de proceso (CPU), la memoria central, la memoria auxiliar, los canales, etc. El objeto de este conjunto de programas es optimizar la utilización de estos recursos y lograr el objetivo final del sistema, que es procesar los programas del usuario en forma eficiente.

Herbert Hellerman (HELLERMAN 01) define como objetivos principales de los sistemas operativos, los siguientes:

- Relevar a los programadores de ciertas tareas tediosas como la ubicación específica de archivos en dispositivos de acceso directo, la programación de rutinas específicas E/S para un dispositivo dado y en general, buscar la independencia entre el programa del usuario y los dispositivos de E/S.
- Minimizar la intervención humana en la iniciación y terminación de los trabajos.
- Programar la ejecución de trabajos mediante la utilización de ciertos criterios determinados (clases, prioridades, TIME SLICING, etc.), de acuerdo con los objetivos de la instalación.

. Optimizar la utilización de los recursos del sistema (CPU, - memoria, canales y dispositivos de E/S, etc.)

. Permitir a varios usuarios y trabajos la utilización compartida de los recursos del sistema, así como de sus programas- y archivos.

El diseño del sistema operativo depende mucho de la arquitectura propia del sistema de cómputo y de los objetivos que se le definan, por lo que la eficiencia de un sistema sólo puede ser evaluada en términos de estos objetivos; y en este sentido, los sistemas operativos que hay en el mercado se diferencian entre sí según el equipo para el que están diseñados, las facilidades o ayudas que proporcionan al usuario y la cantidad de recursos que consumen para su -- propio proceso.

Para mostrar los beneficios y características de un sistema -- operativo, a continuación hablaremos brevemente de su desarrollo, - empezando desde luego por las computadoras que no los tenían.

SISTEMAS DE COMPUTO SIN SISTEMA OPERATIVO

Empezaremos por describir algunas características de los programas de aplicación que son la interfase entre el usuario y el -- sistema de cómputo.

La CPU de una computadora puede ejecutar instrucciones y procesar información exclusivamente cuando ésta se encuentra en la memoria principal, y por lo tanto, cualquier programa por ejecutar - debe encontrarse, al menos en parte, en la memoria.

Normalmente los programas están escritos en lenguajes de alto nivel que son traducidos por los compiladores al lenguaje de la máquina. La información descrita en el programa original, llamado - programa fuente, es convertida en direcciones, y las instrucciones escritas en códigos mnemónicos, en instrucciones máquina. Esto da como resultado que el programa fuente se transforme en un programa objeto.

Cuando no hay sistema operativo, los programas del usuario -- disponen de todo el sistema de cómputo para su ejecución, y la programación de los trabajos (SCHEDULING), o sea el orden en que se - ejecutan, depende de la decisión del personal de la instalación.

Además de las instrucciones y datos convertidos al lenguaje - de la máquina por medio de un compilador, los programas manejan in

formación detallada sobre las características de los dispositivos- de E/S durante su ejecución, de tal forma que si es necesario ha-- cer algún cambio de dispositivo para alguna corrida (cambiar una - unidad de cinta por alguna falla, por ejemplo), se debe volver a - compilar el programa. De acuerdo con esto, este tipo de sistemas- es altamente ineficiente, pues al disponer sin compartir de todos- los recursos del computador durante la ejecución de trabajos, se - desperdician recursos, por ejemplo en el caso de programas peque-- ños de larga ejecución que desperdician memoria o en el caso de -- subutilización de los dispositivos de E/S.

SISTEMAS OPERATIVOS DE UNIPROGRAMACION

Como en el caso anterior, en los sistemas operativos de uni-- programación se procesa un solo trabajo cada vez, con la diferen-- cia de que se cuenta con un sistema operativo que tiene, entre o-- tras, las siguientes ventajas:

- Es posible encadenar 2 o más módulos objeto sin que esto - signifique una reprogramación o recompilación.
- Hay programas de servicio (de utilería o UTILITIES), que - permiten ejecutar tareas de propósito general como listar, copiar, clasificar, intercalar, seleccionar archivos, y en general tareas que resultan de mucha utilidad para el desarrollo de aplicaciones y la producción misma del centro de cómputo.
- Se facilita la ejecución de trabajos en secuencia con una intervención mínima del operador.
- Es fácil almacenar programas o rutinas en bibliotecas del sistema que se encuentran en dispositivos de acceso directo, lo cual evita la carga continua de estos, simplificando a través de llamadores, su carga en memoria.
- Es posible llevar una contabilidad de la utilización de - los recursos que consume cada trabajo en el computador.

Para entender otra importante ventaja del concepto de sistema operativo es necesario establecer la diferencia entre un trabajo en un sistema operativo de uniprogramación y un trabajo en un ambiente sin sistema operativo. En este último, un trabajo es exclusivamente un módulo objeto, esto es, un programa. En un ambiente con sistema operativo un trabajo está compuesto de uno o varios pasos (STEPS) cuyo conjunto se llama JOB STREAM (sucesión de trabajos). Para que el sistema operativo reconozca el nombre del trabajo (a efectos de contabilidad, entre otros), los dispositivos de E/S asignados, y otros requerimientos del JOB STREAM, se utiliza una serie de comandos que conforman el lenguaje de control de procesos (JOB CONTROL LANGUAGE, JCL), mediante el cual el usuario define más características de los trabajos al sistema operativo.

Una vez aclarado lo anterior, diremos que la otra gran ventaja del manejo de un sistema operativo es la posibilidad de catalogar los JOB STREAMS en bibliotecas de procedimientos localizados en dispositivos de acceso directo, lo cual permite, como se apuntó anteriormente, que la iniciación y terminación de trabajos sea operada con menor intervención manual de los operadores.

SISTEMAS OPERATIVOS DE MULTIPROGRAMACION

A pesar de las grandes ventajas sobre los sistemas de cómputo sin sistema operativo, es lógico que en un ambiente de uniprogramación haya muchos tiempos muertos en los recursos del sistema.

Si se quiere tener un buen THROUGH PUT (cantidad de trabajo - desarrollado por el sistema de cómputo en un periodo de tiempo dado) se debe procurar utilizar técnicas que minimicen los tiempos - ociosos de los recursos disponibles, lo cual es posible mediante - la operación simultánea de los procesos de entrada y salida con el proceso concurrente de los programas de aplicación y del sistema - operativo.

La simultaneidad de los procesos de E/S y de los procesos de- la CPU se logra mediante la técnica llamada SPOOL: SIMULTANEOS PERIPHERAL OPERATION ON LINE. Esta técnica es posible debido a la - existencia de canales de E/S o procesadores periféricos, dispositivos que funcionan como pequeñas computadores orientadas a la transmisión de la información (instrucciones, datos, etc.) de las unidades de E/S a la memoria central y viceversa. Para iniciar la operación de un canal, se requiere de una instrucción de la CPU, pero una vez iniciada la transmisión, ésta es efectuada por el canal en forma independiente de la CPU, por lo que en el ínterim, ésta pue-

de seguir procesando la tarea activa en ese momento.

Que haya simultaneidad entre las operaciones E/S y el proceso permitió que la CPU pudiera atender varios procesos en forma concurrente, pues mientras una tarea permanecía en espera de que la operación de E/S requerida terminara, la CPU podía ser utilizada en algún otro proceso. A este concepto se le llama multiprogramación.

A continuación describiremos brevemente las funciones del sistema operativo que permiten lograr lo anterior, las cuales tienen gran impacto en el rendimiento de los sistemas de cómputo, y por lo tanto, deben ser cuidadosamente analizadas dentro del proceso EPC.

SPOOLING

Su función especial es almacenar la entrada y salida del sistema en dispositivos de acceso directo (colas de trabajo). Esta función es controlada por los programas escritor y lector del sistema operativo, los cuales permiten que la lectora de tarjetas (o el dispositivo de entrada de que se disponga, terminal, cinta, etc.) y la impresora funcionen simultáneamente. Ahora bien, estos programas del sistema, como cualquier otro, comparten la memoria con el programa de aplicación, por lo que estas operaciones pueden considerarse ya de multiprogramación.

SCHEDULING

Por otro lado, debido a que el SPOOL lee los trabajos a ejecutar y los almacena en unidades de acceso directo, es necesario te--

ner un procedimiento que defina el orden en que los trabajos deben- ejecutarse, para eso es el SCHEDULER del sistema.

El SCHEDULER es el programa del sistema operativo que elige qué trabajo de la cola de entrada debe activarse en el sistema en un momento dado. Una vez seleccionado, el trabajo, o parte de él, es ubicado en la memoria central (ROLL-IN). Si es necesario, definir qué trabajo debe ser desactivado, dicho trabajo, o parte de él pasará - de la memoria central al almacenamiento periférico (ROLL-OUT).

La forma en que el SCHEDULER define qué trabajo activar y cuál desactivar es una estrategia que se define al sistema operativo al- momento de generarlo y que incluye varios parámetros (prioridades,- clases, etc.) que el SCHEDULER toma de cada trabajo.

Estas estrategias pueden ser tan simples como un FCFS (FIRST - COME FIRST SERVED) o algoritmos extremadamente complejos que afec-- tan considerablemente el rendimiento del sistema. Resulta muy complejo definir o predecir, y más aún evaluar, el efecto que puede -- producir un cambio en la estrategia, porque cualquier cambio perjudicará algunas áreas y beneficiará a otras.

DISPATCHING

Hasta aquí hemos visto de manera general la forma de ubicar un trabajo o parte de él en la memoria; sin embargo, ubicarlo en memoria no significa que inmediatamente inicie su ejecución, sino que - competirá por la CPU con los demás trabajos activos.

Se dice que un trabajo está en estado de espera cuando no está siendo ejecutado por la CPU, lo cual no quiere decir que no esté activo, pues puede estar esperando que se termine alguna acción de --

E/S. La función de DISPATCHER es mantener la cola de los trabajos en estado de espera, y por medio de una estrategia de despacho, establecer a cuál de ellos se le asigna tiempo de ejecución.

Al igual que el SCHEDULER, la estrategia que utiliza el DISPATCHER es de gran importancia para el rendimiento del sistema y se define al momento de generar el sistema operativo. Uno de los parámetros principales del despacho es el tiempo máximo de CPU que se otorga a cada trabajo y que se conoce con el nombre de quantum. En este sentido, el quantum es la cantidad máxima de tiempo que la CPU concede a un trabajo hasta que ocurre alguna interrupción (de E/S, por ejemplo). Al final de este tiempo, o antes si hubo alguna interrupción, el DISPATCHER puede asignar otro quantum al mismo trabajo, esperar la respuesta a la interrupción, o asignar la CPU a otro trabajo que esté en la cola de despacho, todo esto de acuerdo con la estrategia definida de antemano.

SISTEMAS OPERATIVOS CON MEMORIA VIRTUAL

Los sistemas de cómputo tienen diversos tipos de dispositivos de almacenamiento que difieren entre sí por su costo por unidad de almacenaje y velocidad, esto significa que los dispositivos de almacenamiento más rápidos (memoria principal) resultan más caros -- que los dispositivos más lentos. Mientras que el acceso a información en memoria se mide en el rango de microsegundos, el acceso a discos se miden en rango de milisegundos. Incluso dentro de la memoria existen áreas de mayor velocidad (BUFFER STORE) ubicadas entre la CPU y el resto de la memoria central (BACKING STORE).

Estas consideraciones han llevado a la utilización de técnicas sofisticadas tendientes al aprovechamiento máximo de la memoria central. Estas técnicas procuran tener el mayor número posible de programas o parte de ellos residentes en la memoria central (ambiente de multiprogramación).

Para que un programa pueda ser ejecutado, se requiere que éste se encuentre ubicado en memoria (al menos en parte) pero muchas veces su tamaño excede el espacio total de memoria disponible, ya sea porque la memoria del equipo es pequeña o porque el área ---- asignada a ese trabajo lo sea. En tal virtud existen técnicas que manejan el envío de copias de partes del programa desde dispositi-

vos de acceso directo hacia memoria y viceversa, y que permiten el procesamiento de programas largos que no caben totalmente en la memoria asignada. La estrategia de selección de estas partes del programa se llama FOLDING y es una operación común en un sistema de cómputo.

En sistemas que no tienen memoria virtual estas tareas son desarrolladas por los programadores con acciones como:

- Dividir un programa largo en varios pasos o programas que sí quepan en memoria para ser procesados secuencialmente.
- Utilización de técnicas que permitan particionar el programa para que solamente estén residentes en memoria ciertas partes.
- Creación de archivos adicionales para tablas que no pueden mantenerse internas en el programa porque lo hacen muy grande, etc.

El esfuerzo que constituye para los programadores el FOLDEAR trabajos aunado a la búsqueda por la optimización en la utilización de la memoria de un equipo, que es uno de sus recursos más costosos y valiosos, es lo que ha dado nacimiento a las técnicas de memoria virtual que constituyen una forma automatizada de FOLDEAR los trabajos entre la memoria central (memoria real) y el almacenamiento secundario (memoria virtual)

Las técnicas utilizadas para el manejo eficiente de la memoria virtual tienen una repercusión directa en el proceso de EPC y deben ser motivo de un análisis profundo, ya que de su conocimiento puede depender el elevar grandemente el rendimiento del sistema.

Paginación

La paginación consiste en dividir los programas en segmentos y estos en unidades fijas, llamadas páginas (2K o 4K, por ejemplo), - para hacer flexible su ubicación en memoria. (FIGURA 2.2)

Veamos en la figura (2.3) la ubicación del programa D en la memoria con este esquema. Las páginas son ubicadas en áreas de memoria llamadas PAGE FRAMES en las que está subdividida la memoria central siempre y cuando éstas estén en estatus libre, esto es desocupadas. De hecho, la memoria virtual (residente en dispositivos de-acceso directo) se encuentra dividida en segmentos y en páginas de tamaño fijo.

Ahora bien, cuando durante la traducción de una dirección del-programa se requiere una página y ésta no se encuentra cargada en - memoria, se acude a tablas externas ubicadas en dispositivos de acceso directo que indican su ubicación en el dispositivo causando una operación de PAGE-IN.

Durante esta operación el sistema busca una PAGE FRAME desocupada para ubicar la página de referencia y actualiza la información de las tablas indicando que se encuentra ocupada y que dicha página se encuentra en memoria.

Sin embargo, en caso de que no se encuentre desocupada ninguna PAGE FRAME el sistema debe decidir qué página saldrá de memoria para ubicar a la demandada.

La estrategia de definir qué página sacar tiene impacto en el-rendimiento del sistema y en general su análisis es de gran impor-- tancia en el proceso EPC. Analicemos qué puede suceder en esta ac-

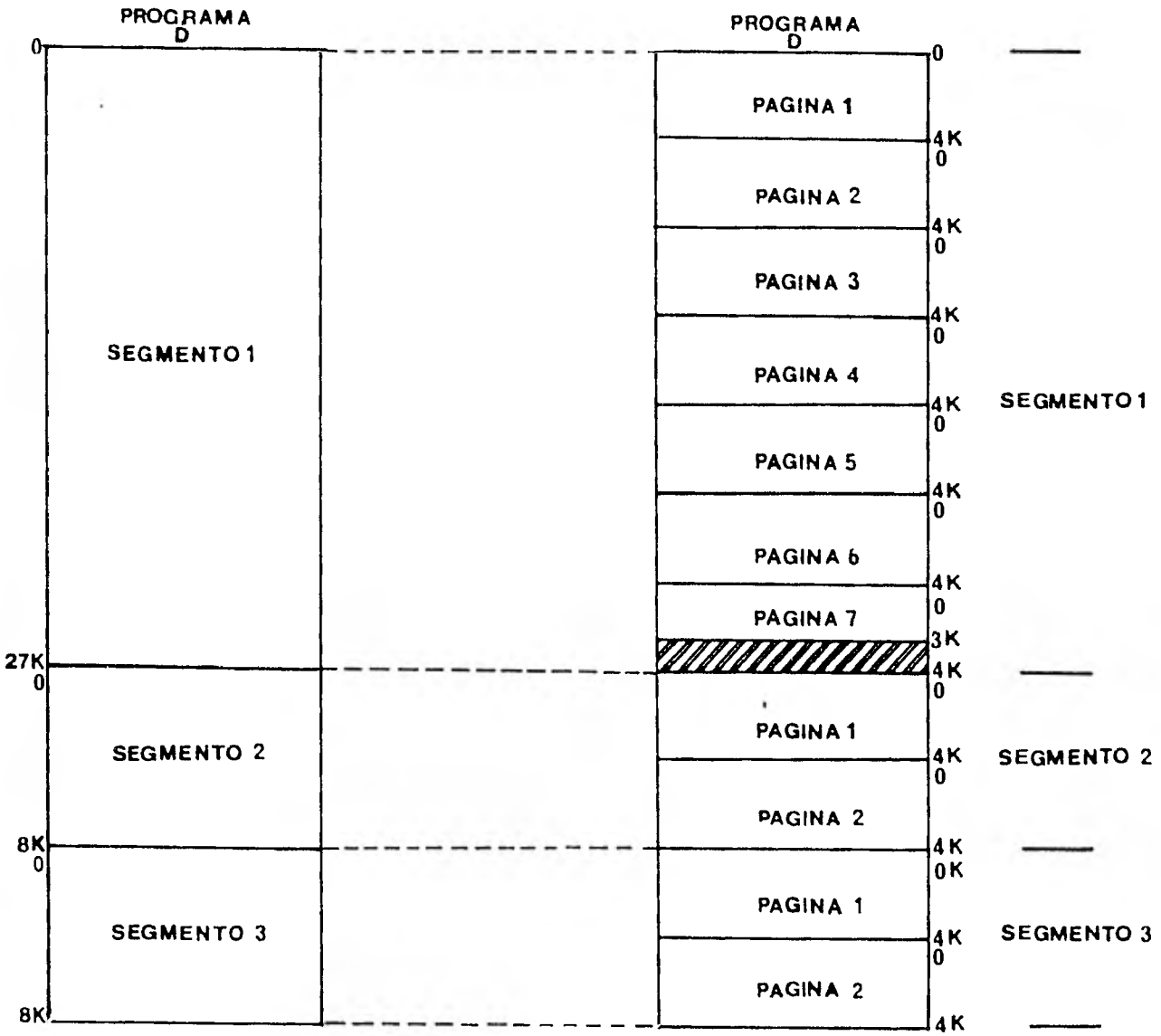

PROGRAMA D
0
SEGMENTO 1
27K
0
SEGMENTO 2
8K
0
SEGMENTO 3
8K
PROGRAMA D
0
PAGINA 1
4K
0
PAGINA 2
4K
0
PAGINA 3
4K
0
PAGINA 4
4K
0
PAGINA 5
4K
0
PAGINA 6
4K
0
PAGINA 7
3K
4K
0
PAGINA 1
4K
0
PAGINA 2
4K
0K
PAGINA 1
4K
0
PAGINA 2
4K
SEGMENTO 1
SEGMENTO 2
SEGMENTO 3

FIG. 2.2

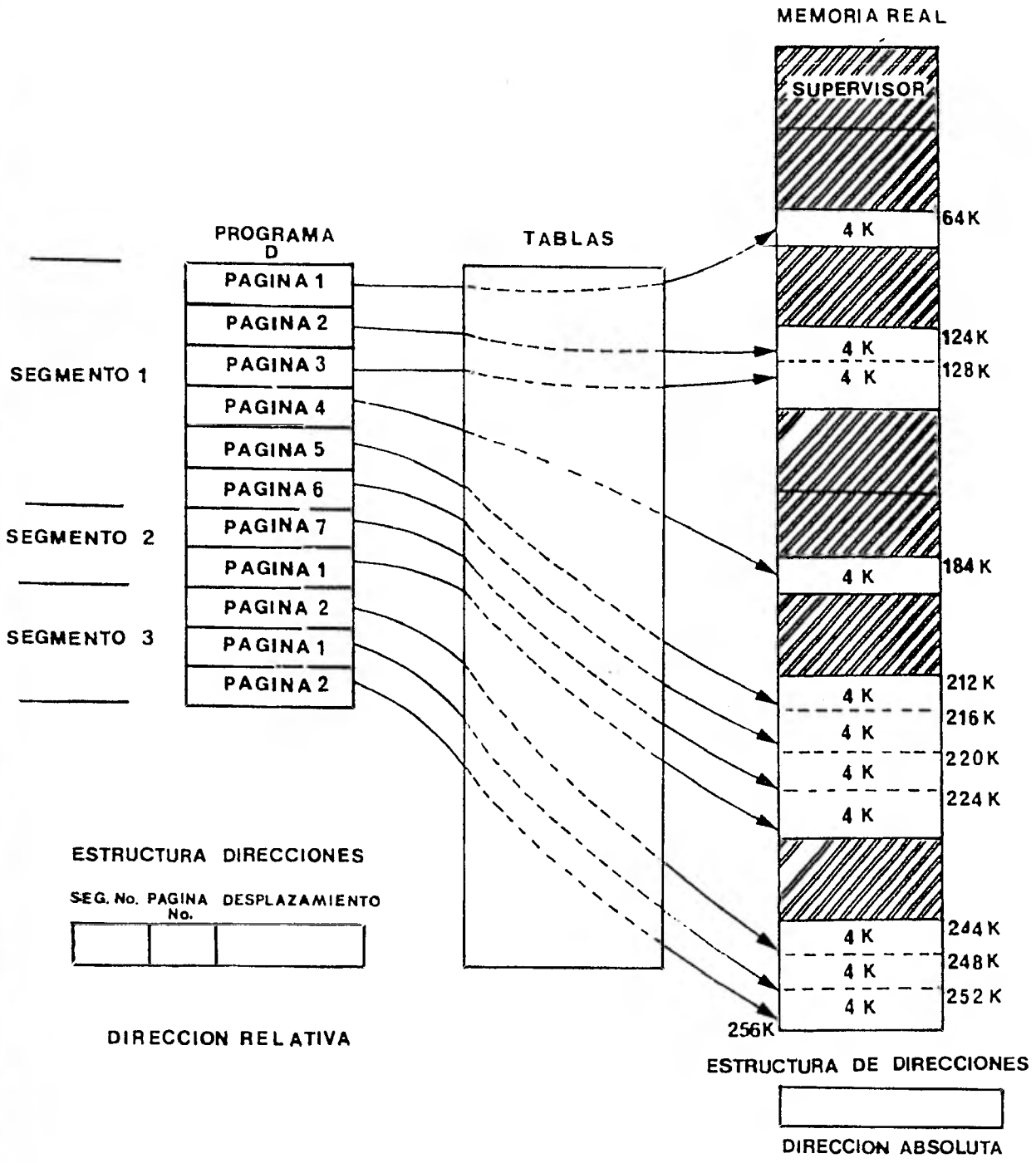

MEMORIA REAL
SUPERVISOR
64K
4 K
124K
4 K
128K
4 K
184K
4 K
212K
4 K
216K
4 K
220K
4 K
224K
4 K
244K
4 K
248K
4 K
252K
4 K
256K
PROGRAMA D
SEGMENTO 1
SEGMENTO 2
SEGMENTO 3
PAGINA 1
PAGINA 2
PAGINA 3
PAGINA 4
PAGINA 5
PAGINA 6
PAGINA 7
PAGINA 1
PAGINA 2
PAGINA 1
PAGINA 2
TABLAS
ESTRUCTURA DIRECCIONES
SEG. No.
PAGINA No.
DESPLAZAMIENTO
DIRECCION RELATIVA
ESTRUCTURA DE DIRECCIONES
DIRECCION ABSOLUTA

FIG. 2.3

ción:

- Se puede reemplazar una página de mucha demanda, por lo que una vez reemplazada, muy rápidamente puede ser vuelta a la- memoria (PAGE-IN), creando tal vez trabajo innecesario al - sistema.
- Se puede elegir para su reemplazo una página cuya información (índices, SWITCHS, etc.), haya sido cambiada desde que fue ubicada en memoria por lo que primeramente se debe almacenar en el almacenamiento secundario donde radica la memoria virtual (PAGE-OUT) y luego operar el reemplazo.

La elección de una estrategia adecuada obedece a un objetivo - básico de optimización del rendimiento del sistema a través de la - reducción de la paginación, la cual se reduce evidentemente mientras mayor sea la memoria central del sistema, ya que se dispone de más - área donde ubicar las páginas de la memoria virtual.

El tener un alto grado de paginación va en contra de la eficiencia del sistema, ya que el estar efectuando una alta paginación - le impide la realización de trabajo productivo.

El supervisor de paginación es un programa del sistema operativo que controla el grado de paginación del sistema mediante la suspensión de uno o varios programas cuando se incrementa la pagina--- ción, de acuerdo con las prioridades asignadas a cada programa.

Esto provoca evidentemente una disminución de la paginación y- el propio sistema puede reactivar el o los programs suspendidos de- acuerdo a niveles que se le establecen al supervisor de paginación- a tiempo de generación del sistema operativo.

Finalmente diremos que la memoria virtual, a pesar de las venta

jas aquí citadas, no es ilimitada, sino que es un sistema que a través de la ubicación dinámica, la traducción dinámica de direcciones y su autorregulación permite la mejor utilización de los recursos del sistema y básicamente de la memoria central.

TIPOS DE PROCESAMIENTO DE LA INFORMACION

En este capítulo describiremos brevemente los tipos más comunes que son procesamiento en lote, procesamiento en línea y tiempo compartido, ya que su forma de operar tiene repercusiones directas en el proceso de EPC, pues sus esquemas son muy diferentes respecto de la utilización de recursos del computador, del concepto de - tiempo de respuesta en cada uno de ellos, de la administración de- los procesos, etc.

Procesamiento en lote

Es el más antiguo. Actualmente está siendo desplazado por el procesamiento en línea para ciertas aplicaciones y por el tiempo - compartido para el desarrollo de sistemas, a pesar de que sus ca-- racterísticas de procesamiento, fundamentalmente los pocos recur-- sos que utiliza, lo hacen muy conveniente para ciertas aplicacio-- nes, por lo cual es difícil que esta forma de procesamiento de información caiga en total desuso.

Como su nombre lo indica, su característica fundamental en el procesamiento de la información es que se forman lotes de transacciones o movimientos que constituyen archivos procesados en bloque sin que el usuario tenga la oportunidad de cancelar, modificar o -

detener algún movimiento hasta que termina el procesamiento de todo el lote.

Este tipo de procesamiento se utiliza cuando no se requiere de una respuesta inmediata; un caso típico son los sistemas de nómina, que al procesar movimientos de faltas, horas extra, etc., no exigen una respuesta inmediata para la toma de decisiones, sino que impactan únicamente la emisión de la nómina quincenal.

En comparación con los sistemas en línea, a los cuales el usuario sí exige una respuesta inmediata, estos procesos tienen poca prioridad, y generalmente son programados para turnos nocturnos, cuando baja considerablemente la demanda de recursos por parte de los usuarios del tiempo compartido o de los sistemas en línea.

Los procesos en lote pueden dividirse en dos grupos, los que forman parte de la producción calendarizada de la instalación y aquéllos alimentados por el personal de la instalación responsable del desarrollo de sistemas, y que fundamentalmente son compilaciones y pruebas de programas y sistemas.

La administración del primer grupo no es complicada, ya que los volúmenes, tiempos y recursos que consumen son conocidos de antemano, así como su calendarización y la distribución de resultados.

El proceso administrativo del segundo grupo, que depende de los usuarios de la instalación, es un poco más delicado, pues debe ser objeto de un análisis que permita programar una secuencia lógica de los trabajos, de tal modo que se reduzca la competencia por los recursos del sistema, así como las intervenciones de los operadores en el montaje y desmontaje de discos, fundamentalmente.

Por último diremos que este tipo de procesamiento es el que - menos recursos del sistema consume, pero que más intervenciones -- "fuera de línea" requiere, desde la preparación de la información- (codificación, captación/verificación, revisión de listados o cifras de control, etc.) en el caso de producción, hasta la planea-- ción de turnos en el centro de cómputo. Por eso, en el proceso de EPC, deben analizarse cuidadosamente los recursos destinados a la- administración de este tipo de procesos y evaluar la conveniencia de utilizar sistemas interactivos, que si bien consumen más recursos del sistema de cómputo, facilitan la labor de los programadores incrementando su productividad en el desarrollo de sistemas y favorecen la obtención de resultados en los sistemas que así lo - requieren.

Procesamiento en línea

El progreso que ha tenido la tecnología de las computadoras - nos hace hablar de millones de instrucciones por segundo, pero dentro del esquema del procesamiento en lote, estas ventajas no son - aprovechadas en toda su magnitud debido a que el tiempo que transcurre entre la generación de los datos hasta la utilización de la- información procesada para la toma de decisiones, es sumamente amplio.

Este aspecto se ha vuelto aún más crítico con la descentralización de las operaciones de las grandes instituciones en que las- sucursales toman mucho tiempo en enviar la información a procesar- desde sus localidades hasta el centro de informática, donde los me

canismos administrativos de control del centro y las propias cargas de trabajo del computador impiden que se atienda de inmediato cualquier operación, y finalmente la instrumentación del envío de regreso de los resultados a las sucursales toma muchísimo tiempo en comparación con el proceso mismo de la información.

Para dar solución a este problema y llevar el poder de cómputo a donde se necesita, se han desarrollado los sistemas denominados "en línea" o "en tiempo real", los cuales permiten que los datos de entrada pasen directamente de su lugar de origen al computador central a través de terminales. Los datos de salida o resultados se transmiten por los mismos medios, en forma directa, a -- donde son requeridos. De esta manera se evitan en gran parte las etapas intermedias mencionadas y se acelera el flujo de datos a/y desde el computador enlazando con líneas de telecomunicaciones a cualquier usuario ubicado a muchos kilómetros de distancia.

Con estos sistemas la administración tanto central como desconcentrada puede ejercer un efectivo control instantáneo sobre sus operaciones en vez de hacer ajustes de difícil coordinación y con grandes intervalos entre sí, aprovechando de esta forma, - de una manera cabal, los avances tecnológicos en cuanto a velocidad en materia de informática.

Esta innovación en el campo de la informática ha dado lugar a un área de desarrollo denominada "teleproceso", que vincula -- las disciplinas de las telecomunicaciones y de la computación.

Los sistemas de teleproceso son de gran importancia en todo el proceso EPC, porque la implementación de estas técnicas obede

ce a requerimientos de alta prioridad en la organización. La planeación de estos sistemas es difícil de llevar a cabo de manera directa, pues deben emplearse modelos estadísticos que describan el arribo de requerimientos o solicitudes hacia el computador central para poder asignar los recursos necesarios (memoria, CPU, etc.) -- y lograr los objetivos de tiempo de respuesta que la institución requiere.

Los sistemas en línea pueden dividirse entre aquellos que requieren una respuesta inmediata (en segundos) y aquéllos que no -- la requieren. Las aplicaciones bancarias de consulta y actualización de saldos en cuentas corrientes son el ejemplo típico de sistemas en línea que requieren una respuesta instantánea. Estos sistemas son también llamados de "tiempo real", que pueden definirse como aquellos sistemas que controlan el medio a través de la recepción y proceso de datos y que actúan o devuelven los resultados -- con suficiente rapidez como para afectar el medio en ese momento.

El otro grupo de sistemas en línea que requieren de una respuesta inmediata es el tiempo compartido, que debido a su importancia y al auge que ha adquirido en los últimos años se describe en el siguiente inciso.

Un ejemplo de los sistemas en línea que no requieren respuesta inmediata es el llamado "entrada remota de trabajos" (REMOTE -- JOB ENTRY).

El RJE es un sistema que permite enviar trabajos y datos de una localidad remota al computador, a través de líneas de telecomunicaciones. Estos trabajos, como cualquier otro, que se alimentan

desde las instalaciones centrales, entran a un esquema de colas de trabajo, prioridades, etc. Una vez que el trabajo ha sido procesado por el computador, los resultados vuelven a la impresora del usuario de la localidad remota por los mismos medios de telecomunicaciones. Cuando su volumen es extenso estos pueden obtenerse también en la(s) impresora(s) central, pues usualmente las localidades remotas tienen impresoras lentas, pues éstas se destinan a volúmenes pequeños de impresión.

Procesamiento en tiempo compartido

Siendo la memoria uno de los recursos más importantes de los sistemas de cómputo, una de las áreas de investigación que más ha avanzado en este campo es cómo utilizar al máximo dicho recurso - procurando dar servicio al mayor número posible de usuarios.

En la figura 2.4 se observan los conceptos de tiempo compartido. Este tipo de proceso consiste en que varios usuarios comparten la memoria y los demás recursos del sistema en forma interactiva mediante la utilización de terminales. Cada terminal activa, usuaria del tiempo compartido, alterna su situación entre "status del usuario" y "status del sistema". Una terminal se encuentra en "status del usuario" cuando la tarea correspondiente a ésta está en estado de espera porque el usuario está tecleando una nueva transacción o leyendo la respuesta. A este tiempo se le conoce con el nombre de THINK TIME. La terminal está en "status del sistema" durante el lapso intermedio entre la introducción de una transacción y la recepción de la respuesta.

ESQUEMA DE OPERACION DEL

TIEMPO COMPARTIDO

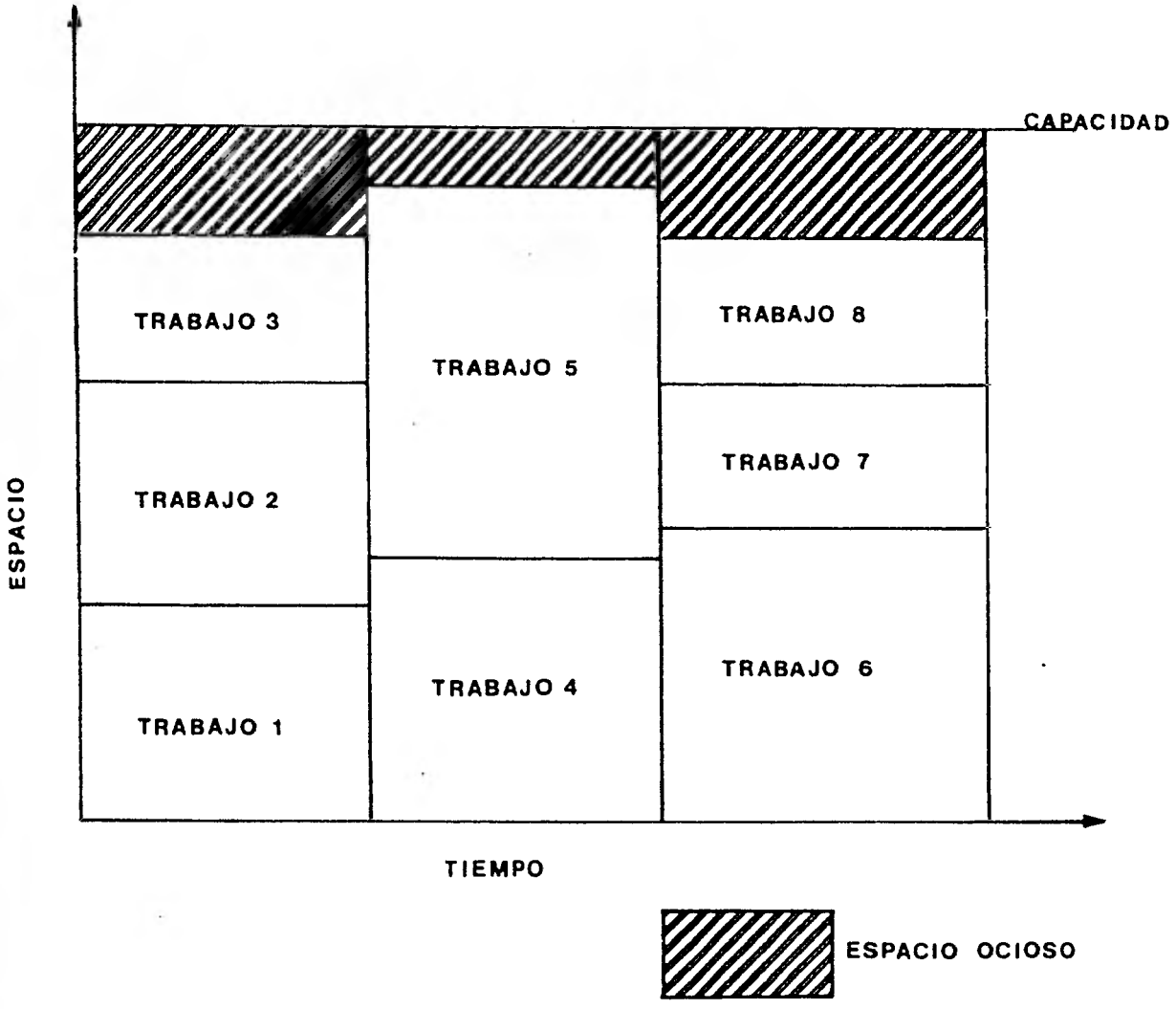


FIG. 2.4

Cuando en el proceso de tiempo compartido una tarea es selec cionada para su procesamiento, el trabajo correspondiente se carga (SWAP-IN)a través de un dispositivo de acceso directo, generalmente más rápido, en un área de la memoria especialmente asignada al efecto. En ese momento el servicio que recibe el trabajo de referencia es idéntico al descrito en el inciso anterior respecto del manejo del DISPATCHER y permanece en la memoria durante un lapso denominado TIME SLICE, cuya duración, definida al momento de generar el tiempo compartido, varía entre 50 y 300 milisegundos. Al término de este tiempo el trabajo vuelve a la memoria auxiliar (SWAP-OUT) dejando el espacio libre para la siguiente tarea.

Al igual que para el SCHEDULER y el DISPATCHER, para que el usuario esté satisfecho en la programación del proceso de tiempo compartido hay una estrategia para definir prioridades de servicio y duración del TIME SLICE que también está relacionada con conceptos como velocidad de la CPU, del SWAP-IN, SWAP-OUT, mezcla de trabajos, etc., y cuyo análisis detallado tiene repercusiones directas en todo el proceso EPC.

Por otro lado, el gran desarrollo del tiempo compartido se debe a sus ventajas sobre el proceso en lote para el desarrollo de aplicaciones, pues incrementa notablemente la productividad de los programadores, aunque el costo en la utilización de recursos del computador es un poco más elevado.

En el MIT se realizó un estudio (SACKMAN 01) con 66 estudiantes de un curso de administración divididos en dos grupos. Ambos

grupos trabajaron con un modelo de simulación de la industria de la construcción, pero uno de ellos utilizó el tiempo compartido y el - otro el proceso en lote.

La información sobre la productividad y actitud de los estudiantes se obtuvo por medio de cuestionarios, análisis de las soluciones presentadas al instructor y la información contable de los re-- cursos del sistema de cómputo utilizados por cada estudiante. Los- principales resultados pueden resumirse como sigue:

1. El costo total (utilización del equipo de cómputo y horas-estudiante invertidas) en ambos grupos fue similar, pero - con diferente distribución. El grupo de tiempo compartido invirtió menos horas-estudiante en la solución del problema, pero el costo del uso de los recursos del sistema fue- mayor.
2. El grupo de tiempo compartido obtuvo soluciones significativamente mejores que las obtenidas por el otro grupo.
3. Más de la mitad de los estudiantes del grupo de proceso en lote no encontró solución adecuada para el problema.
4. Quienes participaron en el estudio prefirieron el sistema de tiempo compartido.

Finalmente diremos que este tipo de herramientas de productividad son importantes en el proceso EPC pues su afinación, es decir - mantener un equilibrio entre la satisfacción de los usuarios del -- tiempo compartido y los recursos asignados a esta tarea, es de capital importancia en el proceso de EPC.

C A P I T U L O III

LA EVALUACION COMO PROCESO ITERATIVO

- La capacidad de un computador
- El computador como conjunto de recursos
- Requerimientos de servicio del usuario
- Tipo y características de las cargas de trabajo
- Disponibilidad y otros factores adicionales
- Proceso sistemático

LA CAPACIDAD DE UN COMPUTADOR

Antes de empezar a describir los elementos que intervienen en la determinación de la capacidad real de un computador explicaremos algunos términos que se utilizarán en este trabajo.

THROUGHPUT: Es la medida que determina la cantidad de trabajo realizada por un sistema. Se mide en términos de número de traba-- jos procesados, tiempo de utilización del procesador, número de --- transacciones procesadas, número de terminales conectadas y ocupa-- des, etc.

El THROUGHPUT real de un computador es normalmente mucho menor que su máxima capacidad teórica debido, en parte, a las ineficien-- cias normales en la administración de un elemento tan complejo, y - por la otra, al gasto de recursos incurrido por todos los programas necesarios para la administración del sistema, por ejemplo:

- el sistema operativo
- los paquetes utilizados para:
  - control de comunicaciones
  - bases de datos
  - tiempo compartido

En la figura 3.1 se muestra cómo llega un momento en que al incrementar la carga de trabajo la cantidad de trabajo productivo dis

RENDIMIENTO TIPICO DE UN SISTEMA EN FUNCION DE LA CARGA

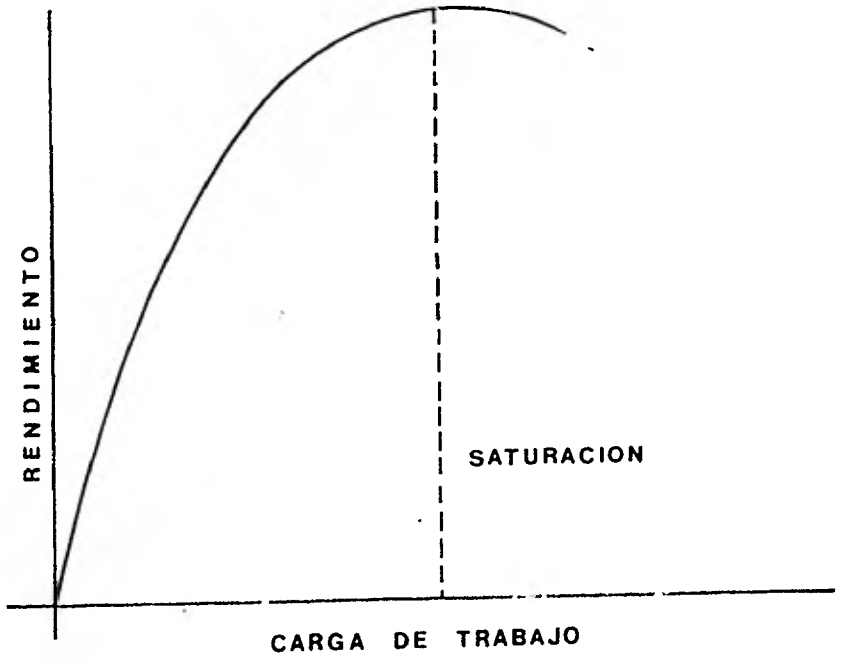


FIG. 3.1

minuye, situación que resulta paradójica. Esto se debe a que conforme se incrementa la carga de trabajo el sistema operativo requiere mayores recursos para la administración del sistema, de tal forma que llega un momento en que la mayor parte del tiempo el sistema lo pasa en administración (SUPERVISOR TIME) y no realizando trabajo productivo (PROBLEM TIME). Un ejemplo de esto son los sistemas con memoria virtual, en los cuales si el número de trabajos excede al máximo recomendable, el sistema pasa mayor tiempo paginando (ver capítulo anterior) que realizando trabajo productivo.

TIEMPO DE RESPUESTA: Es el que transcurre entre el momento en que el usuario libera a ejecución un trabajo o transacción y aquél en que recibe la respuesta.

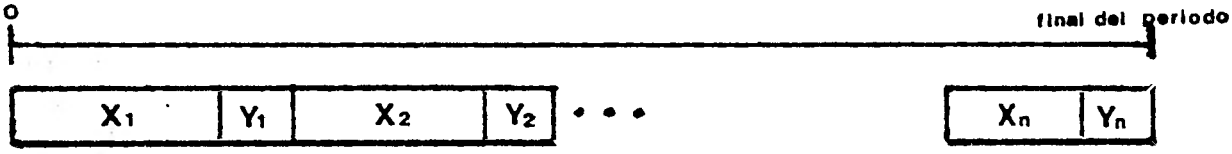
En sistemas con trabajos en lote intervienen algunos otros factores además del tiempo necesario para que el computador ejecute el trabajo e imprima los resultados, por ejemplo tiempo en mesa de control del departamento de producción, tiempo en el departamento que se encarga de administrar la carga de los computadores, tiempo en corte y separación, etc. Por lo tanto, el tiempo de respuesta es mucho menos dependiente de la capacidad del computador que en el caso de los sistemas en línea, por lo que en inglés se ha dado en diferenciar estos dos tiempos de respuesta:

RESPONSE TIME para sistemas en línea y TURNAROUND TIME para sistemas de procesos en lote.

CAPACIDAD PRACTICA DE UN COMPUTADOR: Se define como el máximo THROUGHPUT del sistema cuando éste satisface la calidad y el nivel de tiempo de respuesta requerido por los usuarios, por ejemplo la

capacidad práctica de un sistema en aplicaciones de tiempo compartido es el número máximo de usuarios en línea trabajando simultáneamente manteniendo un tiempo promedio y/o máximo adecuado de respuesta.

DISPONIBILIDAD: Se define como el porcentaje de tiempo que un sistema está en condiciones de ser utilizado para realizar trabajo-productivo.



X_i = lapso iésimo en el cual el sistema está en condiciones de ser utilizado por el usuario.

Y_i = lapso iésimo en el cual el sistema no está disponible para hacer trabajo productivo.

$$\text{tiempo promedio entre fallas} = \frac{\sum_1^n X_i}{n} = \text{tpef}$$

$$\text{tiempo promedio de reparación} = \frac{\sum_1^n Y_i}{n} = \text{tpr}$$

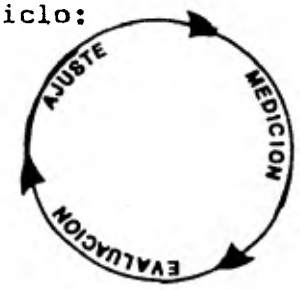
$$\text{disponibilidad} = \frac{\text{tpef}}{\text{tpef} + \text{tpr}} \times 100$$

La disponibilidad de un sistema depende de una serie de factores mencionados más adelante en este capítulo.

AFINACION DEL SISTEMA: Es un conjunto de actividades tendientes a obtener la máxima eficiencia o rendimiento de un sistema con objeto de satisfacer los niveles de servicio requeridos.

Se puede decir que las 3 etapas necesarias para la afinación-

del sistema conforman un ciclo:



En la etapa de medición se utilizan herramientas que nos permiten obtener información del funcionamiento del sistema.

En la etapa de evaluación se analiza la información obtenida y se evalúan los cambios en parámetros del sistema operativo, cambios en la configuración, cambios en la ubicación de los archivos, etc. Esta evaluación se hace siempre pensando en los objetivos de servicio planteados.

En la etapa de ajuste se hacen las modificaciones necesarias - resultado de la evaluación.

EL COMPUTADOR COMO UN CONJUNTO DE RECURSOS

Al hablar de la capacidad de un computador es necesario establecer que el computador es un sistema con una estructura dada compuesto de una serie de elementos relacionados entre sí según un conjunto específico de reglas de interconexión. Cada elemento del sistema es a su vez un subsistema que se compone de elementos de menor nivel y también con una serie de reglas de interconexión entre sus elementos.

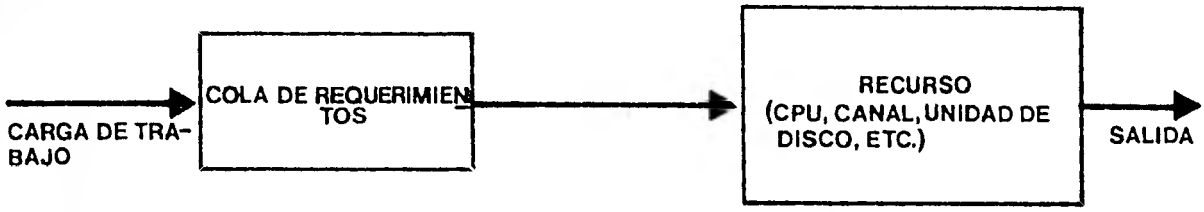
Los componentes del sistema influyen en la capacidad de un computador por sus propias características y por la interacción que tienen entre ellos.

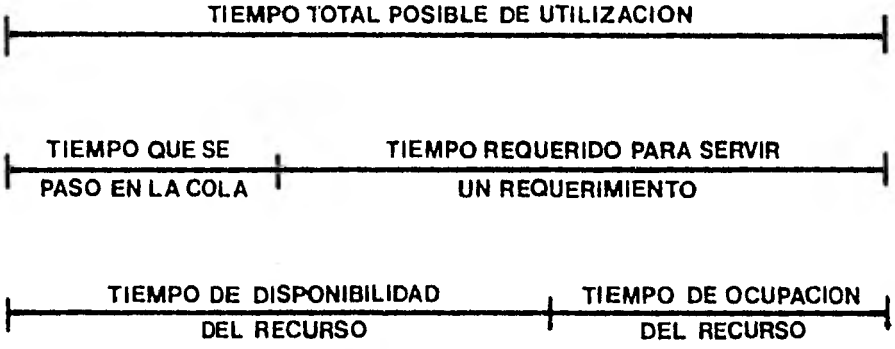
Los factores que influyen para determinar la capacidad de un componente (figura 3.2) son totalmente diferentes de los que se utilizan para determinar la capacidad de todo el sistema.

La capacidad de un componente del sistema visto por sí solo es básicamente dependiente de la velocidad de ese componente.

Esto es, del tiempo que el componente o recurso del sistema necesita para satisfacer un requerimiento. De ahí que si comparamos la suma de todos los tiempos que el recurso necesitó para satisfacer los requerimientos en un periodo (tiempo total de ocupación) con el tiempo total posible de ocupación, podríamos obtener el por-

CAPACIDAD DE UN RECURSO





$$\text{PORCENTAJE DE UTILIZACION DEL RECURSO} = \frac{\text{TIEMPO DE OCUPACION DEL RECURSO}}{\text{TIEMPO TOTAL PROGRAMADO DE UTILIZACION}}$$

Figura 3.2

centaje de utilización del recurso y el de disponibilidad del mismo, y por lo tanto podremos determinar la capacidad disponible del recurso. Cuando el tiempo total de ocupación es igual al tiempo total posible de ocupación, se dice que la capacidad disponible del recurso es nula; esto significa que cada vez que el recurso satisface un requerimiento, tiene ya listo y en espera, otro. En teoría de colas se diría que el recurso está utilizado al cien por ciento, lo cual normalmente daría lugar a grandes colas de requerimientos.

La capacidad teórica máxima de un recurso visto ya como componente de un sistema de cómputo, muy rara vez puede ser utilizada -- porque:

1.- La diferencia de velocidad de los distintos componentes o recursos del sistema hace que los componentes de menor velocidad -- frenen muchas veces a recursos de mayor velocidad.

2.- Puede haber interferencia entre los diferentes componentes que requieren servicio del mismo recurso simultáneamente, siendo que los recursos normalmente sólo pueden resolver secuencialmente las demandas de proceso.

Debido a lo anterior, las características de las cargas de trabajo de una instalación cobran importancia, ya que hay que analizar qué tan factible es que la carga de trabajo pueda ser programada de manera tal que se evite que varios trabajos compitan simultáneamente por un mismo recurso del sistema disminuyendo así la posibilidad de tener cuellos de botella, que en el medio del procesamiento de datos se definen como sobreutilizados, que limitan la correcta utilización del resto de los recursos.

Normalmente el nivel de servicio del usuario se degrada a niveles inadecuados mucho antes de que los recursos del sistema estén - utilizados al 100%.

Como ejemplo presentaremos la siguiente experiencia:

Equipo: IBM

Sistema Operativo: MVS

Recursos analizados: CPU, canal block multiplexor y discos.

Porcentaje de utilización de CPU: 40%

Porcentaje de utilización del canal: 60%

NOTA: MVS (MULTIPLE VIRTUAL STORAGE) es un sistema operativo para - sistemas mayores de IBM que maneja multiprogramación y memo-- ria virtual. Su principal característica es que cada usuario o programa puede utilizar un espacio de hasta 16 MBYTES de memoria virtual al cual se le llama ADRESS SPACE.

En el ejemplo anterior se observa que en teoría la capacidad - disponible de la CPU es 60%. Esto significa 60% de CPU disponible- para algún trabajo adicional, pero como la carga de trabajo de esa- instalación sobreutilizaba el canal creando largas colas de requerimientos, varios de los trabajos se pasaban el tiempo esperando por- operaciones de E/S dirigidas a los discos, y por lo tanto la capacidad disponible de la CPU no podía aprovecharse.

Sin embargo, si la carga de trabajo hubiera podido programarse para balancear adecuadamente los trabajos, por ejemplo trabajos con muchas operaciones de E/S con trabajos orientados más bien a la utilización de la CPU, hubiera podido lograrse un mejor empleo de la - misma.

REQUERIMIENTOS DE SERVICIO DEL USUARIO

En el punto anterior vimos que la capacidad real de los recursos evaluados como parte de un sistema es en muchos casos mucho menor que su capacidad teórica, y que lo que determina el máximo rendimiento de cada componente, y por lo tanto del computador, es la satisfacción del usuario con el tiempo de respuesta. Por eso, antes de empezar a evaluar y proyectar la capacidad de un computador es necesario establecer muy clara y conscientemente los objetivos de servicio, actuales y futuros. El mismo computador con los mismos componentes en varias instalaciones puede tener diferentes niveles máximos de capacidad debido a los diferentes requerimientos de los usuarios con el tiempo de respuesta. Por ejemplo, podría haber dos instalaciones con el mismo computador exactamente, pero que en la primera fuera 20 el número máximo de terminales conectadas a un sistema de tiempo real porque con más terminales concurrentes el tiempo de respuesta dejaría de ser satisfactorio, y que en la segunda este número máximo fuera 35, porque habría menos restricciones en el tiempo de respuesta.

El ejemplo siguiente nos permitirá entender porqué es importante fijar el tiempo de respuesta requerido por el usuario antes de incrementar la capacidad del equipo para satisfacer las demandas

previstas por nuevas aplicaciones y por el incremento de los volúmenes actuales:

Supongamos qua hay una instalación con 4 tipos de usuarios que están satisfachos con el tiempo de respuesta. Sa tiene planeado -- desarrollar un nuevo grupo de aplicaciones para las cuales no hay - capacidad suficiente, por lo que se decide hacer un estudio para incrementar la capacidad del computador sin determinar previamente -- con los usuarios el tiempo de respuesta que requieren.

Como las nuevas aplicaciones no están todavía listas al incrementar la capacidad del computador, como sucede normalmente en estos casos, el tiempo de respuesta mejora notablemente; conforme se van instalando las nuevas aplicaciones, el tiempo de respuesta vuelve a su nivel anterior, y aunque antes del incremento era un tiempo de respuesta satisfactorio, ahora los cuatro usuarios están insatisfechos debido a que nunca tuvieron una idea clara de qué tiempo requerirían para su trabajo.

Ahora bien, entendemos que no es sencillo determinar cuál es - el tiempo de respuesta adecuado, y que normalmente este es un punto de discrepancia entre los técnicos en proceso de datos y los usuarios. Sin embargo, es tan importante este punto que tenemos que encontrar la manera de lograrlo. En la figura 3.3 se detalla un procedimiento práctico y relativamente simple que ha dado resultado en algunas instituciones.

Otras dos formas muy útiles que sirven de complemento al procedimiento anterior son:

a) Basarse en la experiencia de otras instalaciones con aplicacio-

FLUJO RECOMENDADO DE ACTIVIDADES PARA FORMALIZAR EL TIEMPO DE RESPUESTA

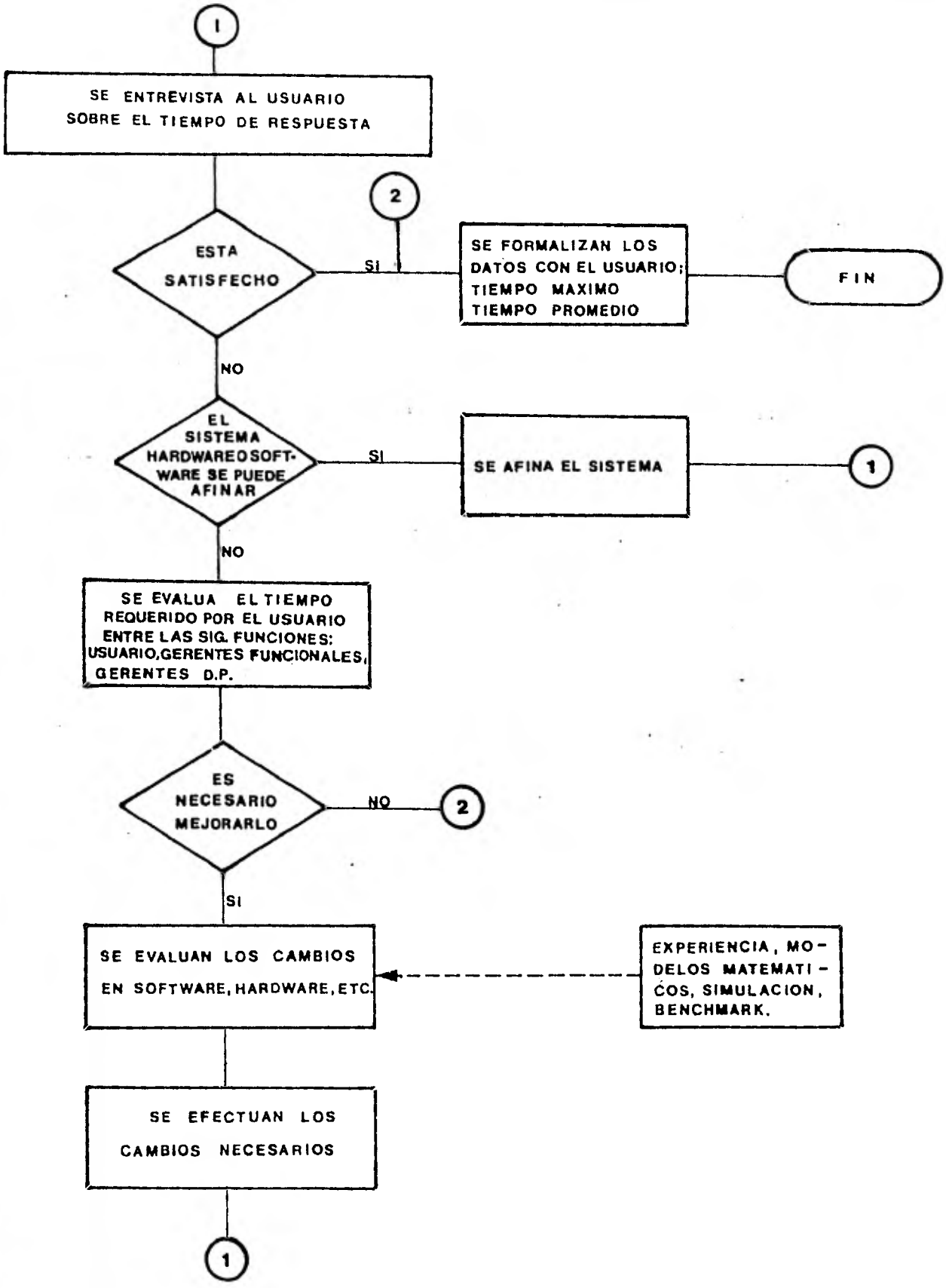


FIG. 3.3

nes similares.

b) Hacer un estudio de tiempos y movimientos y compararlo con la - carga de trabajo del usuario para determinar el tiempo máximo - y promedio.

Cualquiera que sea el método, una vez definido el tiempo de -- respuesta, éste debe ser formalizado con los usuarios y gerentes -- funcionales. El rendimiento y funcionamiento del sistema deben verificarse permanentemente, y el sistema afinarse, mientras sea posible, para cumplir con los compromisos.

Verificar constantemente un sistema permite conocer más profundamente las características del mismo (HARDWARE y SOFTWARE), las de las cargas de trabajo, las herramientas de medición y los reportes- y datos requeridos, y por lo tanto, establecer una base sólida para la utilización de técnicas sofisticadas de proyección y simulación- de la capacidad y rendimiento de un computador.

TIPO Y CARACTERISTICAS DE LAS CARGAS DE TRABAJO

Después de fijar los requerimientos de servicio con los usuarios, el siguiente punto para evaluar o definir la capacidad de un computador es conocer las características cualitativas y cuantitativas de las cargas de trabajo que el computador tiene que procesar.- En aquellos sistemas en los que se conocen, la capacidad del computador se medirá en términos de las unidades de trabajo que realizará en cierto lapso. (Ver figura 3.4)

Generalmente la carga de trabajo en una instalación tiene ciertas propiedades estadísticas que no cambian en periodos razonables, por lo cual es posible:

1.- Definir las características de la carga sobre la base de la distribución de la demanda de los recursos individuales del sistema.

2.- Definir unidades de trabajo y expresar la carga de trabajo en términos de esas unidades.

Unidades de trabajo típicas para las diferentes aplicaciones.

Aplicación	Tipo de Unidad	Medida
Lote	trabajo	trabajos/tiempo (hora, día, etc.)
	transacción	transacciones/tiempo (seg, min, hr)
	consulta	consultas/tiempo
En línea	actualización (sencilla, mediana, compleja)	actualizaciones/tiempo
	comando	comandos/tiempo (seg, min, hr,día)
Tiempo compartido	edición	comandos, edición/tiempo
	compilación	compilaciones/tiempo
	trabajos	trabajos/tiempo

FIGURA 3.4

En la figura 3.5 se describen los parámetros más comunes para la definición de las características cuantitativas de la carga de trabajo; sin embargo, deberán analizarse otros aspectos muy importantes en los estudios de evaluación de la capacidad de un computador, por ejemplo:

- horarios específicos de operación de las diferentes aplicaciones
- número de carretes de cintas y de paquetes de discos que se montarán
- aplicaciones con trabajos que se tienen que ejecutar en forma secuencial (predecesor, sucesor)
- tipo de entrada a las aplicaciones (por ejemplo: grandes volúmenes de tarjetas ocasionan muchos problemas en la lectura)
- grado de dependencia de acciones del operador, etc.

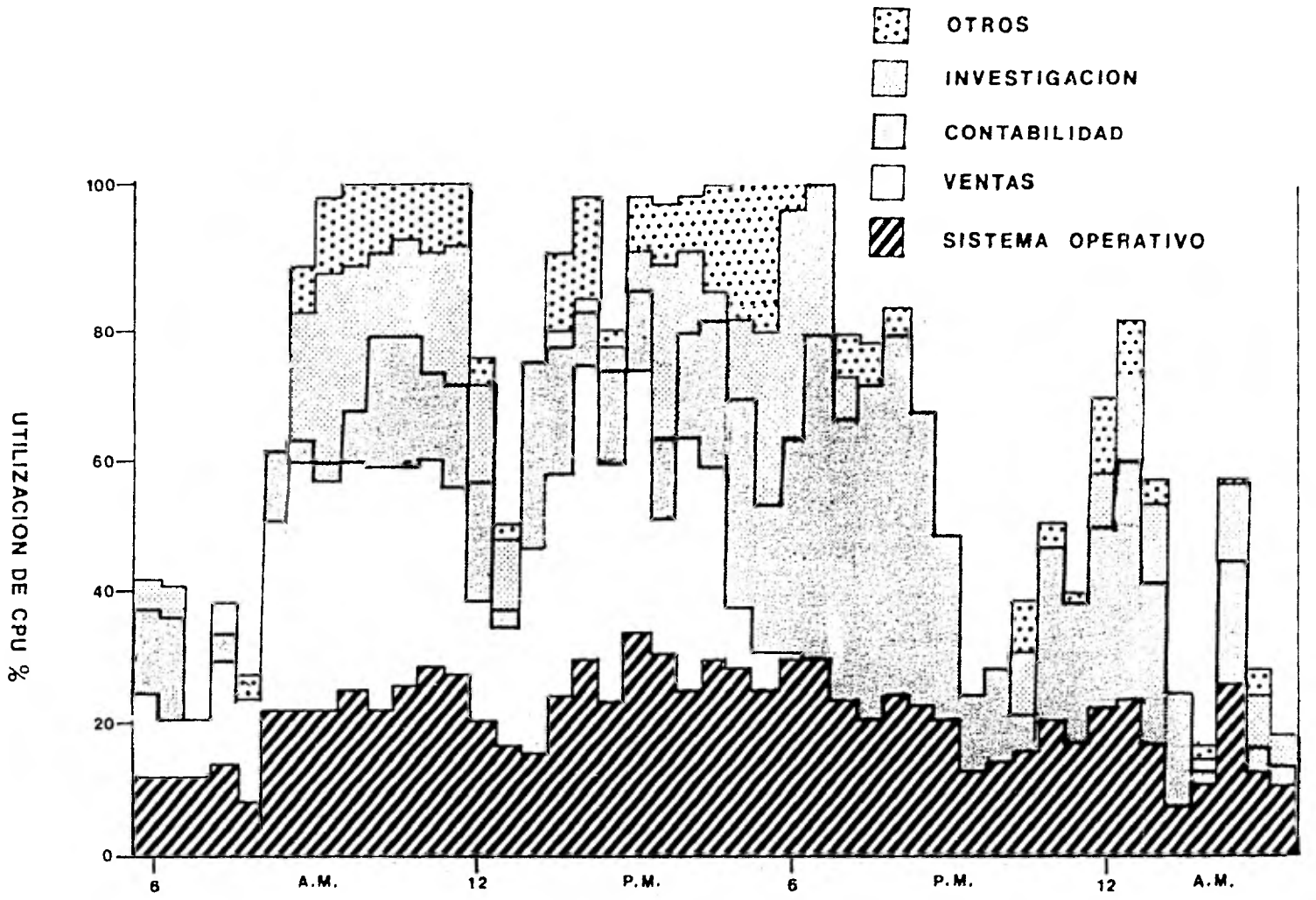
Conocer estas características de las cargas de trabajo es tan importante como conocer los aspectos más técnicos citados en la figura 3.5, sin embargo, en muchos estudios de rendimiento y evaluación de la capacidad no son tomados en cuenta.

Parámetros que definen las características de las cargas de trabajo.

Parámetro	Descripción
Tiempo de CPU por trabajo	Total de tiempo requerido de CPU para un solo trabajo
Requerimientos de E/S	Operaciones de E/S solicitadas para un trabajo
Tiempo de CPU por tares	Tiempo requerido de CPU para - realizar una tarea
Tiempo entre requerimientos	Tiempo entre dos requerimientos de servicio sucesivos
Prioridad	Prioridad asignada a un trabajo
Requerimiento de memoria	Memoria requerida por un trabajo
Promedio de memoria real (WORKING SET)	Promedio de páginas de memoria - real de una tarea
Tiempo de usuario (THINK TIME)	Tiempo promedio que el usuario de una terminal necesita entre transacciones sucesivas
Número de usuarios conectados	Número de terminales que funcio nan simultáneamente
Mezcla de trabajos	Orientación de los trabajos que se procesan simultáneamente (E/S o CPU)
Número de iniciadores activos	Trabajos en lote que se proce- san concurrentemente
Mezcla de instrucciones	Tipo y número de instrucciones por trabajo

FIGURA 3.5

PROMEDIO DE UTILIZACION POR APLICACION DE LA C.P.U. EN UN DIA TIPICO



UTILIZACION DE C.P.U. POR APLICACION

FIG. 3.6

La situación es muy diferente si por ejemplo, en una instala-- ción se analiza el impacto de la carga de trabajo en la utilización de algún recurso, en este caso la CPU, en horarios específicos (ver figura 3.6), que si se analiza la utilización del recurso mediante- los promedios diarios de utilización.

Si en este ejemplo se toma el promedio diario de utilización - de la CPU se cometerá el error de pensar que la CPU todavía tiene - capacidad para el crecimiento previsto de las aplicaciones citadas. Sin embargo, si como sucedió en este caso, la operación de las apli caciones no se puede recorrer a otro horario durante el día para - fines prácticos, no se tiene capacidad adicional para el crecimien- to de las aplicaciones que se ejecutan en las horas pico (8 a 12 y 14 a 19 h).

En resumen, el conocer las características de las cargas de -- trabajo no es más que conocer y entender el medio ambiente de traba jo de una instalación, esto es, la frecuencia requerida de servicio del computador, quién y cómo se hacen los requerimientos de servi-- cio, la cantidad de servicio requerido por usuario para cada compo- nente del sistema, etc.

DISPONIBILIDAD DEL COMPUTADOR Y OTROS FACTORES ADICIONALES

En las instalaciones de proceso de datos hay muchos elementos relacionados entre sí que deben de ser estudiados para poder determinar realmente el rendimiento y la capacidad productiva de una instalación.

Uno muy importante, y que frecuentemente no se analiza con la profundidad debida, es la disponibilidad y productividad del sistema. En este aspecto intervienen varios elementos:

a) Disponibilidad de HARDWARE central (CPU, memoria, discos, canales, etc.). Además de evaluar las fallas de equipo, hay que tomar en cuenta los periodos de mantenimiento preventivo, así como las veces en que se apaga el computador por causas no directamente relacionadas con algún componente del sistema, sino por fallas externas, por ejemplo, de energía eléctrica, de aire acondicionado, o simplemente mientras se hace la limpieza de la sala de cómputo, etc.

b) Disponibilidad del HARDWARE de telecomunicaciones (unidad de control de transmisión, MODEMS, líneas, terminales). Para sistemas en línea este aspecto es muy importante, pues frecuentemente, cuando se analiza la capacidad requerida para cada componente, nos basamos en la capacidad teórica sin pensar que en la práctica muchas veces no es posible lograrla. Como ejemplo relataremos la experien--

cia de una instalación en la que había un sistema en línea con un - computador central y 45 terminales inteligentes distribuidas en el- interior de la República Mexicana. Al hacer el estudio de la capacidad requerida en las líneas de comunicación se llegó a la conclusión de que para algunos casos era necesario rentar un canal de comunicación durante 45 min; teóricamente este tiempo era suficiente- para transmitir todos los movimientos efectuados durante el día en- la localidad remota. Sin embargo en la práctica era totalmente insuficiente debido a los graves problemas de transmisión que hubo -- con los canales de comunicaciones, pues la mayor parte del tiempo - se pasaba en la afinación y puesta a punto del canal, dejando muy - poco tiempo para trabajo productivo.

c) Disponibilidad del SOFTWARE. En este punto hay que considerar- básicamente 3 áreas: el sistema operativo, los paquetes adicionales de soporte (control de comunicaciones, manejo de base de datos, control de tiempo compartido, etc.) y las aplicaciones del usuario.

En relación con este aspecto también pasa desapercibido el análisis de los tiempos perdidos en los procesos y en las recuperaciones de los archivos dañados por problemas con el SOFTWARE.

También en este caso haremos referencia a la experiencia con - una instalación en la que los usuarios permanentemente se quejaban- del tiempo de respuesta de una aplicación en línea. Para encontrar las posibles causas se atacaron 4 áreas de aplicaciones del usuario: sistema operativo, ubicación y organización de los archivos en los- discos (posibles cuellos de botella), el rendimiento de la línea y- los MODEMS de telecomunicaciones.

Sin embargo, al empezar el estudio se detectaron fallas graves en el diseño de las aplicaciones: por un lado no se utilizaba óptimamente el acceso a los archivos y por otro, por las líneas de comunicación se transmitían y se recibían demasiados caracteres innecesarios, lo cual provocaba cuellos de botella.

Cuando se resolvieron estos dos problemas, el tiempo de respuesta se redujo de 29 a 9 segundos en promedio.

d) Operación del sistema. Este aspecto es muy importante para el análisis del rendimiento y capacidad de un computador y mucho se puede ganar participando como observador en días típicos de producción en el centro de cómputo.

En este estudio citaremos solo algunos de los muchos elementos que intervienen en el mejor uso de un computador, pero que pueden dar la pauta al respecto.

1) Ubicación de las diferentes unidades del sistema (cintas, discos, impresoras, etc.)

   Las aplicaciones pueden pasarse mucho tiempo inactivas en memoria esperando que el operador monte y desmonte los volúmenes de cintas y discos necesarios para el proceso.

2) Procedimientos de cambio de turno en operación.

   Durante estos periodos la carga de trabajo alimentada al computador decrece ostensiblemente, y en muchos casos es nula.

3) Organización de los operadores y las consolas dentro del centro de cómputo.

   Este aspecto está muy relacionado con el primer punto, y básicamente se refiere a la rapidez con que puede reaccionar el opera

dor a un requerimiento del sistema.

4) Interacción del sistema operativo y los programas de aplicación con el operador.

Entre más interacción, más se está expuesto a distracciones o - errores.

5) Medios para entrada de los datos y su manejo.

Entre más se dependa de tarjetas para la entrada o salida de -- los datos, más expuesto se está a problemas de lectura o perfo- ración manual de los operadores e interruptores al sistema.

e) Organización del departamento de producción. Este aspecto se - tiene que estudiar sobre todo en el caso de las aplicaciones en lo- te, y deben considerarse los siguientes puntos:

1) Efectividad en la programación de los trabajos: Este as-- pecto es muy importante, ya que una mala programación aumenta el -- riesgo de que las aplicaciones se bloqueen entre sí al competir por los mismos recursos y se creen cuellos de botella que reducen la -- productividad del computador.

2) Preparación y control adecuado de los archivos en cinta: - En muchas instalaciones las cintas no tienen etiquetas, lo cual pue de provocar destrucción de archivos por error y muchos reprocesos.

3) Manejo y control de los datos de entrada y los resultados- del proceso: En muchas instalaciones los usuarios se quejan de que además de tener que esperar mucho tiempo los resultados del proceso, a menudo estos se pierden en el departamento de producción.

f) Procedimientos de respaldo y recuperación de archivos. A este- respecto hay que estudiar el diseño del sistema y el entrenamiento-

de los operadores en relación con los procedimientos de recupera--- ción de archivos.

En las aplicaciones se pierde mucho tiempo si los procedimientos de respaldo y recuperación de información no están bien diseñados o no tienen una difusión adecuada.

Analizar este tipo de situaciones permite evaluar realmente la capacidad, y además hacer sugerencias para optimizar la operación - de un centro de cómputo.

Muchas veces se ha demostrado que es más redituable que el proceso de evaluación de la capacidad sea sencillo y que se estudie -- permanentemente el medio ambiente de producción, que utilizar técnicas de predicción y medición más complicadas.

El encargado de la evaluación deberá ser una persona observadora que no se deje llevar por los resultados de un análisis demasiaco técnico. En la figura 3.7 por ejemplo, se presenta la situación de una instalación en donde se analizó el impacto en la capacidad - disponible del sistema de los inicios no programados por fallas en- el HARDWARE o en el SOFTWARE.

En un primer análisis el tiempo promedio que el equipo estaba- fuera de servicio por fallas se calculó aplicando la siguiente fórmula:

Tpfs = (promedio del número de inicios no programados) (tiempo promedio fuera de servicio)

Sin embargo, al hacer un análisis más profundo se detectó lo - siguiente:

a) Si fallaba un componente del sistema, (no CPU, ni -

CAPACIDAD PERDIDA AL TENER UN INICIO DEL SISTEMA NO PROGRAMADO

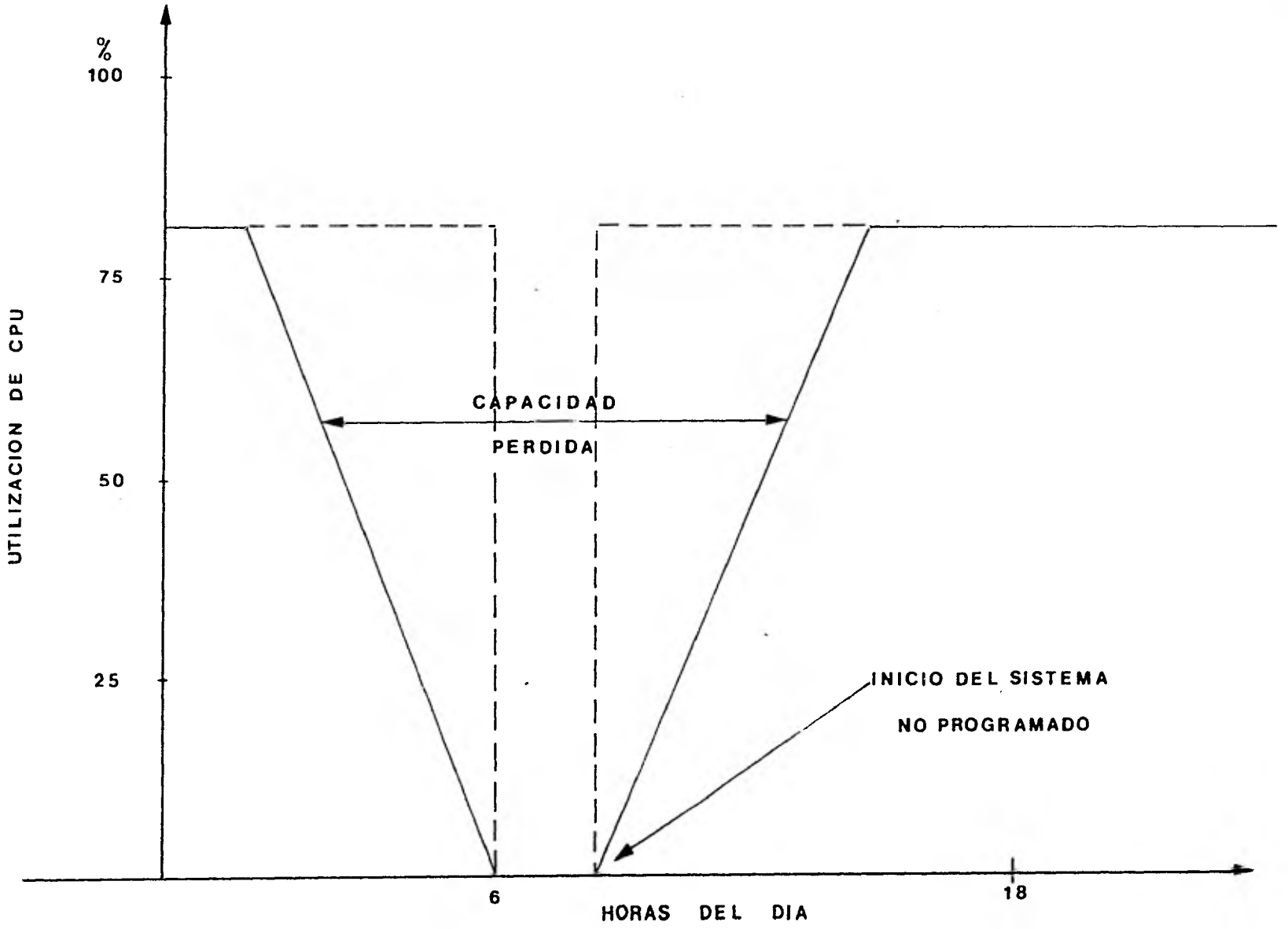


FIG. 3.7

memoria) el tiempo fuera de servicio era el tiempo en que ese componente no estaba disponible.

b) Si la falla se debía a la CPU, a la memoria o al -- sistema operativo, la capacidad perdida del sistema no correspondía nada más al tiempo que permanecía el sistema fuera de servicio, sino que la falla impedía utilizar toda la capacidad del computador - antes de que el sistema quedara fuera de servicio.

c) También se observó que después de suspender el servicio en forma no programada, al reiniciarlo pasaba un tiempo, en - algunos casos mayor que el tiempo de falla, antes de lograr utili-- zar el computador al nivel anterior porque los usuarios de las terminales se levantaban y se ponían a hacer algunas otras cosas antes de reiniciar su trabajo en la terminal.

PROCESO SISTEMATICO

El proceso de evaluación y planeación de la capacidad de la -- mayoría de las instalaciones de proceso de datos se desarrolla en - un ambiente de continuos cambios: cambian el HARDWARE, el SOFTWARE, el personal, las técnicas de diseño y de operación, las cargas de - trabajo... por eso, con el fin de controlar y administrar estos cambios, este proceso debe efectuarse cíclicamente.

Al analizar los aspectos antes mencionados nos damos cuenta de que este proceso es mucho más que una simple recolección de datos - sobre el rendimiento del sistema.

Para que sea realmente productivo debe integrar y controlar muchos elementos, algunos de los cuales son:

RECURSOS

Recursos humanos

- Funcionarios de la institución
  - Usuarios
  - Administrador de la función de D. P.
  - Personal técnico
    - Analistas de aplicaciones
    - Técnico en sistemas operativos

Recursos físicos

- HARDWARE
  - CPU y memoria
  - Dispositivos de E/S y - unidades de control
  - Canales
  - Componentes de teleproceso

- Técnicos en planeación y proyección
- Operadores

- SOFTWARE
  - Sistema operativo
  - Comunicaciones
  - Base de datos
  - Tiempo compartido
  - Aplicaciones

FACTORES QUE DEBEN CONSIDERARSE

. Requerimientos de servicio del usuario

- . Tiempo de respuesta (en lote y en línea)
- . Fechas de instalación de nuevas aplicaciones
- . Percepción del usuario

Disponibilidad

- . HARDWARE
- . SOFTWARE

Capacidad de los recursos del sistema

- . Porcentaje de utilización
- . Promedio de las colas de requerimientos
- . Porcentaje de disponibilidad, etc.

Cargas de trabajo

- . Volumen de transacciones en línea
- . Volumen de programas en lote
- . Horarios por aplicación
- . Orientación (CPU o E/S)
- . Utilización de Disp. E/S, etc.

Herramientas que se utilizarán

- . Medición
- . Reporte
- . Proyección y simulación

Puntos que deben medirse - en cada aplicación

- . Utilización de CPU
- . Utilización de discos, canales y unidades de control
- . Tiempo de respuesta
- . Utilización de memoria real
- . Periodos críticos durante el día
- . Actividades de E/S en los archivos, etc.

Todos estos elementos deben estructurarse y controlarse sistemáticamente para que el proceso de EPC sea efectivo.

En la figura 3.8 se describe el flujo de los datos y la interrelación sugerida entre los diferentes elementos que intervienen en este proceso.

El núcleo de este proceso, descrito en la figura 3.8, es el banco de datos que contiene la siguiente información:

- Volúmenes actuales y previstos de la carga de trabajo
- Datos actuales e históricos del rendimiento del sistema
- Datos sobre el rendimiento previsto del sistema

El proceso de EPC debe implantarse con la conciencia de que en un principio el riesgo de error es grande, pero al hacerlo periódicamente este riesgo va disminuyendo gracias al mejor conocimiento del sistema, de las herramientas y sus posibles deficiencias.

La experiencia nos dice que después de un tiempo este proceso llega a ser de importancia primordial para entender y administrar mejor el complejo ambiente de una instalación de proceso de datos.

FLUJO DEL PROCESO DE E. P. C.

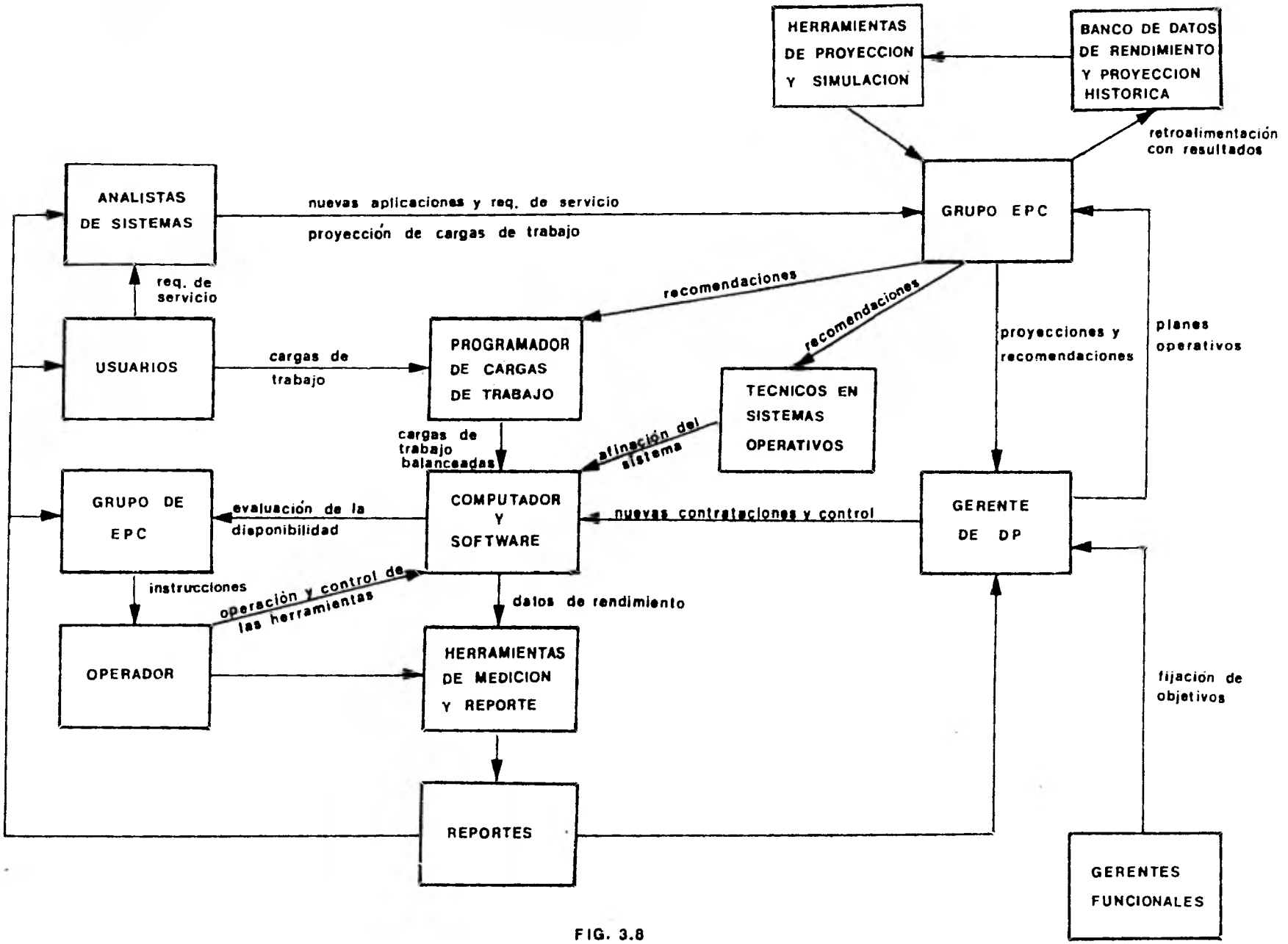


FIG. 3.8

C A P I T U L O S IV

HERRAMIENTAS DE MEDICION Y PREDICCION DEL RENDIMIENTO

- Objetivos de la medición
- Requerimientos de información
- Métodos de medición y tipo de herramientas
- Métodos de proyección de cargas futuras
  - BENCH-MARK
  - Simulación
  - Modelos matemáticos o analíticos

OBJETIVO DE LA MEDICION

El objeto de la medición es recolectar datos sobre el funcionamiento y el rendimiento del sistema para:

- Obtener la información necesaria para llevar a cabo la afinación del sistema.
- Obtener los datos necesarios para estudiar y entender la operación y para establecer una base que nos permita predecir el rendimiento del sistema al incrementar las cargas de trabajo.

Como ya dijimos, es posible entender bastante del rendimiento de un computador y corregir muchos de sus problemas simplemente observando su operación, sin necesidad de utilizar herramientas de medición muy sofisticadas.

Muchas veces la observación directa es la única manera de evaluar los aspectos cualitativos del funcionamiento del sistema. Ya hemos hablado de la importancia de aspectos como la distribución de dispositivos de E/S, de la importancia del operador, y de otros factores que muchas veces parecen demasiado obvios, como la limpieza de la sala de cómputo (el polvo, por ejemplo, causa muchos problemas en los discos y las cintas, lo cual podría influir en el mal funcionamiento de estos dispositivos), la organización y administra

ción del proceso, etc. El primer consejo que se da a un usuario -- cuando reporta que su terminal no funciona es que verifique que esté conectada. Una situación análoga se presenta con el analista -- del rendimiento de un computador cuando se le reporta que éste no - funciona a toda su capacidad: se aconseja empezar la evaluación -- por los detalles más simples y mantener el proceso de recolección - y reporte de los datos en la cantidad mínima necesaria para que no- pierdan su utilidad.

Algunos proveedores tienen gran variedad de herramientas de medición que pueden recolectar datos muy diversos.

Sin embargo, el analista debe mantener un balance entre el volumen de información y la sobrecarga que ocasionan las herramientas al computador, ya que, como es lógico, entre más información se requiera, mayor es el porcentaje de ciclos de CPU distraídos en la obtención de información.

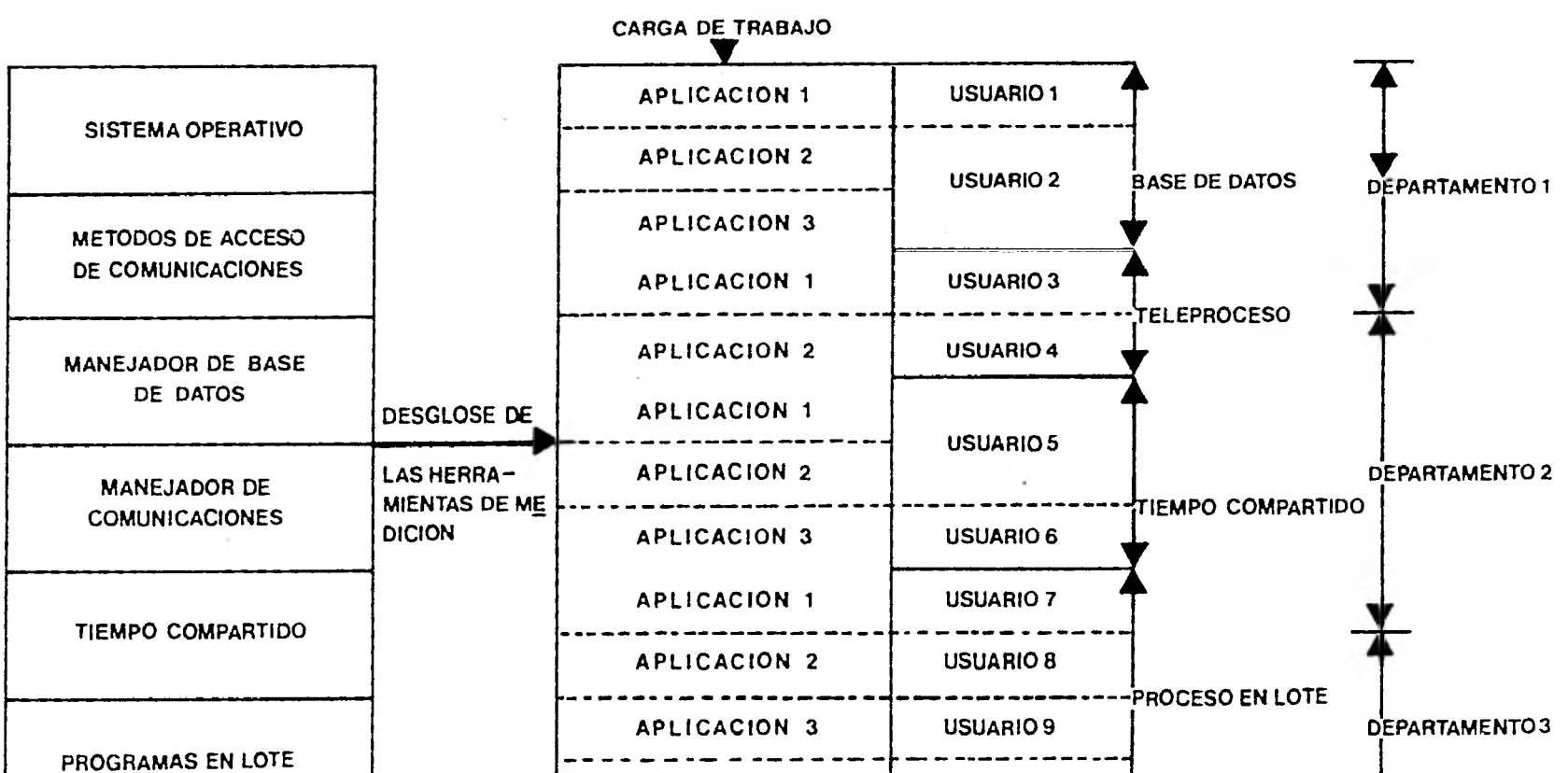
Para que la planeación y pronóstico de la capacidad futura tengan validez es necesario que los datos de funcionamiento a partir - de los cuales se hace el pronóstico sean obtenidos de un sistema razonablemente bien afinado. Por lo tanto, en el proceso de EPC es - imperativo que el sistema se mantenga afinado permanentemente.

Los primeros aspectos que denotan que un computador no funciona adecuadamente son aquellos que el usuario percibe, por ejemplo:

- Tiempos de respuesta (en línea, en lote)
- Disponibilidad del sistema
- Cantidad de trabajo efectuado

Por lo tanto, cualquier esfuerzo de evaluación deberá orientar

DESGLOSE IDEAL DE LAS HERRAMIENTAS DE MEDICION



SUBSISTEMA

- CPU
- MEMORIA PRINCIPAL
- ENTRADA / SALIDA
  - DISCOS
  - CINTAS
  - IMPRESORAS
  - LECTORAS

- TIEMPO DE RESPUESTA
- DISPONIBILIDAD
- UTILIZACION DE LOS RECURSOS (CPU, MEMORIA, CANAL, E/S)
- TIEMPOS PARA MONTAR / DESMONTAR VOLUMENES
- TIEMPOS PARA UBICAR RECURSOS
- TIEMPOS DE IMPRESION, ETC.

FIG. 4.1

se primeramente a la medición y análisis de estos factores, de los- cuales se puede considerar que miden la efectividad del sistema, y- posteriormente, para detectar casos de saturación y cuellos de bote lla, la eficiencia con la que el sistema utiliza sus diferentes com ponentes.

Este trabajo no pretende analizar ni dar ejemplos de todas las herramientas de medición y pronóstico disponibles en el mercado, lo cual sería imposible debido a la gran cantidad que hay; más bien -- describiremos las características ideales de estas herramientas y - los datos típicos necesarios para evaluar los puntos antes citados.

Si bien lo ideal sería utilizar una sola herramienta que permi ta segmentar por usuario la información necesaria para el proceso - EPC (ver fig. 4.1), en la práctica esto se logra utilizando varias herramientas, lo cual dificulta el análisis, ya que, en primer lu-- gar se requieren mayores conocimientos, y en segundo, hay el proble ma de que las herramientas pueden no concordar al medir un paráme-- tro en particular porque no tienen el mismo enfoque de medición. - Por ejemplo, si utilizamos un monitor de SOFTWARE (ver sección Tipo de herramientas) para medir la utilización del canal y comparamos - el resultado con la información obtenida mediante un monitor de --- HARDWARE veremos que los resultados no corresponden, aun cuando es- tas mediciones se hayan hecho en el mismo periodo. La razón es que, a diferencia del monitor de HARDWARE, el de SOFTWARE calcula la uti lización mediante un método que no toma en cuenta los caracteres de control enviados por el canal. Por lo tanto es muy importante cono cer este tipo de detalles cuando se utilizan varias herramientas, y

encontrar algún método que nos permita correlacionar la información obtenida por ellas. La mayoría de los proveedores tienen manuales y guías específicas, dependiendo de la herramienta, que en forma empírica nos permiten ajustar la información recolectada. Por ejemplo, IBM tiene un procedimiento para ajustar el dato de porcentaje de utilización de la CPU llamado CAPTURE RATIOS (COOPER 01) en el que, con base en la experiencia obtenida en muchas instalaciones, se define una guía (ver fig. 4.2) que indica el porcentaje real de utilización de CPU que la herramienta SMF (*) captura para los diferentes tipos de aplicaciones.

(*) SYSTEM MEASUREMENT FACILITY: Paquete de contabilicad de la utilización del sistema utilizado por IBM; obtiene datos como tiempos de utilización de los diferentes recursos del sistema por trabajo, códigos de terminación de cada paso dentro del trabajo, tiempos de permanencia de los trabajos en el sistema.

Porcentajes reales de utilización de CPU capturados por SMF.

	SISTEMAS OPERATIVOS		
TIPO DE APLICACION	MVT/VS1	SVS	MVS
TSO TRIVIAL	0.30	0.25	0.27
TSO DESARROLLO DE PROGRAMA	0.35	0.30	0.32
TSO "FOREGROUND WORK/SPF"	0.40	0.35	0.37
PRUEBA DE PROGR. EN LOTE	0.50	0.45	0.47
PROGRAMACION COMERCIAL	0.65	0.60	0.62
PROGRAMACION CIENTIFICA	0.85	0.80	0.82

TSO (TIME SHARING OPTION): Paquete para manejo de tiempo compartido
MVT, VS1 y SVS: Diferentes versiones de sistemas operativos de IBM

FIGURA 4.2

REQUERIMIENTOS DE INFORMACION EN LA ORGANIZACION

Lo primero que un analista debe hacer al iniciar el proceso -- de EPC es verificar la configuración detallada del equipo y el SOFTWARE de la instalación (ver figs. 4.3 y 4.4).

Es necesario mantener un diagrama actualizado que contenga:

- Modelos del equipo y sus componentes
- Direcciones y/o ubicación de cada componente del sistema
- Ubicación en la memoria real de los subsistemas de SOFTWARE principales
- Nivel de actualización del sistema operativo y paquetes adicionales
- Ubicación de los principales archivos en los dispositivos de acceso directo
- Turnos típicos de operación (ver capítulo sobre implementación).

Una vez definida la configuración, tendrá que hacerse un plan de evaluación por etapas de los componentes del sistema (SOFTWARE y HARDWARE), ya que es conveniente que en un principio solo se eva--- lúen los componentes críticos para no complicar demasiado el proceso EPC.

El mismo criterio se recomienda al seleccionar las herramien--

RECURSOS DE HARDWARE TIPICOS DEL SISTEMA

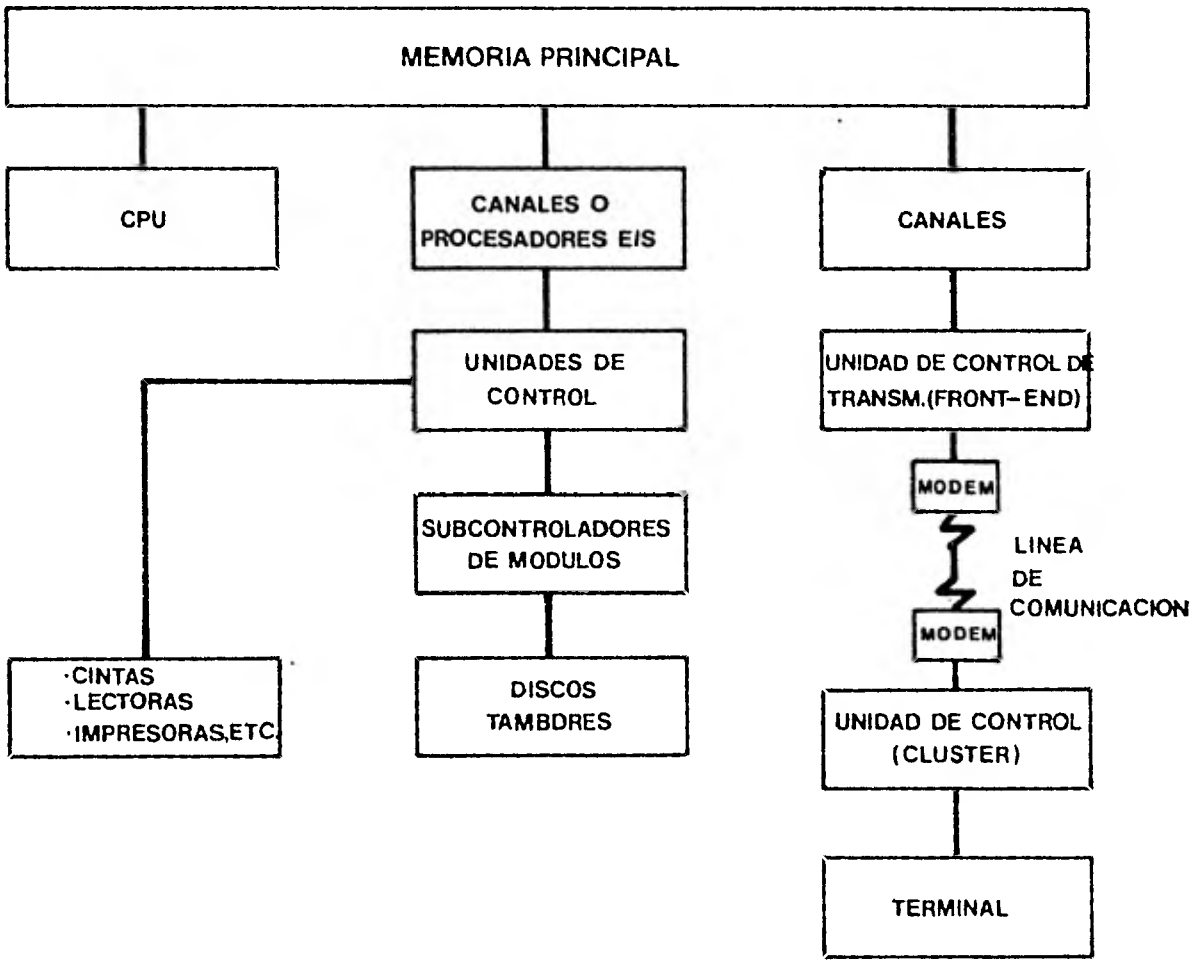


Figura 4:3

COMPONENTES DE SOFTWARE MAS COMUNES

SISTEMA OPERATIVO	MANEJADOR DE COLAS DE E/S (SPOOLING)	PROGRAMACION EN LOTE	PROGRAMAS MATEMATICOS INTERACTIVOS Y PAQUETERIA	PROGRAMACION EN TIEMPO COMPARTIDO Y MONITOR	PROGRAMAS EN TIEMPO REAL Y MONITOR DE TP	MANEJADOR DE BASE DE DATOS	PAQUETES DE RECUPERACION DE TEXTOS

FIGURA 4.4

tas de medición y pronóstico, es decir, que en un principio es recomendable mantener el mínimo necesario de herramientas que permitan obtener resultados a corto plazo y satisfacer los requerimientos de información de todas las funciones involucradas en el proceso EPC:

- Grupo responsable de EPC
- Gerencia
  - de la institución o corporación
  - de los usuarios
  - de informática
    - operación
    - diseño de sistemas
    - soporte técnico

Para poder definir el contenido de los reportes que se proporcionarán a cada grupo es necesario determinar primero el área de interés y después el objetivo de la información, pensando siempre en el tipo de datos que las herramientas más comunes proporcionan.

Cuatro son las áreas de interés en el proceso de EPC:

a) Nivel de servicio
b) Cargas de trabajo
c) Disponibilidad
d) Utilización de los recursos

Por otra parte, la información recabada dependerá de los objetivos específicos que se espera lograr, entre otros:

a) Mantener razonablemente bien afinado el sistema
b) Optimizar la programación de trabajos balanceando las cargas de trabajo.

c) Optimizar los procedimientos operativos

d) Optimizar la programación

e) Conocer la capacidad disponible actual del computador

f) Definir los requerimientos futuros de equipo

A continuación describiremos la información que se recomienda reu-- nir para cada una de las áreas citadas.

a) Nivel de servicio:

En este caso se recaba la información que nos permite definir qué nivel de servicio se proporciona a los principales usuarios -- del sistema.

1) Tiempo de respuesta promedio y máximo por:
   - Usuario (para programas en lote)
   - Transacción crítica (programas en línea y tiempo - compartido)
2) Tiempos totales promedio de ejecución por:
   - Programa crítico (proceso en lote)
   - Transacción crítica (proceso en línea y tiempo compartido)
3) Tiempos de espera promedio y máximo en colas (entrada/salida) por programa o transacción crítica.
4) Total de horas de utilización por periodo y componente crítico del sistema por aplicación crítica, usuario y - subsistema de SOFTWARE. (Tiempo compartido, tiempo real, proceso en lote, etc.)
   - CPU
   - Discos

. Canales

. Porcentaje de memoria real

. Unidades de cintas

. Impresora, etc.

5) Prioridades de despacho por transacción y por programa.

b) Cargas de trabajo

En este caso el objetivo es determinar cómo las cargas de trabajo inciden en la utilización de los recursos y estudiar cómo se comportan éstas a través del tiempo para poder estimar su crecimiento.

1) Horario de ejecución por aplicación crítica

Hay ciertas aplicaciones que solamente pueden ejecutarse a cierta hora, como las aplicaciones en línea o algunas aplicaciones en lote como cámara de compensación, inventarios, etc.

En el capítulo III se explica la importancia de este aspecto de EPC.

2) Total de trabajos procesados, desglosado por:

. Tipo de usuario

. Aplicación crítica

. Función dentro de la institución

. Subsistema (en lote, entrada remota de trabajos, tiempo compartido, operación en línea)

3) Orientación de las aplicaciones críticas:

. Mayor utilización de CPU que de E/S

. Por el contrario, mayor utilización de E/S

4) Características de manejo de los periféricos por aplica

ciones o transacciones críticas:

- Volúmenes de cintas y discos que es necesaio montar
- Requerimientos de formas especiales de impresión
- Relación con el operador. (Proceso automático o necesidad de que el operador decida)
- Volumen promedio de tarjetas perforadas a leer
- Líneas promedio de impresión
- Número de caracteres de los mensajes y bloques de entrada y salida (procesos en línea y tiempo compartido)
- Número de terminales activas promedio por línea de comunicación
- Porcentaje de caracteres que representan títulos de campos o espacios que se envían o reciben por la línea de comunicaciones

5) Utilización promedio de los recursos del sistema por hora/día/mes típicos de producción, desglosado por:

- Tipo de usuario
- Aplicación crítica
- Departamento
- Subsistema

6) Memoria real promedio requerida desglosada por:

- Trabajo crítico
- Subsistema

7) Volumen de transacciones por hora en días típicos y picos de producción desglosado por:

. Aplicación

. Subsistema

8) Volumen de programas en lote que se procesará por día, aplicación y usuario.

9) Diagrama de utilización de la CPU por horas en un día típico de producción y día pico por:

. Aplicación

. Subsistema

10) Volumen promedio mensual de área en disco requerida por aplicación y usuario para archivos maestro y temporales.

11) Variaciones en la programación de las cargas de trabajo:

Este aspecto es importante porque en algunas instalaciones las variaciones en la programación son tan grandes que impiden balancear adecuadamente la carga de trabajo.

c) Disponibilidad

En este aspecto se deberá medir por aplicación y por usuario el porcentaje real del sistema que está disponible para hacer trabajo productivo. Para ello tendrán que evaluarse los siguientes aspectos:

1) Tiempos fuera de servicio:

. Mantenimiento preventivo de la CPU o componentes que impactan el servicio a la aplicación

. Inicios del sistema no programados

. Fallas de componentes que impactan el servicio a una aplicación o usuario aunque el sistema en general funcione.

2) Problemas en la operación del sistema:

. Cambios de operadores al cambiar de turno

. Tiempos perdidos esperando que los dispositivos requeridos por una aplicación estén listos, etc.

3) Problemas por fallas en la información de entrada de una aplicación que hace que se tenga que reprocesar o hace que los programas terminen anormalmente.

4) Fallas en la programación tanto de los programas de la aplicación, como de los paquetes del proveedor o del sistema operativo.

5) Tiempos perdidos en la recuperación de archivos.

6) Tiempos perdidos por fallas en los componentes de los equipos involucrados en los sistemas en línea (líneas, terminales, MODEMS, etc.)

d) Utilización de los recursos

En este caso se mide la eficiencia con la que el sistema administra sus recursos, por lo cual los datos necesarios dependerán del componente del sistema que se esté evaluando.

Por otro lado, como la evaluación depende de medidas internas y toda la bibliografía está en inglés, daremos el nombre del parámetro en inglés y una explicación.

1) Para evaluar la CPU

. "Total wait": tiempo total en que la CPU estuvo sin hacer ningún trabajo en espera de que se terminara de realizar algún evento de E/S o de paginación.

. "Idle wait": tiempo total en que la CPU estuvo sin hacer nin

gún trabajo sin que ningún otro evento de E/S se realizara - concurrentemente.

. "I/Owait": tiempo total en que la CPU estuvo sin hacer ningún trabajo esperando que se terminara algún evento de E/S.

. "Page wait": tiempo total en que la CPU estuvo sin hacer - ningún trabajo esperando que se realizara una operación de "Page-in" o "Page-out".

. Utilización: tiempo total en que la CPU estuvo ocupada entre tiempo total de medición (se mide en procentaje).

. "Problem program time": tiempo total en que la CPU estuvo - ocupada en tareas que no eran del sistema operativo.

. "Supervisor time": tiempo total en que la CPU estuvo ocupada en tareas del sistema operativo.

. "System paging rate": promedio de páginas por segundo -- traídas/almacenadas desde/en memoria virtual para tareas - del sistema operativo.

. "User paging rate": promedio de páginas por segundo traídas/almacenadas desde/en memoria virtual para tareas que no pertenecen al sistema operativo.

. "Swapping rate": promedio de programas por segundo que son traídos/almacenados desde/en memoria auxiliar por haber terminado el "time slice" correspondiente.

. "CPU channel overlap": porcentaje del tiempo de utilización de la CPU que trabajó en forma concurrente con uno o- varios canales.

. "Multiprograming level by time": promedio de trabajos del

usuario que funcionaron concurrentemente.

. "Number of active initiators by time": promedio de iniciadores activos.

2) Para evaluar la utilización de la memoria:

. "Available page frames by time": promedio de páginas reales disponibles.

. "Working set size by task": Promedio de memoria real -- utilizada por tarea.

. "Paging rate": promedio de páginas por segundo traídas/ almacenadas desde/en memoria virtual.

3) Para evaluar los canales:

. "Utilization": tiempo total que el canal estuvo ocupado entre tiempo total de medición. (Se mide en porcentaje)

. "Total Excp's": número de instrucciones de canal ejecutadas.

. "Bytes/channel/time": número de bytes transmitidos por el canal por minuto/hora/día.

. "Channel queue size": promedio de instrucciones de canal que están en cola de espera para ser ejecutadas.

4) Para evaluar los dispositivos de E/S.

. "Utilization": tiempo total que el dispositivo estuvo - ocupado entre tiempo total de medición. (se mide en porcentaje)

- "Excp's/device": número de instrucciones de E/S para -- el dispositivo.
- "Excp's/device/job": igual que el anterior, pero desglo sa la información por trabajo.
- "Bytes/device/time": número de bytes transmitidos por - hora hacia/desde el dispositivo a la memoria.
- "Bytes/excp": promedio de bytes por instrucciones de E/S al dispositivo.
- "Device queue size": promedio de instrucciones de E/S al dispositivo esperando ser ejecutadas.

En este trabajo no se pretende analizar la utilización de esta información para lograr alguno de los objetivos específicos de EPC, más bien pretendemos hacer un análisis de las consideraciones relevantes que forman parte de este proceso y recalcar la importancia - de su formalización al presentar y analizar los principales problemas para su implementación.

Sería imposible dar guías específicas para el empleo de la información, por ejemplo para afinar el sistema, ya que esto depende- mucho del tipo de aplicación, ambiente y objetivos concretos de cada instalación. IBM, por ejemplo, tiene guías específicas para cada paquete:

- IMS/DB/DC (aplicaciones en línea con base de datos)
- CICS (aplicaciones en línea con archivos normales)
- TSO (tiempo compartido)

y dentro de esto, para cada sistema operativo en que funcionan es-- tos paquetes:

- MVS (sistema operativo diseñado para equipos grandes que puede manejar varias memorias virtuales)
- OS/VS1 (sistema operativo diseñado para equipos medianos que maneja una memoria virtual con varias particiones de trabajo)
- DOS/VSE (sistema operativo diseñado para equipos chicos que maneja una memoria virtual con varias particiones -- de trabajo).

METODOS DE MEDICION Y TIPO DE HERRAMIENTAS

Hay tres métodos que nos permiten obtener información sobre el funcionamiento del sistema, las cargas de trabajo y los niveles de servicio:

a) Sistemas de contabilidad: La manera más importante y difundida de obtener información es usar el paquete de contabilidad incluido en el sistema.

Esta herramienta debe servir de base en cualquier proceso de EPC ya que, por una parte, es en general una herramienta conocida y usada, y por la otra, permite desglosar datos específicos como: utilización de CPU, operaciones de E/S; por usuario, aplicación o subsistema de SOFTWARE. Además está incluida en la mayoría de los sistemas operativos de los equipos mayores. Reúne información sobre el empleo de los recursos al inicio o al final de la ejecución, por lo tanto, es la herramienta que crea menor sobrecarga al sistema. Durante el día esta información se graba en un archivo de contabilidad y posteriormente es explotada mediante una serie de programas que obtienen reportes con diferentes grados de sumarización.

En el anexo 1 se muestran ejemplos de estos reportes obtenidos para sistemas IBM.

b) Programas monitores del funcionamiento del sistema (monito

res de SOFTWARE): En el punto anterior vimos que los programas de contabilidad reúnen información en momentos muy bien definidos (al principio y al final de cualquier paso del proceso) y que sirven -- de base en el proceso de EPC; sin embargo, si se necesita informa-- ción más detallada sobre la situación y empleo de los recursos del- sistema durante la ejecución de las diferentes aplicaciones, se tie ne que utilizar otro tipo de herramientas, como los monitores del - funcionemiento del sistema.

Hay dos clases de programas monitores:

1) Monitores de muestreo aleatorio: Como su nombre lo indica, estos programas hacen muestras aleatorias de los valores de las va- riables del sistema, de interés para el analista, las cuales pueden userse para estimar la distribución de estas variables, sus medias- y desviaciones estándar.

Las técnicas utilizadas en teoría de muestreo en estadística - pueden utilizarse para probar la validez y consistencia de los re-- sultados.

Hay algunas herramientas que hacen el muestreo a intervalos es pecíficos, y otras que lo hacen cuando tienen lugar ciertos eventos del sistema (requerimientos de una página (ver memoria virtual), in terrupciones de E/S, etc.)

Al momento de la muestra se leen los valores de contadores y - estatus que permiten tomar una fotografía instantánea de la situa-- ción del sistema en el momento de la muestra.

El resultado de cada una de estas fotografías del sistema se - graba en un archivo que sirve de entrada para un programa estadísti

co asociado con el monitor.

Un ejemplo de la información que ouede grabarse en esta registro sería la situación de los canales (ocupados/desocupados) en elmomento de la muestra. Si el número de muestras es suficientemente amplio se podrá calcular la probabilidad de localizar el canal ocupado y por lo tanto, obtener un buen estimado de la utilización del canal.

Sin embargo, si la frecuencia con que se toman las muestras -- no es adecuada, es importante señalar que podría haber errores en - los resultados obtenidos por el programa estadístico.

Una ventaja de este tipo de programas, comparados con los mencionados en el punto a), es que proporcionan información mucho másdetallada del funcionamiento del sistema, aunqué crean mayor sobrecarga y se corre el riesgo de que haya errores en la información.

2) Monitores por seguimiento (TRACING MONITORS): Los programas seguidores fueron de las primeras ayudas disponibles para corregir problemas de programación. Estos permiten desplegar la situación de elementos críticos (áreas de trabajo, registros, etc.) antes de la ejecución de cada instrucción o de un grupo seleccionadode ellas.

Sin embargo, como herramienta de medición los utilizan casi solamente las casas proveedoras de SOFTWARE, ya que por un lado su empleo exige normalmente modificar el SOFTWARE que se quiere medir, y por otro, realmente representan una gran sobrecarga para el sistema.

Como el objetivo de este trabajo es presentar algo útil para - la mayoría de las instalaciones, no hablaremos en detalle, ni ejem-

plificaremos, estas herramientas.

c) Dispositivos externos al computador: Desde que las primeras computadoras aparecieron en el mercado, los foquitos del tablero de operación fueron diseñados para dar una idea de la situación del -- sistema durante su funcionamiento. Actualmente se usan oscilosco-- pios y otros instrumentos para probar ciertos aspectos del sistema- mientras está funcionando.

Todo esto sentó las bases para que surgiera la idea de utili-- zar contadores electrónicos que son conectados en puntos estratégi- cos del sistema, los cuales permiten medir a cualquier grado de de- talle el funcionamiento de los componentes del mismo.

La mayor ventaja de un monitor electrónico respecto de los -- programas monitores descritos en el punto b) es que, usado adecua- damente, no trastorna para nada el funcionamiento del sistema y -- proporciona más o menos el mismo nivel de información que los moni- tores por muestreo, aunque en este caso se puede tener mayor seguri dad acerca de la veracidad de la información:

F. O. Schulman (SCHULMAN 01) describe en detalle un monitor, - electrónico, construido por IBM, que mediante 256 sensores conecta- dos en puntos adecuados del sistema alimentan 48 contadores cuyo -- contenido se graba en cinta magnética, para evaluar el rendimiento- de un sistema de tiempo compartido.

En México este tipo de herramientas se utilizan muy poco por-- que la mayoría de las instituciones están en la fase inicial (ver - capítulo Implementación) del proceso de EPC; por lo tanto, es un po co difícil encontrar experiencia en esta área. Por eso el analista

tendrá que basarse casi exclusivamente en la bibliografía para utilizar este tipo de herramienta.

J. A. Morris (MORRIS 01) hace un resumen de las características y ventajas de varios monitores electrónicos que se encuentran actualmente en el mercado.

Este capítulo tiene un doble propósito: mostrar las diferentes herramientas que existen para obtener información sobre el funcionamiento actual y previsto del computador, y presentar algunos de los problemas relacionados con la relación, comprensión y veracidad de los datos.

Todos los datos recolectados por medio de herramientas de medición de grandes sistemas de cómputo deberán ser vistos con cierto escepticismo; Sir Arthur Eddington (CHANDRA 01) establece que en astronomía "no se puede creer en observaciones efectuadas antes de que éstas sean confirmadas por una teoría". En el campo de la computación, Saul Rosen (ROSEN 01) establece que "no se puede estar seguro de los datos obtenidos sobre el rendimiento de un gran sistema de cómputo a menos que estos puedan ser confirmados en términos de un modelo conceptual del sistema. Hasta las estadísticas más simples pueden ser tendenciosas".

Como ya se ha aclarado, hay gran variedad de herramientas que permiten medir y evaluar el funcionamiento del sistema; sin embargo es conveniente aclarar que los programas de contabilidad que normalmente vienen integrados al sistema operativo proporcionan en la mayoría de los casos la información necesaria para cubrir gran parte de los objetivos de EPC, y dado que esta herramienta es ya bastante utilizada en la mayoría de las instalaciones para soporte en la administración de los centros de cómputo, es muy recomendable utilizarla sobre todo en las fases iniciales del proceso EPC.

Del resto de las herramientas de medición presentadas, los monitores de SOFTWARE son posiblemente la herramienta más utilizada después de los paquetes de contabilidad. Estas herramientas generalmente no vienen integradas al sistema operativo y por lo tanto hay varias casas proveedoras de este tipo de paquetes. Básicamente estas herramientas se utilizan en actividades de afinación del sistema y por lo tanto son muy dependientes del tipo de subsistema y sistema operativo que se requiere afinar.

Otra área en la cual se utilizan los monitores de SOFTWARE y que cada vez está cobrando mayor importancia es la de su utilización como fuente de información para las herramientas de predicción del rendimiento ya sea para la construcción de los modelos del sistema o para la calibración de los mismos (ver métodos de proyección de cargas futuras y medición del rendimiento en este capítulo).

Los monitores de HARDWARE, como ya se dijo, son muy poco utilizados en México y básicamente se orientan a la evaluación de nuevos modelos de computadora porque no crean sobrecarga al sistema.

METODOS DE PROYECCION DE CARGAS FUTURAS Y PREDICCION DE RENDIMIENTO

Para estudiar el funcionamiento y capacidad de un computador - hay que evaluar una serie de eventos, parámetros y relaciones entre cada uno de los componentes del sistema.

La evaluación se puede hacer a diferentes niveles de detalle,- pero se hace siempre de un cierto modelo del sistema, ya que sería- sumamente complicado hacerlo tomando en cuenta todos los elementos.

Los modelos pueden ser muy simples, como las guías básicas de- seguimiento de operación de un computador a través del tiempo. Por ejemplo, en una instalación se sabía que cuando el número de usua-- rios concurrentes de tiempo compartido excedía de 40, el tiempo de- respuesta era tan alto que había muchas quejas. En este caso se tra taba de un modelo basado en la experiencia que permitía predecir el funcionamiento del subsistema de tiempo compartido en esa instala-- ción.

Al utilizar herramientas de medición como las especificadas en los puntos anteriores se está evaluando solamente un modelo del sis tema y de las cargas de trabajo; esto permite que el analista se -- concentre en los aspectos relevantes del funcionamiento del mismo,- eliminando detalles innecesarios. Por ejemplo, es perfectamente sa bido que en la mayoría de las instalaciones el 80% de los trabajos-

las herramientas normales de medición.

Buchholz (BUCHHOLZ 01) presenta las características básicas de una carga de trabajos sintética, muy utilizada en el medio, para -- evaluar equipos de cómputo y resume su estudio con el siguiente comentario: "el trabajo descrito aquí es un procedimiento muy simplificado que ejercita tanto a la CPU como a los principales dispositivos de E/S controlado por medio de parámetros que regulan el porcentaje de utilización de los recursos y que son especificados en forma independiente del sistema en donde se ejecute". Cambiando los parámetros del trabajo sintético se puede alterar la relación entre utilización de CPU y utilización de E/S.

Sreenivasam y Kleinvman (SREEN 01) presentan argumentos a favor del uso de BENCH-MARKS sintéticos y comentan las ventajas del método, en relación con otros, para estudios de evaluación de la capacidad y afinación del rendimiento, por la facilidad con que se -- pueden construir y porque son mucho más flexibles que los BENCH-MARKS con cargas de trabajo reales.

Algunos de los problemas que presentan son:

1) Es complicado utilizarlos para simular ambientes de proceso complejos, por ejemplo ambientes con sistemas concurrentes de -- tiempo compartido, tiempo real y proceso en lote.

2) Este método, como el anterior, requiere del sistema y de los componentes que se están evaluando, por lo que resulta en algunos casos costoso y complicado utilizarlo para la evaluación de varios sistemas o componentes.

c) BENCH-MARK que simula una carga de trabajo utilizando DRIVERS.

representan aproximadamente 30% de la carga del computador y el 20% restante representa aproximadamente 70% de la carga, por lo tanto el analista deberá concentrarse en estudiar más detalladamente este -- 20% de trabajos, ya que son los que van a dar la pauta en los requerimientos de afinación y de equipo a futuro.

Un modelo del sistema no es más que una representación abstracta de recursos relacionados entre sí. Para que un modelo pueda utilizarse para el proceso de EPC debe proporcionarse la siguiente información:

- Descripción de la carga de trabajo
- Estructura del sistema
- Métodos de programación de trabajos del sistema (SCHEDULING)
- Indices del funcionamiento

En la descripción de la carga de trabajo se establecen las características de ésta, por ejemplo los datos básicos de tiempos de- arribo y ejecución para cada trabajo o transacción (ver cargas de - trabajo en este mismo capítulo).

La cantidad de información requerida y la forma de describir - la carga de trabajo dependerá de la naturaleza del estudio:

- Afinación del sistema
- Depuración de las aplicaciones
- Comparación de propuestas de equipo y SOFTWARE entre varios proveedores.
- Planeación de la capacidad futura
- Optimización de procedimientos de operación, etc.

Hay varios métodos para describir las características de las -

cargas de trabajo y evaluar el impacto de éstas en la utilización - de los recursos:

a) Utilizar un modelo real de las aplicaciones actuales (BENCH-MARK con aplicaciones reales)

En este método se escoge un subconjunto de trabajos de entre - los que normalmente están funcionando en la instalación, y mediante herramientas de medición de las especificadas anteriormente se evalúa el impacto en los principales recursos del sistema.

Hay varios criterios para escoger una carga de trabajo repre-- sentativa:

1.- Los trabajos que corren más frecuentemente

2.- Los trabajos cuya carga representa el mayor porcentaje de utilización de los recursos del sistema

3.- Los trabajos cuyos requerimientos de horario o periodos - de ejecución son críticos

4.- Todos los trabajos que se ejecutan en un turno típico de- operación.

Este método es muy utilizado para las actividades de afinación, optimización de operación y depuración de aplicaciones.

Los problemas más comunes que este método ocasiona son:

1) Trabajos que son escogidos por alguno de los criterios deberían de ser rechazados por otro.

2) Algunos trabajos muy representativos requieren de mucho -- tiempo de ejecución y muchos recursos del sistema, por lo- que es poco práctico y costoso utilizarlos.

3) Es muy costosa la conversión de estos programas como para-

utilizarlos en la evaluación de equipos y sistemas de varios proveedores.

4) En ambientes relativamente sofisticados es altamente costoso transportar catálogos, bibliotecas, bases de datos de un proveedor a otro.

5) Es muy difícil simular en otra instalación las cargas de trabajo de aplicaciones interactivas.

6) Es muy difícil simular las características de las cargas de trabajo a futuro.

7) En algunos casos, para que la información sobre la carga de trabajo sea representativa, es necesario tomar en cuenta intervalos de medición y estudio muy grandes, lo cual hace que este método sea totalmente imoráctico para esos casos.

b) Cargas de trabajo sintéticas (SYNTHETIC BENCH-MARK)

Una carga de trabajo sintética simula la misma utilización de los recursos del sistema que la de una carga de trabajo real, sin que los trabajos que la producen sean trabajos productivos reales. La estructura de demanda de servicios forma el marco del BENCH-MARK sintético. Estas demandas pueden ser obtenidas midiendo el sistema bajo la carga de trabajo real o estimarse para una carga de trabajo proyectada.

Un trabajo sintético puede simular requerimientos de CPU o de E/S (lecturas/escrituras en archivos) mediante interacciones controladas de instrucciones. Un BENCH-MARK sintético es un conjunto de trabajos ensamblados con el único propósito de ejercitar al sistema para que pueda evaluarse su funcionamiento y capacidad a través de

Este método, como los dos anteriores, tiene que ejecutarse en el computador que se está analizando o en uno similar. Un DRIVER no es un modelo de carga de trabajo sino su generador; utiliza como entrada - información específica del medio ambiente y características de la - carga de trabajo que está simulando (SCRIPTS); da como resultado -- transacciones con las mismas características que la carga real o -- proyectada.

Una variante de este generador que se utiliza para evaluar sistemas en lote (FERRARI 01) consta básicamente de dos tipos de pro-- gramas, uno que se encarga de generar instrucciones específicas de- utilización de los recursos que simulan la carga de trabajo del sistema a partir de las distribuciones de utilización recabadas por -- las herramientas de medición y el otro que interpreta estas instrucciones y se encarga de ejercitar los recursos de acuerdo con éstas.

A causa de las dificultades producidas por los sistemas en línea y de tiempo compartido en la evaluación de la carga de trabajo- con BENCH-MARK reales o sintéticos, recientemente se ha puesto mucho énfasis en el desarrollo de paquetes que utilizan este último - método para evaluar este tipo de sistemas. Saltzer describe (SALTZER 01) cómo paquetes diseñados según este método fueron utilizados con gran éxito para medir y proyectar el impacto de la carga de trabajo de un sistema de tiempo compartido en un computador.

Otro ejemplo muy útil del empleo de DRIVER para simular la carga de trabajo de un sistema de tiempo real es el paquete de IBM ---- (TPNS 01) llamado TPNS, que no es más que un conjunto de programas - para simular o reemplazar la carga de trabajo producida por los ---

usuarios de las terminales. Esta simulación de la actividad de las terminales es transparente a los programas de aplicación que procesan las transacciones. Este paquete utiliza como entrada una descripción del medio ambiente del sistema en línea (tiempo para pensar, tiempo para teclear, distribución de transacciones por segundo, etc.) y produce transacciones que los programas de aplicación del computador procesan como si éstas hubieran sido tecleadas por los usuarios.

Este programa TPNS puede residir en la misma computadora en que se encuentran los programas de aplicación, lo cual implica una carga adicional al sistema, o puede correrse en otra computadora (fig. 4.5) que simulará la actividad de las terminales sin crear ninguna carga adicional al sistema que se está evaluando; por eso, esta opción se recomienda cuando el objetivo es evaluar el funcionamiento y los tiempos de respuesta del sistema.

Las principales desventajas de los DRIVERS son:

1) Este tipo de simuladores en general depende mucho del tipo de computador que se está evaluando, por lo que no son una herramienta práctica cuando se quiere hacer una comparación entre equipos.

2) En algunos casos estos paquetes tienen que correr en el sistema que se está evaluando y crean una carga adicional cuyo impacto debe medirse para no distorsionar la evaluación del sistema.

Para evaluar el sistema, los métodos que hemos mencionado hasta ahora requieren del computador o de los componentes que se evalúan y de un modelo representativo de las cargas de trabajo reales-

UTILIZACION DEL "DRIVER TPNS" COMO SIMULADOR DE CARGAS DE TRABAJO

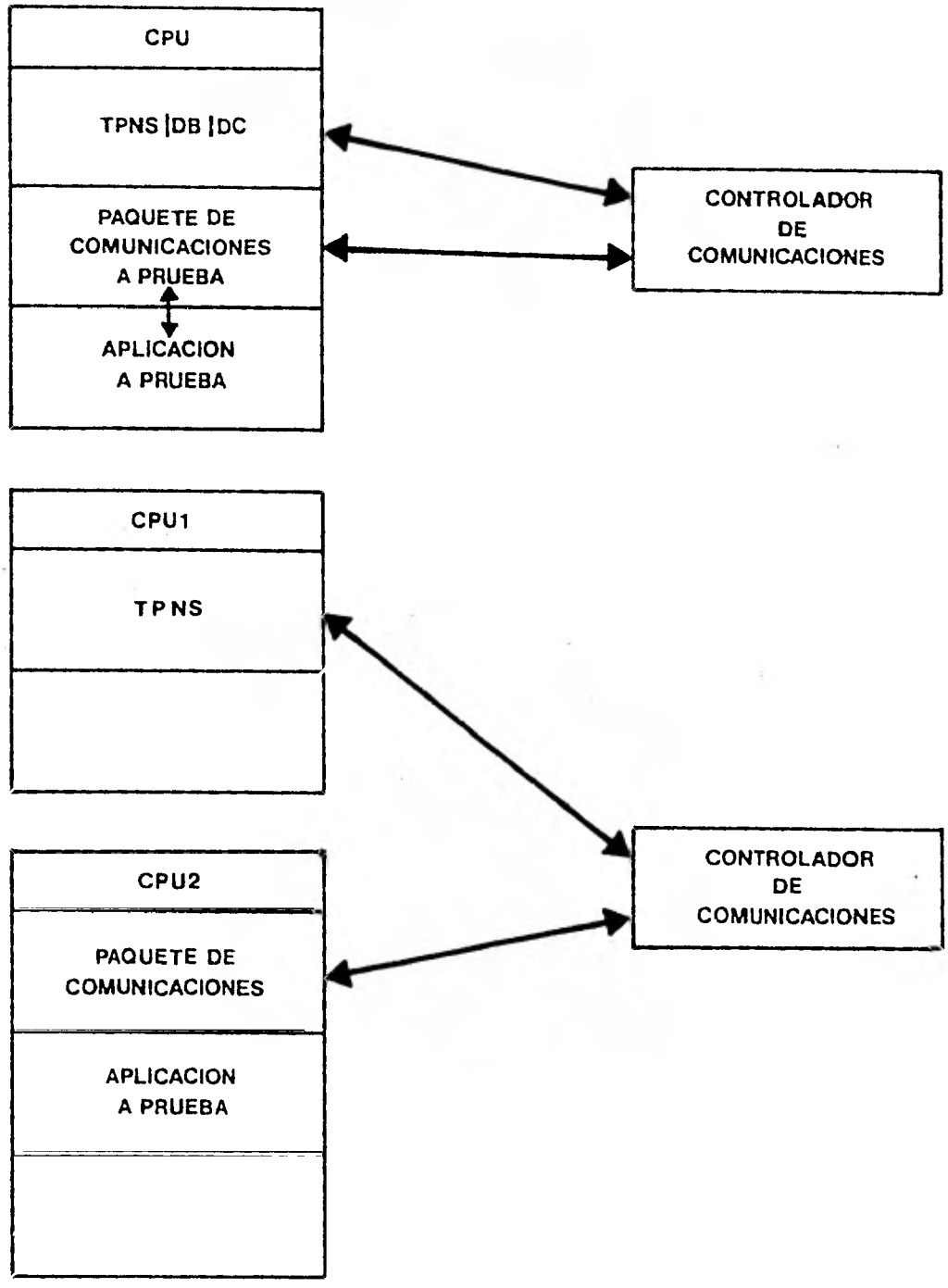


Figura: 4.5

o proyectadas que son alimentadas a trabajos reales o sintéticos.

Normalmente en cualquier estudio relacionado con la función de EPC se consideran varias alternativas, y generalmente sólo una de - ellas puede ser evaluada cada vez, por lo que es muy recomendable - que la carga de trabajo sea fácilmente reproducible.

La carga de trabajo real del sistema, esto es, la carga generada por la comunidad de usuarios en estado normal de producción del- sistema, es generalmente irreproducible, o reproducible a un costo- muy elevado.

Por otro lado, ya hemos visto que para evaluar los efectos de- una carga de trabajo en el computador que se tiene en la instala--- ción o evaluar los efectos de ciertos elementos (parámetros de generación del sistema, ubicación de los archivos en disco, etc.), en - el rendimiento del sistema, las herramientas de medición son el método más eficaz. Desafortunadamente este método (herramientas de - medición y carga de trabajo real) pierde efectividad cuando se tiene que responder a preguntas como ¿qué pasa si hago este cambio en- mi sistema? o ¿cómo mejorar el rendimiento? por las dificultades -- que se plantean cuando hay varias alternativas.

Las técnicas de modelación han cobrado mucha importancia para- la resolución de este tipo de problemas pues permiten gran flexibilidad para especificar la configuración del sistema y las caracte-- rísticas de la carga de trabajo.

Hay dos métodos que utilizan técnicas de modelación para la -- evaluación de sistemas de cómputo; a continuación se describen:

a) SIMULACION: En la mayoría de los casos la técnica de simu-

lación es una combinación de modelación y medición del rendimiento.

El proceso de simulación requiere de un modelo del sistema -- (descripción de las características de los componentes del sistema por simular), de un modelo de las cargas de trabajo (características de las cargas de trabajo) y de un programa que simula el com-- portamiento del sistema (simulador) que da como resultado una serie de reportes para el análisis y predicción del rendimiento.

A este tipo de simulación se le llama discreta, pues el interés se concentra solamente en aquellos eventos discretos que cam-- bian la situación del sistema a través del tiempo.

Los programas simuladores pueden ser escritos en casi cual--- quier lenguaje (Ensamblador, Cobol, Fortran, Algol, Pascal, PL/1,- etc.) o pueden escribirse, y es lo más común, en lenguajes especialmente diseñados para la resolución de problemas de simulación discreta (GPSS, Simscript, Gasp, Aspol, Simula, etc.). Teichroew -- (TECHROEW 01) presenta una comparación muy interesante entre varios de los principales lenguajes de simulación que hay en el mercado y describe sus principales características y desventajas.

La técnica de simulación como herramienta para la evaluación - de sistemas de cómputo está cobrando cada vez más importancia, ya - que tiene una serie de ventajas sobre otros métodos, por ejemplo, - la flexibilidad con la que se pueden variar las características del sistema por simular o las características de la carga de trabajo y- el hecho de que, al menos teóricamente, se puede describir un sistema al nivel de detalle deseado, lo cual es imposible por ejemplo con los modelos analíticos que se describen más adelante.

En el mercado hay varios paquetes de propósito general que -- utilizan esta técnica y que permiten evaluar sistemas complejos de proceso de datos. Como ejemplo podríamos citar dos de las herra-- mientas más utilizadas por IBM para evaluar sistemas orientados a- procesos en línea y proceso distribuido: FIVE y SNAP-SHOT y dos pa quetes, también muy utilizados, desarrollados por proveedores inde pendientes: Scert 70, comercializado por Compress, Inc. y SAM, co mercializado por Applied Data Research. Estos paquetes utilizan - como entrada: 1) una descripción de las características de los com ponentes del sistema (HARDWARE y SOFTWARE) con la cual construyen- un modelo funcional del sistema y 2) las características de los -- programas y cargas de trabajo que ejercitarán al modelo.

Producen como salida una serie de reportes que permiten prede cir el rendimiento del sistema simulado.

La mayoría de los paquetes o programas simuladores reciben la información de las características de las cargas de trabajo y de - los componentes del sistema utilizando uno de los siguientes méto- dos o una combinación de los mismos:

1) Mediante una serie de instrucciones especiales. Como -- ejemplo citaremos un ejercicio efectuado con el simulador FIVE de IBM en el cual se puede apreciar el grado de detalle al que se pue de llegar y el tipo y forma de proporcionar la información.

En este ejemplo se está simulando una red de comunicaciones - (ver fig. 4.6) con 5 terminales tipo pantalla y una impresora len- ta conectada a una sola línea de comunicaciones. Tres de las pan- tallas están dedicadas a consulta y dos a captura. La línea está-

MODELO DEL SISTEMA POR SIMULAR

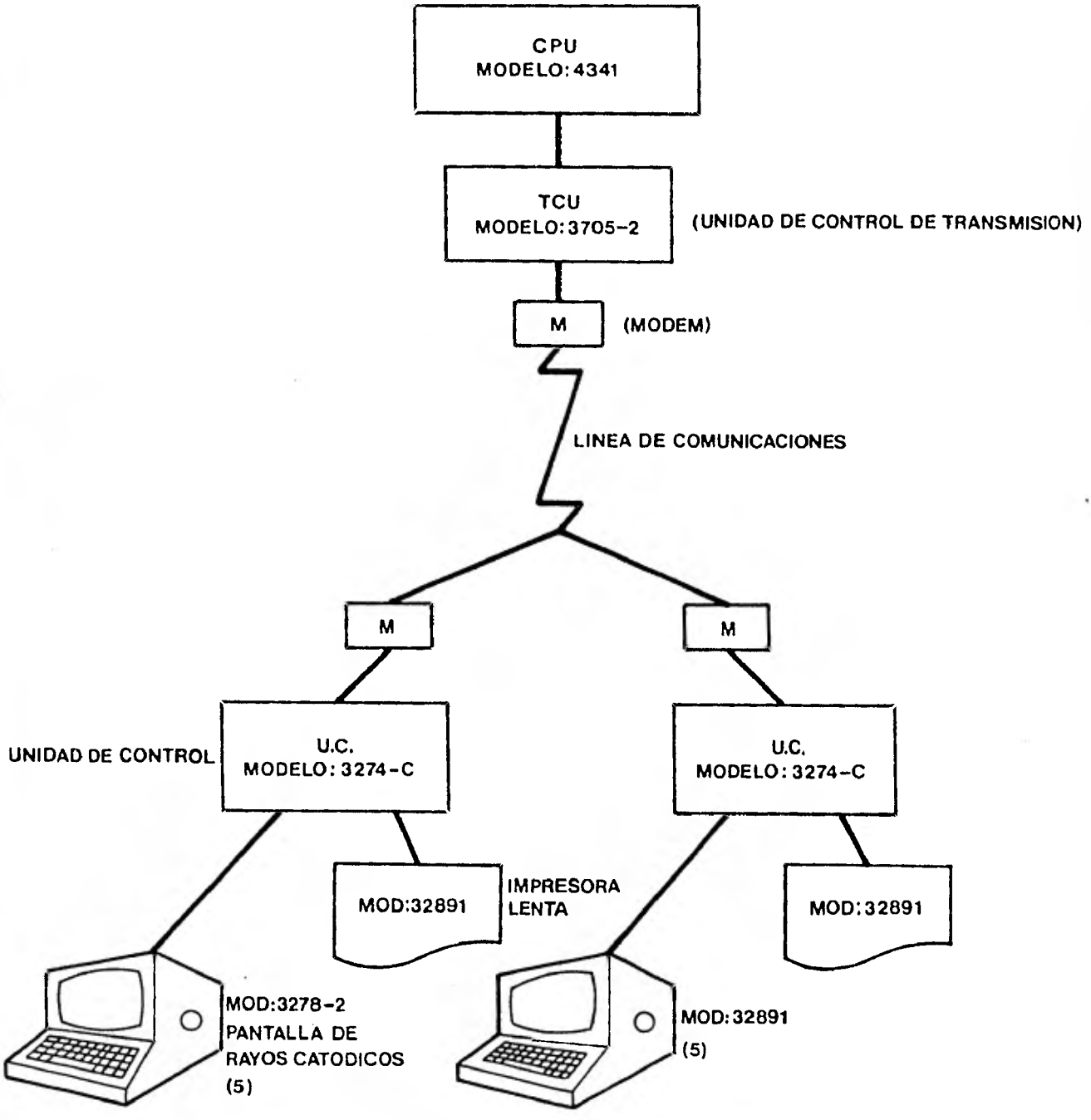


Figura: 4.6

controlada por una unidad de control de transmisión que a su vez se conecta al canal block multiplexor de la CPU.

Un ejemplo de las instrucciones para dar la información de esta configuración y de las cargas de trabajo se presenta en la figura 4.7.

No es el propósito de este trabajo presentar detalladamente y explicar cada uno de los parámetros de la Fig. 4.7 debido al grado de conocimiento e información adicional que se requeriría. Baste decir que el primer grupo de instrucciones definen al simulador los componentes que tendrá que simular, para lo cual se codifica una instrucción por componente definiendo el modelo de éste (el simulador contiene tablas internas que describen las características importantes del HARDWARE de todos los componentes mercadeados por IBM) y las características específicas de los componentes de la instalación, como por ejemplo: tamaño de la memoria, tamaño de los BUFFERS de E/S asignados, características físicas de los teclados para las pantallas, parámetros definidos en el paquete encargado de administrar la red (NCP,NETWORK CONTROL PROGRAM), utilización y ubicación de los archivos del sistema, características de las tareas que se ejecutan en la CPU, etc. El segundo grupo de instrucciones que se relaciona con el primero por medio de la instrucción DEVICE permite simular la carga de trabajo que se alimenta por el dispositivo DEVICE correspondiente.

En el ejemplo la actividad se inicia mandando de la terminal al HOST una transacción de 30 caracteres; en la CPU se simula la ejecución de 5000 instrucciones, después de lo cual se responde a la pan

Ejemplo de los parámetros que describen las características del sistema por simular en el paquete "FIVE":

```
HOST   CPU       MODEL = 4341
TC     TCU       TYPE = 37052, STORAGE = 32, BFRS = 60, UNITSZ = 100,
                       INBFR = 10, CHANTYPE = 1, DELAY = 2,
                       BFRPAD = 28, SLODWN = 12
LN     LINE      LNCTL = SDLC, SPEED = 300, MODEM = (25,3),
                        DUPLEX, CSB1 = 2, PAUSE = 0
C      CU        TYPE = 32741C, NUM = 2, MILES = 100, MODEM = (25,3),
                       PASSLIM = 8, MAXIN = 7, MAXDATA = 265
SINQ   OEVICE    TYPE = 32782, NUM = 3, AWD = INQ, ARRV = 30,
                       INRATES = 3, PACING = (1,1), UPACING = (3,1),
                       APL, CRYPTO
SDE    DEVICE    TYPE = 32782, NUM = 3, AWD = DE, ARRV = 20, ...
PRT    DEVICE    TYPE = 32891, LSCZE = 80, PSIZE = 66, BELT = 94,
                       PBUFFER = 2048, SCS, MAXRU = 1700,
                       PACING = (1,1), CRYPTO


INQ    AWD       TYPE = 1
       SEND      CHAR = 30
       PROCESS   INST = 5000
       SEND      CHAR = 1000, DES = PRT, NEW
       SEND      CHAR = 200
       AEND
```

FIGURA 4.7

talla que inició la actividad con un mensaje de 200 caracteres y un mensaje de 1000 caracteres que se manda a la impresora.

Esto es solamente un ejemplo muy sencillo de la información -- que se puede proporcionar. En la realidad la información debe ser más detallada para que los resultados sean más seguros.

H. C. Nguyen (NGUYEN 01) presenta un documento con las principales características de este simulador.

2) Otro método para proporcionar las características del sistema y cargas de trabajo son las funciones de distribución.

En este caso se asume que las demandas de servicio son variables aleatorias y la carga de trabajo es descrita por sus distribuciones. Las distribuciones reales son frecuentemente aproximadas - por distribuciones exponenciales, ya que las propiedades de esta -- distribución simplifican significativamente el análisis matemático- de los sistemas.

Otras distribuciones utilizadas son: la distribución geométrica, la hiperexponencial o la distribución de Erlang. En general es- difícil ajustar las características de una carga de trabajo a una - distribución, pero los modelos de carga de trabajo basados en estas distribuciones han mostrado que el rendimiento obtenido es suficientemente cercano al obtenido en el sistema real (UNAM 01).

Para aclarar este método citaremos como ejemplo un simulador - muy simple que fue utilizado por la Universidad de Pardue (STEWART 01) para evaluar las características básicas de su sistema de tiempo compartido.

En la figura 4.8 se describe el modelo del sistema. El modelo

MODELO MUY SIMPLE DE UN SISTEMA DE TIEMPO COMPARTIDO

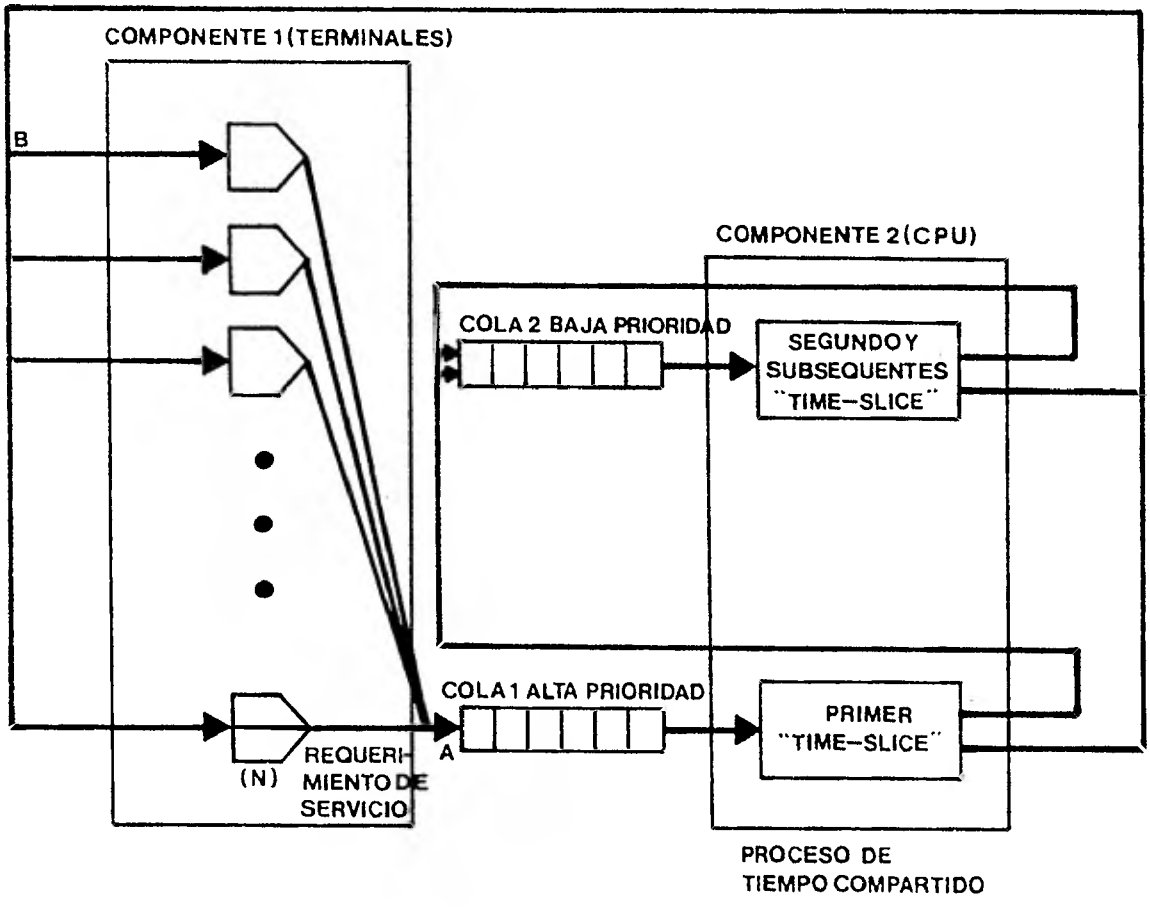


Figura 4:8

asume que el número de terminales activas permanece constante durante todo el proceso. Cada terminal representa una tarea que circula a través del sistema. Cuando una tarea completa un ciclo se considera que la transacción correspondiente ha terminado.

Cuando una tarea entra en el componente 1 (terminal) en B es demorada un determinado número de segundos, seleccionados al azar, de una distribución que representa al THINK-TIME del usuario. Cuando la tarea sale del componente 1 y llega a A, hace un requerimiento de servicio de S milisegundos, donde S es seleccionada, en forma aleatoria, de la distribución que representa al tiempo de servicio (SERVICE TIME); el tiempo de respuesta es el tiempo que le toma a la transacción regresar al componente 1 en B.

De A la transacción va a la cola de espera 1, la cual es atendida con la técnica "primero que entra primero que sale". Cuando esta tarea alcanza el primer lugar en la cola se le da una fracción de tiempo de atención de CPU (TIME SLICE) en el componente 2 (CPU). Este lapso es seleccionado al azar de la distribución que representa al TIME SLICE que el sistema asigna a las tareas. Si esta tarea termina dentro de su TIME SLICE asignado, deja el componente 2 y regresa a B. Si la tarea no ha terminado y finaliza el TIME SLICE que tiene asignado se pone en la cola de espera 2 que tiene el mismo criterio de selección que la anterior, ya que la selección de tareas de la cola 1 tiene prioridad sobre la selección en la cola 2, se selecciona una tarea de la cola 2 si, y solo si, la cola 1 está vacía. Las tareas que requieren de varios TIME SLICE vuelven a la cola 2 cada vez que este TIME SLICE finaliza.

Las distribuciones utilizadas en este método pueden ser mate-- máticas o empíricas.

En el ejemplo anterior se utilizaron las siguientes distribucio nes matemáticas:

- THINK-TIME: se utilizó una función de distribución exponen cial con media de 24 segundos.
- Requerimiento de servicio: 20 milisegundos más una canti- dad obtenida al azar de una función de distribución hiper- exponencial con media de 700 milisegundos, y desviación -- standar de 1200 ms.
- TIME SLICE: 250 ms más una cantidad obtenida al azar de -- una función exponencial con media de 100 ms

3) Otro enfoque utilizado por algunos simuladores consiste en utilizar distribuciones de frecuencia derivadas de la actividad de- medición del sistema, las cuales pueden ser alimentadas al simula-- dor de diferentes formas. Una muy interesante es proporcionando da tos directamente obtenidos por los orogramas monitores que utilizan- la técnica de rastreo (TRACE PROGRAMS).

Este concepto fue utilizado por primera vez en 1969 por Cheng- (CHENG 01): En este enfoque (TRACE-DRIVING-MODELING) los datos ob- tenidos mediante un rastreo TRACE del sistema real son utilizados - para ejercitar al modelo. La información sobre la carga de trabajo y las actividades de los componentes del sistema, como respuesta a- esta carga, se alimentan como entrada al modelo directamente de los resultados obtenidos del monitor

Un rastreo es el registro de eventos seleccionados preservando

la secuencia exacta en que estos eventos ocurren en el sistema.

Los simuladores que utilizan esta técnica son herramientas excelentes para la afinación del sistema (JOSLIN 01). Su principal ventaja es el grado tan alto de detalle en la descripción de la carga de trabajo, de tal manera que la correlación e interferencia de demanda de recursos en el sistema real puede preservarse con suficiente precisión.

Consideraciones sobre la técnica de simulación

La simulación tiene características que la hacen una herramienta muy atractiva para el proceso de EPC. Los lenguajes de simulación simplifican mucho la escritura de simuladores y, al menos teóricamente, es posible simular un sistema de cómputo a nivel de detalle, lo cual permite mayor seguridad en la veracidad de los resultados.

Rosen (ROSEN 01) considera que "...el uso de la simulación combinada con el uso de técnicas de medición (para proporcionar los datos de entrada y para validar el modelo) como una área prometedora y productiva para investigación y análisis del medio ambiente de sistemas de cómputo".

Sin embargo, se presentan algunas dificultades muy serias que hacen que algunos autores y analistas se muestren escépticos en cuanto a los resultados que se pueden obtener con esta metodología. El principal problema con este método es la gran cantidad de recursos de cómputo necesarios para realizar ciertos estudios. La experiencia demuestra que el proceso de desarrollo, prueba y validación

de los resultados de un modelo de simulación de un sistema complejo puede ser un proceso largo y costoso.

Un estudio profesional del comportamiento de un sistema exige estudiar el impacto que producen muchos parámetros independientes unos de otros. En estricto rigor, el número de veces que el simulador debiera ejecutarse es el producto de los diferentes valores que puede asumir cada parámetro.

La complejidad para implementar el modelo es muchas veces de la misma magnitud que la del desarrollo del sistema real, y los problemas que se presentan al probar e implementar un programa de simulación son de la misma magnitud que los que se tendrían para probar e implementar el sistema real. Huesmann (HUESMANN 01) establece que "quizás la razón principal de la popularidad de la técnica de simulación en estudios sobre rendimiento de un computador es que no hay otra alternativa adecuada".

De estos últimos comentarios podemos concluir, en resumen, que esta alternativa puede ser muy útil como herramienta para ciertos estudios, aunque no se debe perder de vista el costo y el esfuerzo que exige.

b) MODELOS MATEMATICOS O ANALITICOS: El objetivo de un modelo matemático es abstraer algunas de las características esenciales de un sistema y encontrar las relaciones entre los parámetros del modelo que se prestan a un trato matemático. Los sistemas de cómputo son estructuras extremadamente complejas que no se prestan fácilmente a una descripción en términos de un sistema de ecuaciones.

Muchas veces es necesario hacer importantes suposiciones para-

simplificar y mantener las complejidades matemáticas y de cómputo - del modelo en un rango manejable y soluble a través de métodos conocidos.

Sin embargo, un modelo que ha sido simplificado hasta el punto de dejar de ser representativo del sistema que está siendo estudiado proporciona resultados obviamente inútiles. La experiencia en - este renglón ha demostrado que aun en casos triviales es difícil -- mantener la información en un nivel manejable y que permita a su -- vez describir las intrincadas relaciones incluidas en el proceso, - por lo que las perspectivas generales para el uso de este método no parecen muy viables.

La función de un computador se puede resumir así:

- La interacción de un conjunto de recursos que proporcionan servicio, con la característica de que un recurso solamente puede atender un requerimiento de un servicio a un tiempo.
- Un conjunto de requerimientos de servicio (carga de trabajo).
- Una cola de espera asociada a cada recurso para los requerimientos que no pueden ser atendidos inmediatamente.

Esto es, el funcionamiento de un computador se puede ver como- un sistema de colas o redes de colas donde los clientes son trabajos, transacciones, requerimientos de E/S, etc., y los puntos de -- servicio son los recursos del sistema: la CPU, un canal, una línea- de transmisión, un dispositivo de E/S o un subsistema de SOFTWARE - (tiempo compartido, teleproceso, etc.).

La mayoría de los modelos analíticos enfoca el análisis del -- funcionamiento de un computador utilizando los conceptos de la rama de las matemáticas llamada "teoría de colas".

La mayoría de estos modelos está orientada al análisis de la - administración de un recurso específico del sistema como CPU, manejo de memoria y organización de archivos, etc. Estos modelos resultan poco adecuados para llevar a cabo una evaluación profunda del - funcionamiento de todo el sistema, sin embargo son útiles en la medida en que permiten obtener una aproximación de primer orden del - rendimiento de una forma rápida y no costosa. Las técnicas de medición descritas anteriormente deben ser usadas para calibrar el modelo, proporcionando valores apropiados para los parámetros y distribuciones estadísticas que componen éste y validando los resultados- obtenidos con la situación real.

Para que un sistema o componente pueda ser evaluado mediante - un modelo de colas es necesario conocer la siguiente información básica.

- Tiempos de arribo de los requerimientos
- Tiempos requeridos de servicio
- Número de puntos de servicio
- Técnica de selección de los requerimientos en las colas de espera. En los sistemas de cómputo hay varias técnicas de selección entre las que se encuentran:
  - FCFS El primero que llega es el primero que se sirve
  - LCFS El último que llega es el primero que se sirve
  - SRFS El requerimiento de servicio más corto es el - primero que sirve

. RSS Selección aleatoria

. PR Selección por prioridad asignada

Hay varios tipos de modelos de colas que se utilizan para evaluar el rendimiento del sistema o de alguno de sus recursos. Kendall (KENDALL 01) estableció una notación que permite identificar el tipo de modelo y sus características básicas:

A/B/C/K/N/Z, donde:

A: Identifica el tipo de distribución utilizada para representar el tiempo de interarribo de los requerimientos. Esto es, el tiempo que transcurre entre dos requerimientos sucesivos.

B: Identifica el tipo de distribución utilizada para representar el tiempo requerido de servicio por requerimiento.

C: Número de puntos de servicio del modelo.

K: Número máximo de requerimientos que el modelo puede manejar en un tiempo dado (en cola + en servicio).

N: Número máximo de requerimientos que pueden ser alimentados al modelo en un tiempo dado.

Z: Disciplina de manejo para las colas de espera.

Por ejemplo: M/E3/2/10/100/FCFS identifica un modelo que tiene una distribución Poisson para representar el tiempo de interarribo, una distribución Erlang-3 para representar el tiempo de servicio; hay dos puntos de servicio; un límite de 10 requerimientos en el sistema (8 en cola y 2 ejecutándose); máximo 100 requerimientos por alimentar en un tiempo y una técnica de selección de "el primero que llega es el primero que se sirve".

El modelo más simple que se puede representar (fig. 4.9) consiste en un solo procesador (punto de servicio) y una sola cola de espera para las tareas por procesar.

Cada tarea se describe con dos parámetros: su tiempo de arribo al sistema y el tiempo requerido de servicio. La carga de trabajo de este sistema muy simple se especifica mediante la distribución de los tiempos entre arribos sucesivos (interarribo) y la distribución de los tiempos de requerimientos de servicio. Para evaluar el rendimiento de este sistema se evalúa el tiempo que una tarea pasa en el sistema (tiempo en colas más tiempo en servicio). Para poder representar y manejar este modelo de sistema tan simple con un modelo de colas se hacen las siguientes suposiciones, suposiciones que por otro lado, se hacen en la mayoría de los modelos de colas utilizados para evaluar un computador:

1.- Los tiempos de interarribo y de servicio son estadísticamente independientes.

2.- Los tiempos de interarribo sucesivos son estadísticamente independientes.

3.- Todos los tiempos de interarribo tienen la misma distribución.

4.- Los tiempos de servicio para requerimientos sucesivos son estadísticamente independientes.

5.- Todos los tiempos de servicio tienen la misma distribución.

En los sistemas reales de cómputo no se mantiene una independencia dentro de requerimientos de un mismo trabajo, sino que en la mayoría de los casos hay varios trabajos compitiendo simultáneamente -

REPRESENTACION GRAFICA DE UN SISTEMA DE COLAS CON UN SOLO PUNTO DE SERVICIO Y ALIMENTACION DE REQUERIMIENTOS FINITA

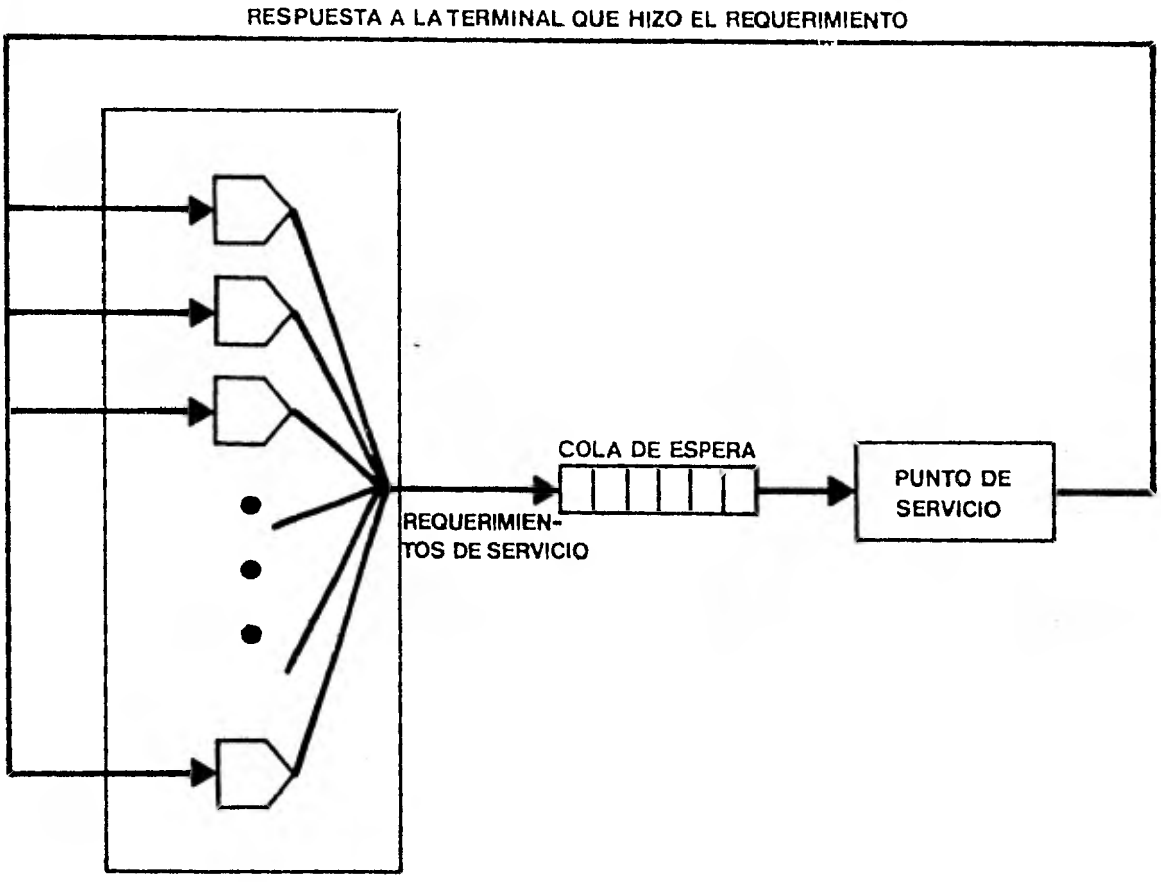


Figura 4:9

por los recursos del sistema, por lo tanto, es muy probable que requerimientos sucesivos pertenezcan a diferentes trabajos y sí sean independientes.

Otra suposición que se hace frecuentemente en los modelos de colas es el hecho de que se utilicen distribuciones exponenciales (Poisson, por ejemplo), para representar los tiempos de interarribo y servicio. Las distribuciones exponenciales son muy útiles porque cumplen con la propiedad de "Markov" o de "Memoryless". Esta propiedad significa que en cualquier tiempo T, la probabilidad de que se necesiten N unidades de tiempo adicional para completar una tarea que está en servicio es independiente del tiempo que la tarea ya ha recibido, o que la probabilidad de un nuevo arribo sea independiente del tiempo del último arribo. Estas distribuciones simplifican mucho el análisis de los modelos, y en la mayoría de los casos se considera que permiten una buena aproximación a la distribución real de los tiempos de interarribo.

Sin embargo, esta misma distribución es pobre cuando representa los tiempos requeridos de servicio, por lo que en muchos casos se utilizan o deben utilizarse distribuciones generales que se aproximen más a la realidad, aunque con esto se complique mucho más el análisis del modelo.

Modelación matemática y simulación

Un solo conjunto de fórmulas y sistemas de ecuaciones en un programa contienen la misma información y en muchos casos más, que la que pudiera obtenerse de un número enorme de corridas de simula-

ción. Por lo tanto es una gran ventaja tener una solución matemática para representar un sistema, suponiendo obviamente que la representación sea adecuada. Sin embargo, hay algunas características de los sistemas que difícilmente se pueden representar mediante fórmulas matemáticas.

En la mayoría de los casos los estudios de simulación y modelación matemática no son utilizados para obtener soluciones exactas a preguntas específicas de sistemas reales. Los modelos matemáticos son a menudo utilizados solamente para obtener un conocimiento más profundo del comportamiento y medio ambiente de prototipos o sistemas ideales. Los modelos de simulación van un poco más allá de los modelos matemáticos ya que pueden ser más precisos en la representación del medio ambiente y complejidad de un sistema real, sin embargo, no hay que olvidar que normalmente está implícito en este tipo de modelos un alto costo de desarrollo e implantación por lo que no son muy recomendables como herramientas en las primeras etapas del proceso EPC.

C A P I T U L O V

IMPLEMENTACION

- Situación típica de las instalaciones actuales
- Fases de implementación y actividades por fase
- Requerimientos de personal

SITUACION TIPICA DE LAS INSTALACIONES ACTUALES

Para implementar el proceso de EPC en las instalaciones de proceso de datos es necesario evaluar primero en qué situación se encuentra la instalación, ya que no todas están en el mismo nivel encuanto a las actividades típicas relacionadas con este proceso.

Las instalaciones actuales se pueden tipificar en 3 categorías:

a) Situación inicial: En esta categoría se encuentra la gran mayoría de las instalaciones en México; son aquellas instalaciones que no tienen bien definido un plan periódico de afinación del sistema; tienen poco conocimiento de las herramientas de medición. En general utilizan el paquete de contabilidad del sistema como únicafuente de información; dentro de su organización no tienen un grupo responsable de la función de EPC; se basan mucho en el proveedor para realizar y justificar sus requerimientos de equipo y para absorber no su plan de crecimiento, sino las cargas actuales que con elequipo actual se procesan a costa de muy mal servicio, etc.

b) Situación media: En esta categoría se encuentran las instalaciones que ya tienen una mayor experiencia en varias herramientas de medición por haberlas utilizado para afinar el sistema; conocen los problemas y las inconsistencias de algunas de ellas, y porlo tanto, han encontrado medios para subsanar estas deficiencias; -

conocen y entienden mejor el funcionamiento de su sistema; mantie-- nen una historia incipiente del funcionamiento del mismo y están -- buscando métodos para iniciar en forma sistemática un proceso perió dico de evaluación y planeación de la capacidad de cómputo de su -- equipo.

c) Situación avanzada: En esta categoría se encuentran muy po cas instalaciones en México y son aquéllas que tienen ya perfecta-- mente sistematizado su proceso de EPC y un grupo muy bien preparado que se encarga de él; los planes operativos y estratégicos de la -- institución concuerdan con el plan de adquisición de equipo; hay un banco histórico de datos muy completo sobre el funcionamiento y car ga de trabajo actual del equipo y del funcionamiento y cargas pre-- vistas, por lo cual es posible comparar perfectamente lo previsto - con el crecimiento y funcionamiento real para corregir nuevas esti- maciones y mejorar los procedimientos de proyección. Tienen la in- formación necesaria para utilizar técnicas sofisticadas de proyec-- ción y pronóstico, como la modelación y la simulación.

Es importante la diferencia en la situación de las instalacio- nes, ya que, como veremos más adelante, se sugiere implantar por fa ses el proceso de evaluación, planeación y control de la capacidad- porque se considera ilógico utilizar, por ejemplo, técnicas sofisti cadas de modelación o simulación cuando los datos y parámetros que- se usan en el modelo no son muy confiables. Para obtener datos con fiables se requiere de un periodo de familiarización con las herra- mientas que permita conocer y neutralizar sus posibles deficiencias y obtener una muestra realmente representativa de la situación de - una instalación.

FASES DE IMPLEMENTACION Y ACTIVIDADES POR FASE

En el punto anterior vimos la necesidad de implementar el proceso de EPC tomando como base la situación en que se encuentra la instalación. En este punto sugerimos una estrategia de instalación del proceso tomando en cuenta la tipificación de las instalaciones expuesta en el punto anterior.

Pensando en lo complejo que pudiera ser el implementar desde un principio este concepto en una forma total, en el caso de las instalaciones que se encuentran en la situación inicial se ha considerado necesario recomendar la implementación por fases o módulos que permitan obtener resultados parciales, lo cual daría como resultado:

a) Que la organización se acostumbre poco a poco al proceso. Esto es muy importante porque, como es natural, en un principio existe cierta retiscencia al cambio y dudas acerca de la utilidad de su función.

b) Al implantar por fases, el grupo responsable va adquiriendo poco a poco experiencia en la instalación, procesos y herramientas, lo cual permite una mayor seguridad en las estimaciones y proyecciones efectuadas.

c) Al obtener resultados en lapsos más cortos, se demuestra

más fácilmente a los directivos la utilidad de la función.

Ahora bien, para alcanzar el objetivo de la implementación -- por fases es necesario crear, desde el primer momento, un bando de datos (fig. 5.1).

Con los datos del rendimiento y funcionamiento del sistema, - la información de las proyecciones efectuadas y todos los datos necesarios para en su momento comparar y analizar los resultados obtenidos y las posibles desviaciones, tendremos la información necesaria para que la instalación vaya adquiriendo experiencia y confianza en el proceso.

Se recomienda instalar el proceso en 4 fases:

I. Afinación del sistema.

Para que cualquier actividad de evaluación de la capacidad -- proyección y predicción de funcionamiento tenga sentido, primera-- mente es necesario tener un sistema razonablemente bien afinado.

Existe mucha literatura sobre cómo afinar un sistema y cada - proveedor proporciona guías específicas de acuerdo con el tipo de- HARDWARE, sistema operativo y subsistema de SOFTWARE (teleproceso, en lote, proceso compartido, etc.); por lo tanto, este trabajo no - tiene como objetivo presentar una metodología de afinación, simplemente se pretende presentar los alcances da esta primera fase y -- las consideraciones más importantes.

Los resultados que se esperan de esta primera fase son:

1) Mantener razonablemente bien afinado el sistema.
2) Ubicar al grupo de EPC en la organización y acostumbrar a ésta a la nueva función.

EL PROCESO DE EVALUACION Y PROYECCION

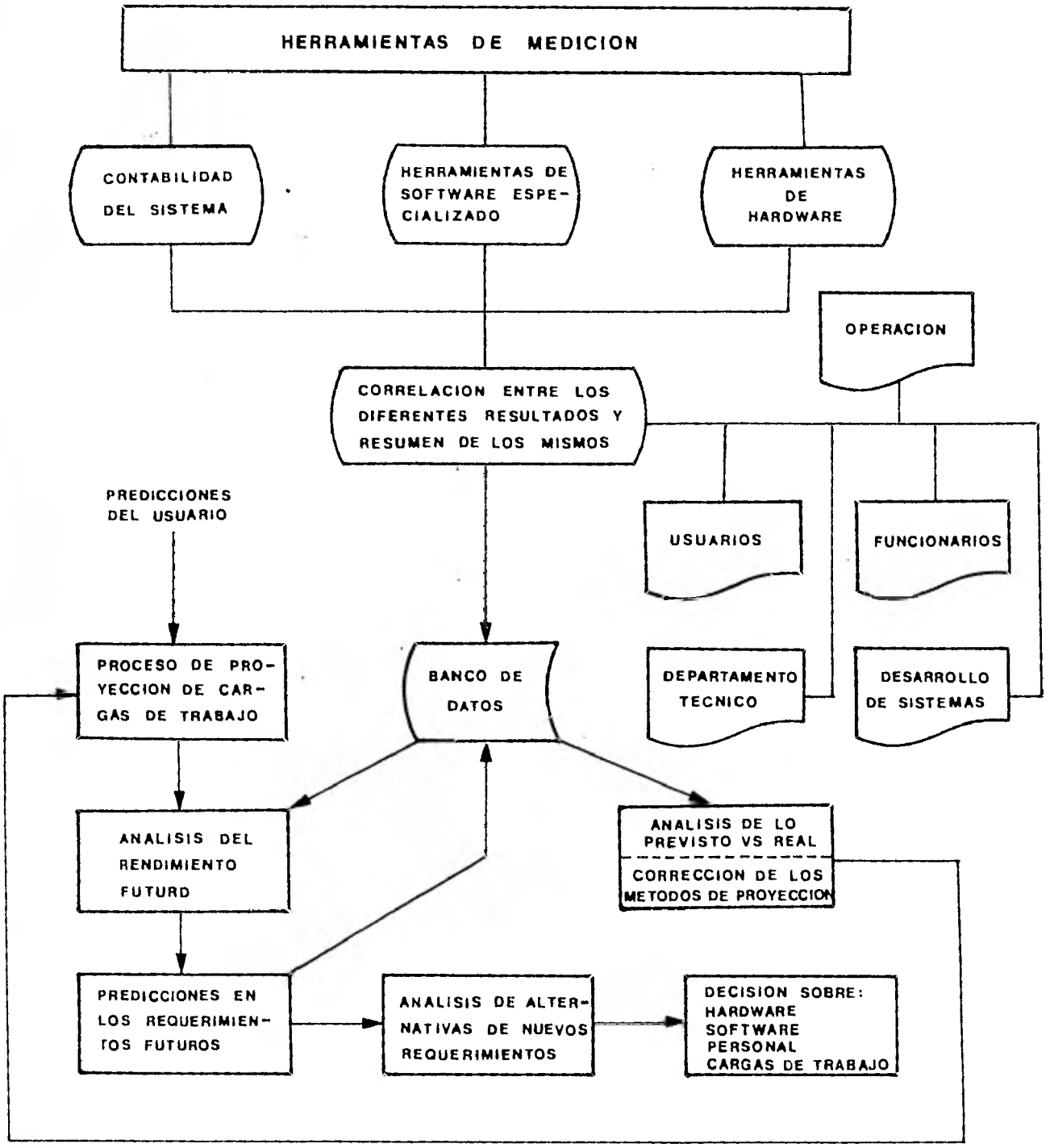


FIG. 5.1

3) Conocer las herramientas de medición.

4) Obtener sugerencias para mejorar los procedimientos que afectan la operación del sistema.

Para lograr estos resultados proponemos lo siguiente:

1) Formar el grupo de EPC (ver requerimientos de personal en este mismo capítulo)

2) Definir las herramientas de medición que se utilizarán.

3) Definir las funciones de la institución que requieren información periódica y el formato de los reportes.

4) Instalar las herramientas de medición y reporte.

5) Identificar las aplicaciones que son representativas de la producción. Estas aplicaciones son las que serán evaluadas, ya que sería impráctico, y en muchos casos imposible, estudiar y evaluar todas las aplicaciones. Además, en la mayoría de los casos un número muy reducido de aplicaciones representan un porcentaje muy considerable de la carga del computador.

6) Definir los componentes críticos del computador. En este caso tampoco vale la pena evaluar y estudiar componentes que no son críticos, ya sea porque claramente se nota que no representan un problema en el rendimiento del sistema o porque son muy fáciles de adquirir en el mercado debido a su precio y a la facilidad en cuanto a tiempo de entrega y proveedores. En el proceso de EPC hay que tener siempre presente que es necesario mantener la información al mínimo necesario para que siga siendo útil.

7) Afinación del sistema.

Como hemos dicho, esta actividad debe ser periódica y básicamente tiene como objetivo optimizar la operación de un computador tratando de que todos los elementos que intervienen (al menos los que se consideran críticos) funcionen eficientemente de acuerdo con ciertos objetivos planteados de antemano. Hay que recordar que esto no significa que por ejemplo el canal deba de tener un porcentaje de utilización del 100% para que sea eficiente; se debe tratar de balancear la utilización de los canales para evitar que se conviertan en un cuello de botella en la operación de todo el sistema.

8) Optimizar los procedimientos administrativos del centro de cómputo.

Esta actividad es paralela a la anterior, y por lo tanto deberá ser periódica.

Se ha aclarado ya que muchos problemas en el funcionamiento del sistema se resuelven al observar periódicamente y sugerir modificaciones muy simples en los procedimientos administrativos del centro de cómputo.

9) Detectar posibles inconsistencias entre las diferentes herramientas.

Ya se mencionó que algunas herramientas tienen diferente enfoque de medición y que por lo tanto dos herramientas pueden dar resultados diferentes para un mismo elemento y una misma muestra. Esta fase persigue detectar esas incon

gruencias y establecer un procedimiento que permita subsanar esas diferencias, de tal manera que todos los datos - se correlacionen y por lo tanto, se complemente la información.

II. Conocimiento de las cargas de trabajo y requerimientos de servicio.

Una vez afinado el sistema se está en posibilidad de analizar el impacto de las cargas de trabajo en los recursos del sistema y de fijar con el usuario los requerimientos de servicio.

El objetivo de esta fase es entonces conocer el comportamiento del equipo, analizando el impacto de las cargas de trabajo, para sentar las bases que permitan determinar los requerimientos de equipo y de recursos humanos para los próximos 2 ó 3 años y mantener más objetivamente un nivel de servicio que tenga satisfechos a los usuarios.

Los resultados que se esperan en esta fase son:

1) Información sobre el empleo de los recursos por aplicación
   - Humanos
   - SOFTWARE
   - HARDWARE
2) Sugerencias para optimizar la programación de trabajos deacuerdo con la utilización de los recursos
3) Niveles de servicio formalizados con los usuarios
4) Conocimiento de las características de las cargas de trabajo

5) Plan para mejorar la disponibilidad del sistema

6) Banco de datos con el comportamiento previsto y real del sistema

7) Requerimientos de recursos inmediatos y para los siguientes 2 ó 3 años

Las actividades sugeridas para lograr esto son:

1) Tipificar los turnos de trabajo.

En este caso no se está hablando de los turnos de operación, sino de los periodos del día en que se tipifican las cargas de trabajo del computador (ver fig. 5.2)

2) Definir las características de las cargas de trabajo de las aplicaciones críticas.

- Orientación (CPU o E/S)
- Número de volúmenes de discos y cintas que utilizan.
- Prioridades
- Horarios
- Volumen de impresión, etc.

3) Diseñar la base de datos donde se localizarán los datos de funcionamiento y proyección.

4) Recolectar datos de utilización de los recursos del equipo.

Esta es una actividad periódica en la que hay que obtener sumarizaciones de utilización de los recursos por:

- Departamento
- Usuario

PERIODOS TIPICOS DE CARGAS DE TRABAJO

EN UNA INSTALACION

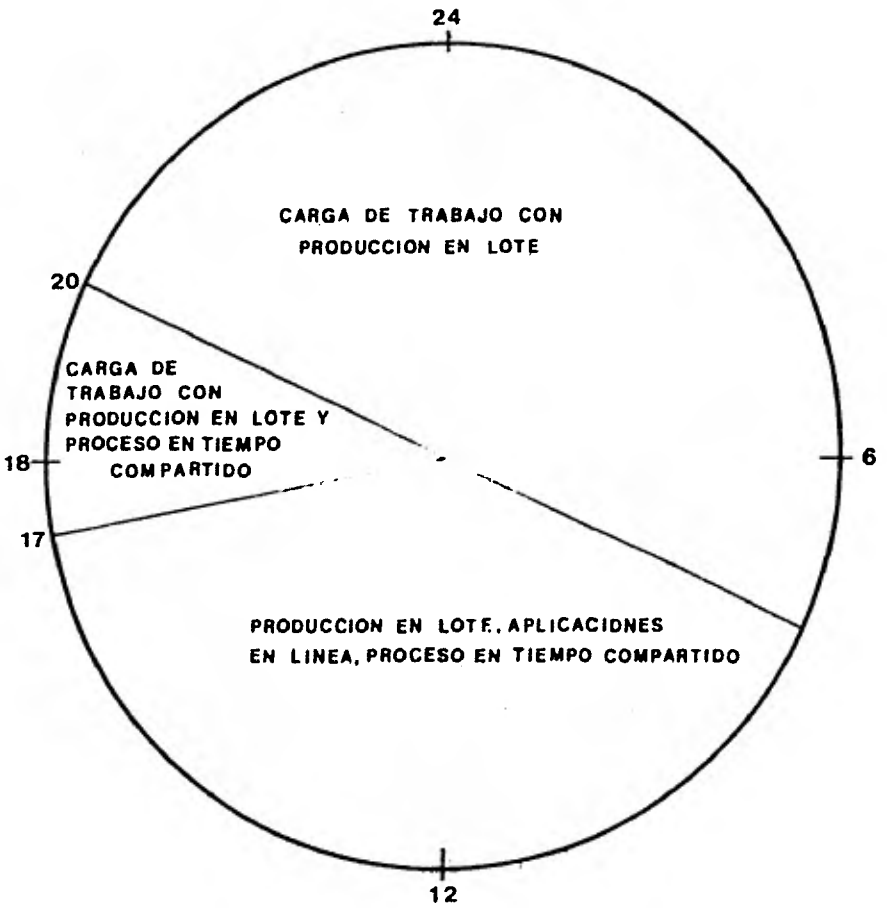


FIG. 5.2

. Aplicación

. Subsistema de SOFTWARE

. Sistema operativo

. Sistemas en línea

. Tiempo compartido

. Paquetes matemáticos, etc.

Estas sumarizaciones en días, semanas y meses deberán representar periodos pico y periodos de cargas normales.

5) Optimizar los procedimientos de programación de trabajos.
A partir de la información que se tiene de las características de las aplicaciones, y ya encontrada la relación que hay entre carga de trabajo, utilización de los recursos y tiempo de respuesta, es posible formular un algoritmo más adecuado que permite programar los trabajos para balancear en lo posible las cargas de trabajo en el computador.

6) Formalizar requerimientos de servicio con el usuario.
Definir con el usuario cual sería el tiempo de respuesta promedio y máximo requerido por aplicación.
En este caso podrá haber usuarios descontentos con el servicio actual para los cuales se deberá iniciar un estudio que permita definir métodos para optimizar el tiempo de respuesta, por ejemplo:

. Optimización de los procedimientos administrativos.

. Optimización de los programas de aplicaciones

. Aumento en la prioridad de los programas

. Reubicación de los archivos de la aplicación

. Adecuación del sistema operativo para esta - aplicación, etc.

Es necesario balancear la importancia de esta aplicación - con la de las otras, y ubicar mayores recursos del sistema, en caso necesario, para mejorar el tiempo de respuesta a - costa obviamente de las otras aplicaciones.

A partir de este tipo de estudios se pueden ya determinar los requerimientos inmediatos de equipo y de recursos para mejorar la operación del sistema.

7) Mantener permanentemente informados a los representantes - de las funciones definidas con reportes sumarizados y ac-- ciones que se han llevado a cabo.

8) Obtener promedios de disponibilidad del sistema por aplica ción crítica.

En este caso hay que recordar que la disponibilidad es la- que el usuario percibe, por lo tanto hay que tomar en cuen ta tiempos perdidos en

. reprocesos
. recuperación de archivos
. retransmisiones por fallas en líneas
. reinicio del servicio, etc.

9) Planificar para mejorar la disponibilidad del sistema por aplicación.

Después de estudiar la disponibilidad por aplicación se - podrían sugerir mejoras como:

. exigir mejor servicio de mantenimiento al proveedor

. rediseñar procedimientos de respaldo y recuperación de archivos

. actualizar los niveles del SOFTWARE

. mejorar la programación, etc.

10) Definir algoritmos de conversión de unidades de trabajo del usuario (UTU) a unidades de trabajo del sistema (UTS) por - aplicación.

En esta fase encontraremos métodos para convertir las unidades de trabajo que maneja el usuario, por ejemplo:

. facturas / día

. pedidos / día

. clientes / día, etc.

a unidades de trabajo del sistema como:

. registros que se leerán / día

. tarjetas que se leerán / día

. registros que se almacenarán por archivo

. transacciones por terminal / día

. accesos a la base de datos

. número de programas, etc.

Esta actividad es muy importante, y puede ser muy difícil - si no se conocen adecuadamente las aplicaciones correspondientes.

11) Estimar las cargas de trabajo futuras.

El primer paso para hacerlo es conocer perfectamente los - planes de crecimiento de la institución y traducir estos - planes a un estimado en las cargas de trabajo del computa-

dor. El crecimiento en las unidades de trabajo del usuario puede repercutir en:

a) crecimiento en los volúmenes para las aplicaciones actuales
b) incremento en la complejidad de las aplicaciones actuales
c) implantación de nuevas aplicaciones.

Hay además otra razón muy importante por la que podrían incrementarse las cargas de trabajo del computador. La demanda constituida por todos los requerimientos de servicio actual que no se alimentan al computador por alguna limitación en los recursos actuales y que al momento de incrementarse la capacidad del centro de cómputo empiezan a ser servidos. A este tipo de requerimientos se le llama demanda latente.

Un método para estimar la demanda latente es comparar la carga de trabajo actual con estimaciones efectuadas en otras instalaciones para el mismo tipo de aplicación. Por ejemplo, para el desarrollo interactivo se sabe que el número de programadores promedio por terminal para instituciones comerciales en las que no hay problemas de capacidad es de 2.5 a 3 programadores por terminal. Entonces, una forma muy simple y práctica es comparar el número de programadores de la instalación con el número de terminales. Si la proporción es mayor que el promedio en otras instalaciones, hay muchas posibilidades de una demanda la

tente en esa aplicación, por lo que es conveniente iniciar una investigacióm.

12) Proyectar la utilización de los recursos de acuerdo con la carga estimada.

Hay una gran variedad de herramientas para llevar a cabo - la proyección. (Ver capítulo herramientas de medición y -- predicción). Los modelos pueden ser de diferentes tipos y con diferentes niveles de detalle.

Pueden ser muy simples, por ejemplo aquéllos en que se proyecta linealmente la carga de trabajo y la utilización de- los recursos correspondientes y que se basan en las medi-- ciones periódicas efectuadas; o mucho más detallados y costosos, como aquéllos en los que se mide el funcionamiento- y rendimiento de varios modelos de computadores con una réplica exacta de la carga de trabajo y aplicaciones esperadas.

En la figura 5.3 se describe toda la gama de herramientas que se utilizan para este concepto.

El costo, la complejidad y por tanto el tiempo de imple-- mentación va creciendo conforme se utilizan modelos de un nivel de detalle mayor.

Por lo tanto en esta fase se aconseja utilizar técnicas sim--- ples de proyección (ver fig. 5.4) para dar oportunidad a que la instalación adquiera confianza en los datos recolectados y en los beneficios de la función de proyección.

Connotados autores dudan de la utilidad de técnicas tan avanza

GAMA DE TECNICAS PARA ANALISIS DEL RENDIMIENTO

REGLAS PRACTICAS EMPIRICAS	PROYECCIONES LINEALES	MODELOS DE COLAS	SIMULACION DISCRETA	PRUEBAS COMPARATIVAS (BENCH-MARK) SINTETICAS	SIMULACION DE REDES	PRUEBAS COMPARATIVAS CON CARGA REAL

FIG. 5.3

PROCESO DE PROYECCION LINEAL

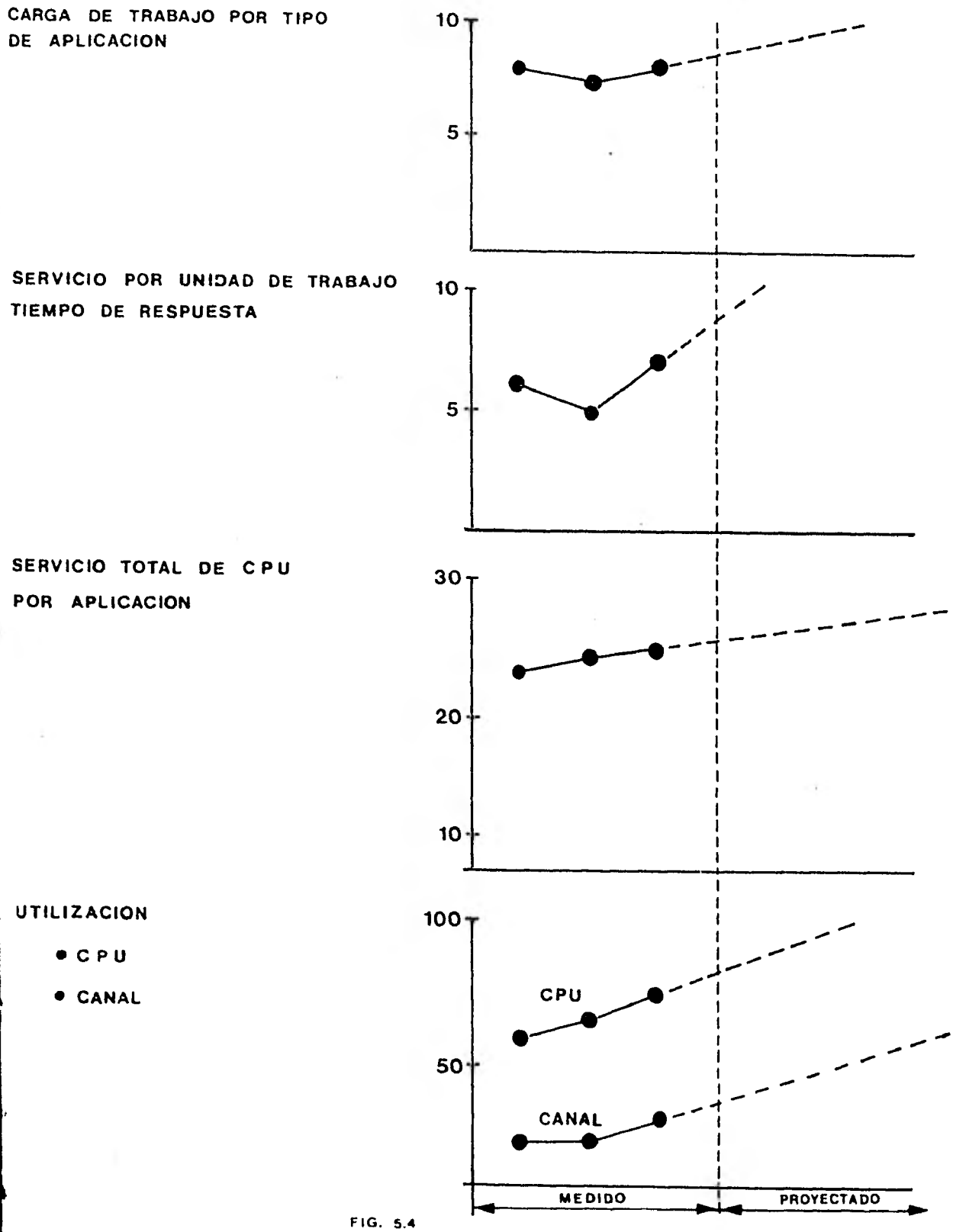


FIG. 5.4

das como la simulación para estimar la utilización de los recursos de un computador; por ejemplo, Grenader y Tsao (GRENANDER 01) establecen:

"Nosotros dudamos de... que las actividades de medición y simulación de un sistema puedan mejorar significativamente el entendimiento de un sistema de cómputo".

Nielsen (NIELSEN 01) menciona que un proveedor de sistemas de cómputo estableció "que el desarrollo de un modelo adecuado de simulación de un sistema de tiempo compartido costó tanto como el desarrollo del mismo sistema".

Bell (BELL 01) establece la necesidad "de definir claramente los objetivos antes de realizar el esfuerzo de desarrollar un modelo de simulación, ya que el esfuerzo es a menudo muy grande y además, el analista debe considerar muy cuidadosemente la validez de los resultados".

III. Utilización de herramientas sofisticadas de proyección de cargas de trabajo y predicción del rendimiento.

Ya que se conoce bien el impacto que las cargas de trabajo producen en el funcionamiento del sistema, que éste está razonablemente afinado y que se han optimizado los procedimientos administrativos relacionados con la operación del mismo, se sugiere utilizar técnicas más complejas, que permitan confirmar o rectificar las observaciones previas.

Los resultados que se espera obtener de esta fase son:

1) Estimaciones del funcionamiento futuro que nos permitan --

prever cuellos de botella.

2) Confirmar o rectificar los requerimientos de equipo a futuro.

3) Anticipar el impacto en el servicio de cambios en:

   . HARDWARE
   . Sistema operativo y paquetería
   . Aplicaciones
   . Reconfiguración, etc.

4) Sugerencias para optimizar el plan operativo del centro de información.

5) Información más confiable para la correcta evaluación de propuestas de equipo y SOFTWARE de varios proveedores.

Para conseguir estos resultados es necesario:

1) Evaluar los diferentes paquetes de modelación y simulación de varias casas proveedoras:

   En el mercado hay muchos paquetes que permiten proyectar las cargas de trabajo para predecir o simular el funcionamiento del equipo actual o de algún equipo propuesto.

   Algunos de los parámetros que se recomienda evaluar para decidir qué tipo de paquete es más conveniente son:

   . Requerimientos de ejecución
   . Costo (de compra y de utilización)
   . Flexibilidad para cambio de parámetros
   . Complejidad
   . Técnica que utiliza

. Requerimientos de información.

2) Diseño y programación de un paquete propio

Esta opción no es muy conveniente, y solamente la mencionamos pensando en casos en que no se encuentre nada adecuado en el mercado.

Hay varios lenguajes que permiten desarrollar programas -- para simular la actividad de un computador, por ejemplo -- Algol, PL/1, Fortran, Pascal; también pueden utilizarse -- lenguajes especiales de simulación de eventos discretos como GPSS, Aspol, Simula, etc. Sin embargo consideramos que para desarrollar nuestros modelos y poder simular la actividad de un computador y sus principales componentes se requiere de un esfuerzo y unos conocimientos que rara vez se encuentran en las instalaciones estándar de proceso de datos en México, por lo cual no consideramos conveniente profundizar a este respecto en este trabajo.

3) Implementación del paquete

Consta de tres fases:

. Estudio

. Calibración

. Ejecución y evaluación de resultados.

En la fase de estudio se define y reúne la información necesaria. Posteriormente se calibra, esto es, se ejecuta - el paquete con datos actuales y se comparan los resultados obtenidos con los resultados de las mediciones de la situación real hasta que se comprueba que los parámetros que --

define el modelo corresponden a los parámetros reales del-sistema, de tal manera que se obtengan los mismos resultados. Una vez calibrado, se ejecuta variando las condiciones, por ejemplo:

- Porcentaje de crecimiento de las cargas de trabajo
- Modelo de CPU
- Número de canales, etc.

4) Evaluar alternativas de crecimiento para absorber las cargas futuras.

5) Almacenar los datos de funcionamiento y crecimiento de las cargas de trabajo previstos en el banco de datos.
Esto tiene por objeto comparar en el tiempo e ir corrigiendo los métodos de estimación.

6) Evaluar el plan operativo del centro de cómputo.
Es necesario evaluar la factibilidad de los compromisos -- efectuados con los usuarios de acuerdo con el impacto que-pudieran tener en el servicio de las aplicaciones actuales y llamar la atención de los gerentes para los casos que se consideren riesgosos.

7) Sugerir la implementación de equipo, niveles de SOFTWARE y paquetería que pudieran mejorar la utilización de los re-- cursos y el servicio al usuario.

REQUERIMIENTOS DE PERSONAL

Antes de implementar el proceso de EPC es necesario organizar el grupo que se encargará del mismo. Este grupo, para poder realizar adecuadamente su función, tendrá que comprender las necesidades y responsabilidad del:

- Usuario
- Grupo de sistemas y programación
- Operación del sistema
- Programación de trabajos
- Gerentes o funcionarios de proceso de datos
- Otro tipo de funcionarios
- Personal del grupo de soporte técnico

Por otro lado, el grupo tendrá como funciones básicas:

- Análisis del funcionamiento del sistema
- Detección de problemas operativos
- Modelación y predicción de requerimientos

Por lo tanto, es recomendable formarlo con personal con experiencia en 4 áreas básicas:

- Operación
- Soporte técnico
- Programación de aplicaciones

. Matemáticas, estadística, simulación, teoría de colas, modelación, etc.

Obviamente no hay una definición exacta del número adecuado -- de personas para el grupo; esto dependerá de las dimensiones de la- institución y de la etapa de desarrollo en que ésta se encuentre. - En muchos casos, y debido a la experiencia que se requiere, se reco mienda empezar con dos personas, una de ellas con experiencia en -- análisis y la otra con experiencia técnica en sistemas y matemáti-- cas. En algunas instalaciones (no en México) este grupo ha llegado a estar formado hasta por 7 u 8 personas.

Una vez seleccionado el grupo de trabajo es muy importante dar le la responsabilidad adecuada para que su función pueda realmente- influir en las áreas citadas, lo cual mucho dependerá del lugar que esta función ocupe en la organización.

En la figura 5.5 se sugiere una ubicación que permitirá que -- las sugerencias de optimización que este grupo haga a las diferen-- tes áreas de proceso de datos realmente se lleven a cabo, pues en - muchos casos en los que esta función le reporta a alguna de las --- áreas de segundo nivel descritas en la fig. 5.5 se ha notado que -- sus sugerencias pierden efectividad.

Muchas veces a este grupo se le asigna además la función de -- auditoría, lo cual lleva implícito el riesgo de que se le pongan -- más barreras en el desempeño de sus funciones,por lo que definitiva mente se recomienda separar las dos funciones.

ORGANIZACION DE PROCESO DE DATOS

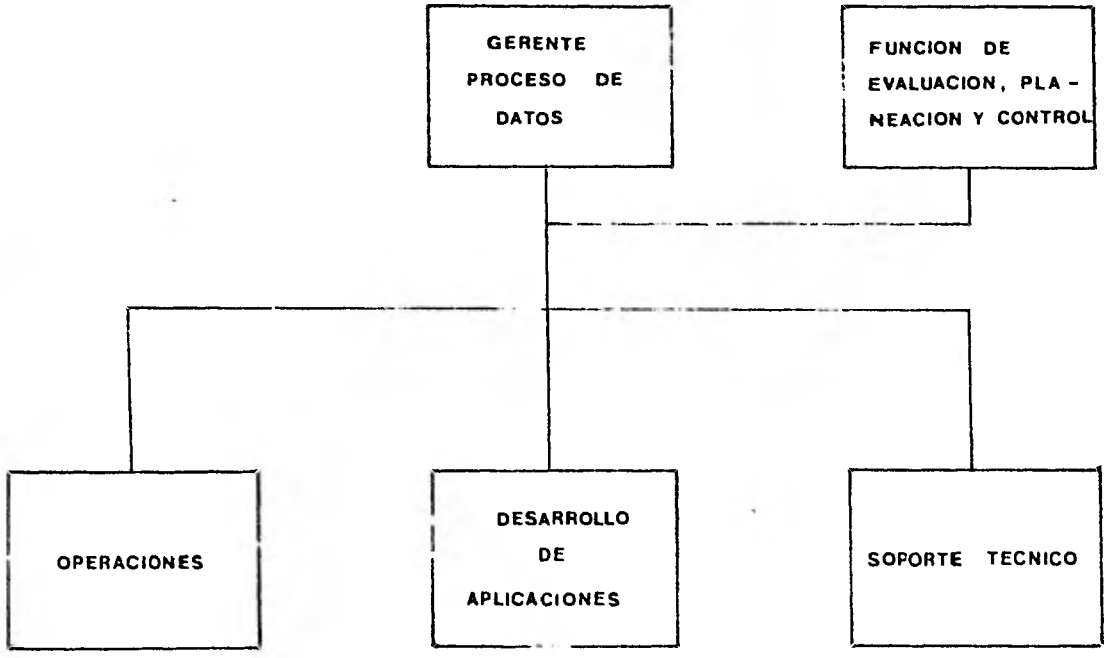


FIG. 5 5

C A P I T U L O VI

EJEMPLO

- Antecedentes
- Metodología
- Terminología
- Fases del estudio
- Actividades realizadas y resultados obtenidos

A N T E C E D E N T E S

Este estudio fue efectuado en diciembre de 1980. Es el fruto de un año y medio de estudio, análisis y recolección de información sobre el medio ambiente de un computador IBM modelo 370/158. Los objetivos principales fueron determinar la capacidad disponible de la CPU y estimar la fecha en la cual la unidad central de proceso sería insuficiente para cumplir adecuadamente con los requerimientos de servicio.

Como resultado de este estudio se definió la necesidad de incrementar el poder de cómputo para satisfacer las demandas previstas sobre todo en el área de sistemas en línea (TSO y APL). Se analizaron varias alternativas (siempre dentro del mismo proveedor ya que ésta fue una condición impuesta por los funcionarios de la instalación en la que se efectuó el estudio), y se hicieron algunas sugerencias en cuanto a la programación de trabajos y la administración de la operación.

En este estudio participó un grupo de personas empleadas en la instalación en donde se desarrolló el proyecto, a quienes agradecemos su colaboración; pero muy especialmente queremos agradecer la participación de la señora RENATA BOKER, principal responsable, ---

quien sentó las bases y dió las facilidades necesarias para llevarlo a cabo.

El estudio se basa en la información obtenida mediante las siguientes herramientas:

a) SMF (SERVICE MANAGEMENT FACILITY): es el paquete de contabilidad para computadores IBM (véase Herramientas de medición) que obtiene estadísticas de la utilización de los recursos del sistema, desglosando la información por programa, lo cual permite obtener resúmenes por usuario, aplicación o subsistema.

b) RMF (RESOURCE MANAGEMENT FACILITY): es un paquete que funciona como el SMF (véase Herramientas de medición) cuyo objetivo es complementar la información de éste para los sistemas operativos MVS (MULTIPLE VIRTUAL STORAGE).

e) SLR (SERVICE LEVEL REPORTER): es un paquete que permite obtener una serie de reportes y gráficas de la información recolectada por SMF y RMF. Este paquete permite seleccionar, clasificar, sumarizar y graficar la información almacenada por los dos paquetes anteriores.

En la instalación donde se efectuó el estudio (*) hay dos computadores modelo 370/158. Uno de ellos (objeto de este estudio) estaba orientado a los problemas de producción y apoyo adeministrativo de la Institución. El otro estaba orientado a soporte de investigación y mecadeo, y era administrado por un departamento diferente, por eso, como sucede en muchos de estos casos, los planes de cre

(*) La Institución que amablemente nos permitió publicar este estudio, nos pidió no mencionar su nombre, por lo que en adelante la llamaremos "la instalación".

cimiento de los dos computadores eran en cierta forma independientes. Decimos "en cierta forma" porque las decisiones que tomaban en un caso influían necesariamente en el otro. Sin embargo, hasta el momento del estudio no se pensaba en una solución integral para las necesidades de crecimiento.

El sistema operativo del computador estudiado era MVS (MULTIPLE VIRTUAL STORAGE) con dos tipos de subsistema; la principal fuente de utilización de los recursos del computador era la gran producción de procesos ejecutados en lote (BATCH) durante las 24 horas del día, seis días a la semana, La mayor parte de estos trabajos apoyaba a los sistemas administrativos de la institución y por lo tanto estaba programado en lenguajes orientados a procesos administrativos (COBOL y PL/1). En la figura 6.1 se observa la distribución del número de trabajos procesados al día en el mes de octubre de 1980, lo cual da una idea de las dimensiones de la instalación.

Durante el día y parte de la noche se daba servicio de proceso compartido mediante el paquete llamado TSO (TIME SHARING OPTION), básicamente orientado al desarrollo y mantenimiento de las aplicaciones del usuario y al mantenimiento del sistema operativo. Este tipo de proceso representaba en octubre de 1980 aproximadamente el 40% de la carga de trabajo total de la unidad central de proceso (CPU).

Un punto muy importante y que definitivamente influyó para la realización de este estudio, fue que la instalación ya tenía tiempo utilizando la herramienta SMF como fuente de información para la distribución por departamento usuario del costo operativo, y para

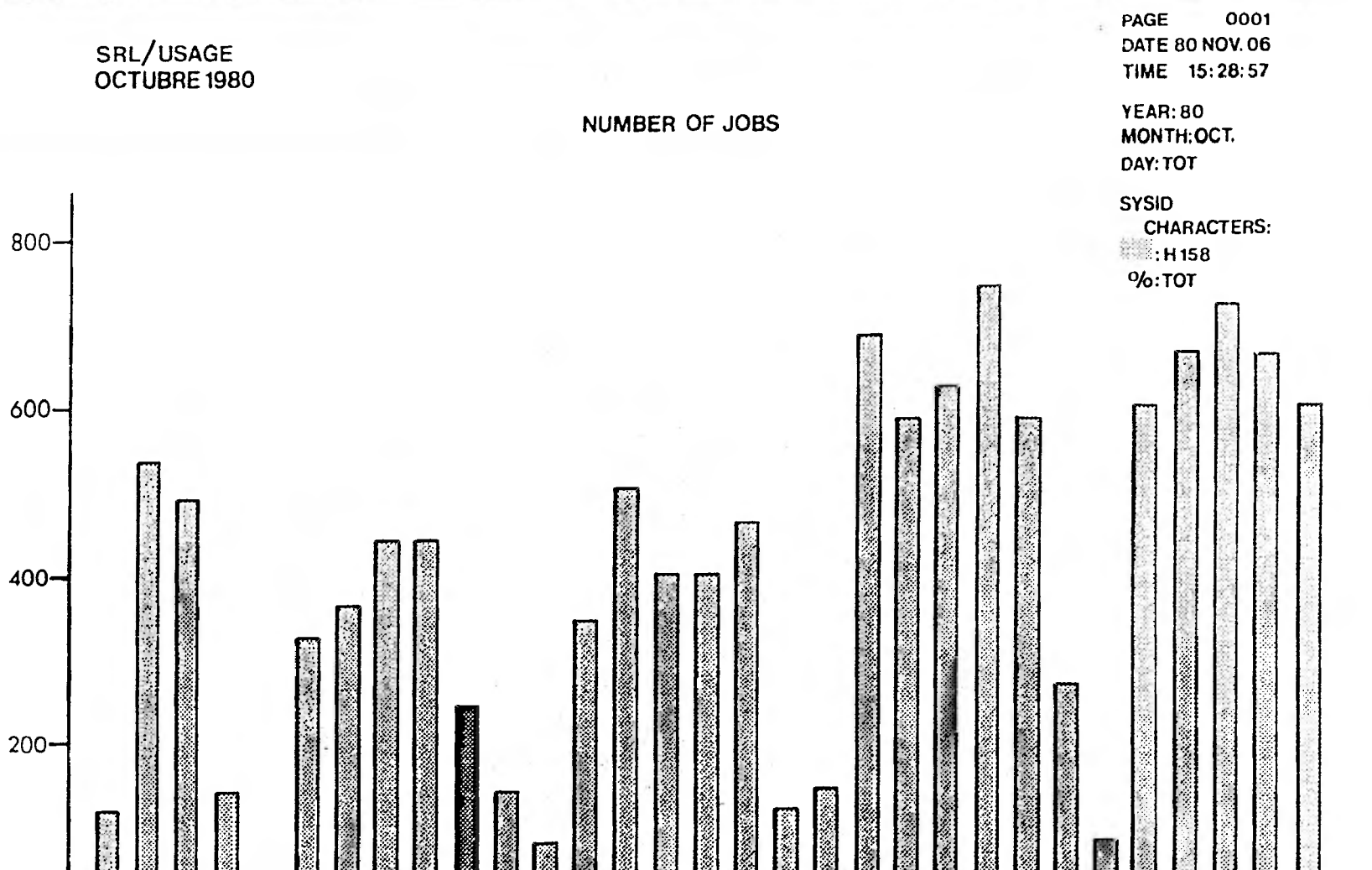

SRL/USAGE
OCTUBRE 1980
NUMBER OF JOBS
PAGE 0001
DATE 80 NOV. 06
TIME 15:28:57
YEAR: 80
MONTH: OCT.
DAY: TOT
SYSID
CHARACTERS:
: H158
%: TOT
800
600
400
200
0
1
3
5
7
9
11
13
15
17
19
21
23
25
27
29
31
DAY

Figura 6.1

las actividades periódicas de afinación. Por lo tanto, para este estudio hubo la oportunidad de analizar la información detallada del funcionamiento del computador desde enero de 1979 hasta noviembre de 1980. Aunque este trabajo incluye solamente cuadros y gráficas correspondientes a junio de 1980, en realidad se analizaron detalladamente los meses de marzo, julio y octubre de 1979, enero, mayo, junio, julio, agosto, septiembre y octubre de 1980.

METODOLOGIA

En este trabajo se utilizó una metodología denominada USAGE -- (UNDERSTAND YOUR SYSTEM AND APPLICATION GROWTH ENVIRONMENT) (COOPER 01) que básicamente se utiliza para calcularla capacidad real y analizar el comportamiento de las cargas de trabajo durante varios meses y proyectar con métodos simples de regresión lineal los requerimientos futuros.

Esta metodología considera que la CPU (Unidad Central de Proceso) es el recurso más crítico del computador, y aunque acepta que no es el único que requiere planeación, enfoca su estudio y evaluación al impacto que las cargas de trabajo tienen en el mismo.

Para ello, la carga de trabajo actual es dividida y sumarizada en las aplicaciones o subsistemas más importantes. Se evalúa el impacto de estas aplicaciones en la CPU y se estima el crecimiento futuro previa consulta a los departementos usuarios correspondientes, basados en guías establecidas a través de la experiencia y en el estudio del crecimiento histórico de la instalación.

Esta metodología utiliza un modelo muy simple para predecir la utilización de CPU en el futuro. El porcentaje de crecimiento identificado en la fase de predicción del estudio se aplica a las horas de CPU medidas para las aplicaciones consideradas más importantes,-

y el resultado es la cantidad de CPU que se requerirá en el futuro. Aunque éste es un método muy simple, ha resultado ser útil y lo suficientemente preciso para iniciar el proceso de EPC en las instalaciones (COOPER 01). Hay otro tipo de técnicas más apropiadas para instalaciones con más experiencia en el proceso de EPC y en el uso de técnicas como la modelación matemática y la simulación, o cuando se quiere estudiar un sistema orientado a procesos en línea, en los que el tiempo de respuesta es un factor crítico.

Finalmente es importante comentar que para que la metodología seguida pueda cumplir con los objetivos es necesario que el sistema se mantenga razonablemente bien afinado durante el proceso.

TERMINOLOGIA

TIEMPOS DE SMF: Son los tiempos de CPU grabados por medio de SMF y que son la fuente básica de información para este estudio.

PERIODOS DE PRODUCCION: Ya hemos dicho que este estudio está basado en el análisis de la información obtenida sobre varios turnos de producción. El procedimiento podría haber sido sacar promedios mensuales de utilización de CPU por mes y proyectar los requerimientos futuros con base en estos promedios mensuales.

Sin embargo, se ha demostrado que este método no es adecuado para determinar los requerimientos de capacidad porque la carga de trabajo, y por lo tanto la utilización de la CPU varía significativamente durante el mes dependiendo de:

a) La hora: normalmente la CPU se utiliza más entre las 11 y las 14 h, que es el horario durante el cual se da servicio en línea y en el que la mayoría de los usuarios utilizan el computador.

b) El día: en la mayoría de las instalaciones no se puede comparar la carga de trabajo durante los días hábiles (lunes a viernes) con la del sábado o domingo, por ejemplo.
Por lo tanto, para no distorsionar las cifras, es muy útil analizar la información dividiendo el día en periodos de --

producción, los cuales no necesariamente tienen que coincidir con los turnos de operación establecidos en la Instalación, sino que se escogerán según:

1) Las características de la carga de trabajo: Por ejemplo, diferenciar entre periodos con carga de trabajo orientada a sistemas en línea con una orientada a procesos en lote.
2) El volumen de la carga de trabajo: Por ejemplo, diferenciar el periodo con una gran carga de trabajos en lote del periodo con ligera carga de estos trabajos.

En nuestro estudio dividimos el día en los siguientes periodos de producción:

PERIODO 1: DE 7 a 18 h
PERIODO 2: DE 18 a 0 h
PERIODO 3: DE 0 a 7 h
FINES DE SEMANA

En el periodo 1 se efectuaba del 60 al 75% de todo el trabajo en línea; en el periodo 2 se efectuaba del 15 al 20% y la mayor parte del trabajo en lote (más o menos el 40%); durante el tercer periodo, se efectuaba del 2 al 5% del trabajo en línea y del 20 al 30% del trabajo en lote, aproximadamente. Los fines de semana había muy poco trabajo, por lo que también se decidió estudiarlos como un periodo aparte.

TIEMPO DE RELOJ: Este tiempo es simplemente el lapso transcurrido entre un evento y otro. Es necesario tenerlo en cuenta porque, como veremos más adelante, permite calcular y entender la disponibilidad del sistema y la utilización de CPU para un periodo es-

pecífico.

TIEMPO TRANSCURRIDO MEDIDO (ELAPSED TIME): Como su nombre lo indica, es el lapso en el que la CPU está prendida. Este tiempo puede ser calculado a partir del registro 70 para sistemas con el paquete RMF o del registro 1 para sistemas sin RMF.

Si durante un periodo específico en el LOG de la consola del sistema no se encuentran registrados los IPL's (INITIAL PROGRAM LOAD: serie de actividades para arrancar un sistema) el tiempo transcurrido registrado deberá ser igual al tiempo del reloj. Por lo tanto, una primera indicación de la disponibilidad del sistema puede obtenerse al dividir el tiempo transcurrido entre el tiempo de reloj de un periodo.

TIEMPO EN ESPERA (WAIT TIME): Es el tiempo que la CPU está prendida pero no está trabajando. Este tiempo es grabado en el registro 70 por RMF y en el registro 1 por SMF cuando no hay RMF.

TIEMPO REAL DE CPU: Es el tiempo total que la CPU es utilizada durante un periodo específico. Ni RMF ni SMF registran este tiempo, pero puede ser calculado restando del tiempo transcurrido el tiempo en espera para el mismo periodo.

UTILIZACION DE CPU: Es el porcentaje que se obtiene al dividir el tiempo real de CPU por 100 entre el tiempo transcurrido. Este porcentaje es obtenido también por otras herramientas de medición, por lo tanto se sugiere verificarlo durante el estudio para asegurar que SMF y RMF miden bien los datos anteriores.

A continuación definimos los parámetros que estaremos utilizando en el desarrollo del estudio:

TSMF = Total de tiempo de utilización de CPU capturado por - SMF

ERMF = ELAPSED TIME capturado por RMF (registros 70 ó 1)

WRMF = Tiempo en espera (WAIT TIME) capturado por RMF (registros 70 ó 1)

TRCPU = Tiempo real total de utilización de CPU. Es la diferencia entre el ERMF y WRMF

% U = Porcentaje de utilización de CPU. Es el cociente que resulta de dividir el TRCPU entre el ERMF.

% A = Porcentaje de disponibilidad. Es el cociente que resulta de dividir el ERMF entre el TR.

% P = Porcentaje de utilización de CPU debido a paginación. Se obtiene multiplicando el promedio de páginas/segundo por el factor de sobrecarga debido a la paginación.

PH = Tiempo de CPU dedicado a paginación.

BE = Punto de equilibro. Es el cociente de dividir el total de CPU utilizada para producción y desarrollo entre el total de CPU utilizada.

%UTP = Porcentaje de utilización de las aplicaciones en línea. Es el cociente de dividir las horas de utilización de - CPU para las aplicaciones en línea (TRCPUTP) entre el - ELAPSED TIME (ERMF).

CAP = Límite recomendable de horas de utilización de CPU según el ambiente de operación de que se trate (ver análisis de la disponibilidad y porcentaje de utilización de la CPU por periodo en este capítulo)

% C = Porcentaje de utilización de capacidad recomendada de CPU, que resulta de dividir el TRCPU entre el CAP.

TR = Tiempo total programado para el uso del computador.

TRCPUDU = Tiempo real de utilización de CPU por departamento - usuario

TSMFDU = Tiempo de CPU obtenido por SMF por departamento usuario.

FASES DEL ESTUDIO

Para poder realizar el estudio el proyecto se dividió en 2 - fases:

I.) Recolección y análisis del comportamiento de las cargas de trabajo actuales.

En esta primera fase se realizaron todas las actividades necesarias para la organización del proyecto, se determinó el periodo - para el cual se iba a hacer el análisis, se formaron los grupos de- trabajo, se definió el nivel de detalle en el que se desglosaría la información sobre las cargas de trabajo, se analizó el impacto de - las cargas de trabajo en la utilización de CPU, se confrontaron los datos obtenidos con guías basadas en la experiencia de otras insta-- laciones, se analizaron las diferencias significativas y se sugirie ron medidas para la optimización de algunos procedimientos.

II.) Proyección de cargas de trabajo y análisis de alterna- tivas.

En esta fase se estudiaron las diferentes fuentes que pudieran proprcionar información sobre el crecimiento estimado de las cargas de trabajo:

1.- Plan operativo

2.- Historia de la utilización por departamento

3.- Experiencia de la gerencia y los usuarios.

Los datos obtenidos de estas fuentes se compararon con los datos obtenidos de los meses analizados y se graficó el estimado de la capacidad de cómputo requerida para los años de 1981, 1982 y 1983. Se analizaron varias alternativas para satisfacer esta capacidad requerida dentro del mismo proveedor por la limitación puesta por los funcionarios de la Institución.

ACTIVIDADES REALIZADAS Y RESULTADOS OBTENIDOS

Las actividades y resultados obtenidos se detallan a continuación:

a) Desglose de las cargas de trabajo.

Lo primero que se hizo fue desglosar la carga de trabajo en grupos cuyas características y comportamiento fueran similares, con objeto de entender su comportamiento y poder tener mayores elementos para proyectar con mayor precisión los requerimientos futuros.

Es importante mantener un balance al desglosar la información de las cargas de trabajo para no sobrecargar la información y complicar demasiado el estudio. En este caso, por ejemplo, el desglose se efectuó dividiendo primeramente las cargas de trabajo en producción, desarrollo y soporte a la producción (soporte técnico del sistema, reprocesos, reorganizaciones, respaldos, programación de trabajos, mantenimiento de programas, etc.) Posteriormente cada uno de estos grupos se subdividió en aplicaciones en línea (TSO) y aplicaciones en lote. Después se seleccionaron las aplicaciones que representaban mayor carga al sistema y se agruparon por departamento usuario. Las aplicaciones no relevantes se agruparon en el renglón de "otros" (véase figura 6.2)

FIGURA 6.2 Desglose de la información de las cargas de trabajo.

C A R G A	PORCENTAJE
PRODUCCION EN LOTE	
GBG	38%
PRIVADAS	15%
SIST. ADMON	43%
OTROS	4%
PRODUCCION TSO	
GBG	15%
FSO Y GOBIERNO	14%
PRIVADAS	68%
OTROS	3%

DESARROLLO EN LOTE	PORCENTAJE
EDUCACION	13%
SIST. INF	82%
OTROS	5%
DESARROLLO TSO	
PRIVADAS	20%
SIST. INF	73%
OTROS	7%

SOPORTE EN LOTE	PORCENTAJE
PRODUCCION.....	30%
SOPORTE TECNICO	32%
MANTENIMIENTO	37%

SOPORTE TSO	
SOPORTE TECNICO	100%

FIGURA 6.2

b) Ajuste de las cifras obtenidas por SMF

En un sistema operativo MVS como el de la instalación citada,- el desglose de la utilización de CPU por aplicación se logra analizando dos bloques de control que se crean al inicio de cualquier -- trabajo.

1) TCB (TASK CONTROL BLOCK): Entre otras cosas, en este bloque de control se registra el tiempo de CPU atribuible directamente al programa de usuario.

2) SRB (SERVICE REQUEST BLOCK): Entre otras cosas, se registra parte del tiempo de CPU que el sistema operativo consume haciendo tareas para el programa del usuario correspondiente. Se subrayó "parte del tiempo" porque hay tiempos de CPU para algunas tareas del sistema operativo que no se registran, (véase figura 6.3) por ejemplo:

   a) responder comandos del operacor
   b) impresión y lectura de datos o instrucciones desde/ hacia el SPOOL
   c) los tiempos del SCHEDULER
   d) control y soporte de multiprogramación y memoria - virtual
   e) reintentos de operación de E/S no recuperadas, etc.

El SMF, único paquete de IBM que graba por programa los tiempos de CPU utilizados, se basa únicamente en la información de estos dos bloques, por lo que los tiempos obtenidos mediante esta herramienta representan parte del tiempo total de CPU consumido por - las aplicaciones. Por esta razón, en la metodología USAGE se esta-

DIVISION DEL TIEMPO DE CPU

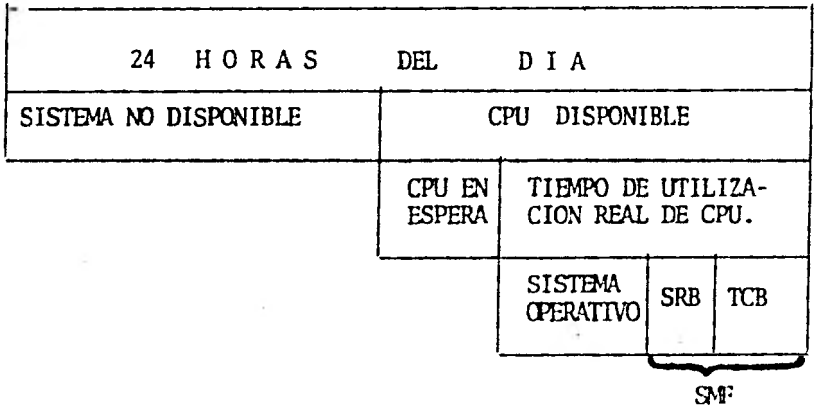


FIGURA 6.3

blece un procedimiento que permite ajustar la información obtenida por SMF. El objeto de este procedimiento es encontrar el porcentaje real de CPU que el SMF captura para cada aplicación. A este porcentaje se le llama CAPTURE RATIO y en este trabajo lo abreviaremos como CR. Este porcentaje varía entre 25% y 97% dependiendo de cómo los trabajos de la aplicación utilizan los recursos del sistema y de las características del sistema operativo en el que corren.

Para encontrar el CR por aplicación se utiliza la herramienta RMF que mide correctamente el ELAPSED TIME (ERMF) y el total de tiempo en espera de la CPU para un periodo dado (WRMF).

Con estos dos tiempos se puede calcular el tiempo total de utilización de CPU para todas las aplicaciones en el periodo (TRCPU) de la manera siguiente:

$$TRCPU = ERMF - WRMF$$

En este procedimiento se toman trabajos representativos de la aplicación o aplicaciones para las cuales se quiere obtener el CR. Estos trabajos seleccionados se corren solos en el computador y se calcula el CR de la siguiente forma:

$$CR = \frac{TSMF}{TRCPU - PH}$$

donde:

TSMF = Tiempo de utilización de CPU por aplicación capturado por SMF a partir de los registros:

5 para aplicaciones en lote

35 para TSO

PH = Tiempo de CPU utilizado para paginación que se calcula:

PH = %P X ERMF

%P = Porcentaje de utilización de CPU dedicado a paginación

que se calcula:

%P = Promedio de páginas/seg X factor estimado de paginación para el sistema dado.

El promedio de páginas por segundo para el periodo medido se obtiene del registro 71 grabado por RMF. El factor estimado de paginación es una guía dada por el proveedor que varía según el modelo de CPU, tipo de sistema operativo y nivel de actualización de éste. Por ejemplo, para el sistema estudiado (S/370-158-3 con MVS -- 3.7) se estima que una página por segundo requiere aproximadamente del 0.4% de CPU.

Es muy recomendable comparar los CR's obtenidos mediante este procedimiento con guías obtenidas mediante el análisis de aplicaciones típicas para varias instalaciones (COOPER) e investigar, en caso de diferencias significativas, las razones.

Por otro lado, el procedimiento de cálculo de CR's deberá repetirse hasta que:

$$\left[TRCPU - \sum_{i=1}^{n} \frac{TSMF_i}{CR_i} \right] \leq .10 \times TRCPU$$

donde n es igual al número de aplicaciones o grupo de aplicaciones con características diferentes en la instalación para los diferentes periodos de producción y con todas las aplicaciones corriendo normalmente.

Una vez encontrados los CR's por aplicación estos deberían permanecer fijos durante el estudio, y durante estudios posteriores, - si no sería muy difícil establecer comparaciones entre lo proyectado y lo real y validar las proyecciones efectuadas.

En la figura 6.4 se observan los CR's encontrados para los tipos de aplicación y sistema operativo de la instalación.

c) Análisis de la disponibilidad y porcentaje de utilización de CPU por periodo.

El primer paso para analizar la disponibilidad del sistema es calcular el tiempo transcurrido medido por los diferentes periodos de producción.

Como ya se dijo, éste se obtiene del registro 70, y deberá -- ser aproximadamente de 460 a 480 h mensuales para instalaciones -- que trabajan 5 días a la semana durante 24 h.

Si el ERMF calculado es considerablemente menor a 460 h. Para una instalación con 24 h al día de operación, es muy probable que - haya problemas de disponibilidad del sistema ya sea por fallas en - el equipo o en el SOFTWARE o por tiempos fuera de servicio debido a cambios o pruebas con máquina dedicada.

Si esta situación se presenta en varios de los meses estudia-- dos, es necesario sugerir mejoras en los procedimientos de opera--- ción y mantenimiento correspondientes para mejorar la situación.

Si el ERMF es considerablemente mayor a 480, esto puede indi-- car dos cosas:

1) El sistema puede permanecer prendido durante largo tiempo - sin realizar trabajo, en cuyo caso el tiempo en espera tam-

CR POR TIPO DE APLICACION

APLICACION	CR
Producción en lote	0.89
Desarrollo en lote	0.81
Soporte en lote	0.62
Producción TSO	0.50
Desarrollo TSO	0.41
Soporte TSO	0.48

FIGURA 6.4

bién será muy alto y por lo tanto, para no distorsionar las cifras será necesario restar manualmente del ERMF el tiempo que el sistema ha pasado prendido sin estar programado para hacer trabajo. Este tiempo, que llamaremos "tiempo no programado para trabajo" puede ser calculado a partir de los - LOGS de consola, y básicamente se deberá a que los fines de semana se deja prendido el equipo.

2) El sistema puede estar sobreutilizado: éste es el primer indicio de que el sistema requiere más recursos para procesar la carga de trabajo de la instalación.

   En las instalaciones que se encuentran en este caso, la producción y niveles de servicio son fuertemente impactadas -- por cualquier falla, o por cualquier suspensión debida a -- mantenimiento del HARDWARE o del SOFTWARE, por lo que poco- a poco van dejando de lado estas actividades, con lo que la disponibilidad del sistema empeora, y por lo tanto se crea- un círculo vicioso.

El segundo paso en esta fase es calcular el tiempo real de CPU- y el procentaje de utilización por periodo de producción utilizando las siguientes fórmulas:

$$TRCPU = ERMF - WRMF$$

$$\%U = \frac{TRCPU}{ERMF} \times 100$$

El porcentaje de utilización promedio para un periodo puede ser obtenido utilizando otras herramientas de medición; por lo tanto, - para efectos de verificación, es recomendable obtenerlo por otros -

medios. La utilización real de CPU (TRCPU) y el porcentaje de utilización promedio (%U) son medidas que nos permiten, en primera instancia, darnos cuenta de la capacidad disponible del computador.

Se dice que la CPU ha llegado a la máxima capacidad razonable para un periodo cuando:

$$TRCPU = .9 \times ERMF$$

o lo que es lo mismo cuando:

$$\%U = 90$$

Esto significa que para instalaciones con tres turnos de operación durante 5 días a la semana la CPU ha llegado a su máxima capacidad cuando:

$$TRCPU \approx 400 \text{ h}$$

Cuando el TRCPU es considerablemente menor a 400 h el sistema deberá investigarse, ya que puede ser que esté mal afinado y haya cuellos de botella en algunos componentes de E/S.

Otro punto muy importante con respecto a estas medidas es que el porcentaje de utilización máximo recomendable varía según el medio ambiente de proceso de la instalación (véase la fig. 6.5).

Por otro lado existen trabajos en lote que requieren de un tiempo de respuesta ágil; por ejemplo algunas instituciones consideran importante mantener una respuesta rápida para compilaciones y pruebas; por lo que dependiendo de los objetivos de tiempo de respuesta que se definan, en la figura 6.5 se sugieren porcentajes máximos de utilización de CPU de tal forma que si se rebasan dichos porcentajes los niveles de servicio en el tiempo de respuesta para sistemas en línea en momentos críticos y para trabajos en lo-

UTILIZACION MAXIMA TEORICA DE CPU

TIPO DE CARGA DE TRABAJO	TIEMPO REQUERIDO DE RESPUESTA PARA TRABAJOS EN LOTE	% U
APLICACIONES EN LINEA SOLAMENTE		40
APLICACIONES EN LOTE Y EN LINEA	APROX 30 MIN	50
APLICACIONES EN LOTE Y EN LINEA	APROX 60 MIN	60
APLICACIONES EN LOTE Y EN LINEA	APROX 3 H	70
APLICACIONES EN LOTE Y EN LINEA	SIN RESTRICCION	90

FIGURA 6.5

te con las características citadas se ven seriamente degradados.

En la figura 6.5, a partir del segundo renglón, se apunta el porcentaje de utilización recomendable para la mezcla de trabajos en línea y en lote. Es importante aclarar que para cualquier caso el porcentaje máximo de utilización de CPU para sistemas en línea con tiempos de respuesta críticos es 40% considerando que en ciertos momentos la demanda es 2.5 veces mayor al promedio, lo cual es una situación común en la mayoría de las instalaciones.

Es importante aclarar que éstas son simples recomendaciones, basadas en la experiencia, que hay que adaptar de acuerdo con las características específicas de la instalación. En estas recomendaciones se considera que el tiempo de respuesta para las aplicaciones en línea es un factor crítico, y que por lo tanto es necesario dejar CPU disponible para los periodos en los que el número de transacciones que llegan al computador crece al máximo.

En la figura 6.6 se muestra el análisis efectuado a partir de los tiempos ERMF y WRMF para el mes de junio. En el periodo 1 el límite de capacidad se calculó como:

$$CAP = .7 \times ERMF$$

a diferencia de los periodos 2 y 3 donde se calculó como:

$$CAP = .9 \times ERMF$$

Esto se debe a que durante el periodo 1 se debe mantener un buen tiempo de respuesta para las aplicaciones en línea (TSO) y un tiempo de respuesta promedio de 2 h para las aplicaciones en lote. No así en el caso de los otros dos turnos en donde la respuesta para el proceso en lote no es tan crítica y las aplicaciones en lí--

CAPACIDAD UTILIZADA Y DISPONIBILIDAD DEL SISTEMA
DURANTE EL MES DE JUNIO DE 1980.

		PERIODO 1 (P1) HRS.	PERIODO 2 (P2) HRS.	PERIODO 3 (P3) HRS.	FIN DE SEMANA (W) HRS	T O T A L
ERMF	ELAPSED TIME OBTENIDO DEL REGISTRO 70 (RMF)	192.99	126.75	100.40	51.7	471.86
WRMF	WAIT TIME OBTENIDO DEL REGISTRO 70 (RMF)	63.68	47.45	58.19	32.57	201.59
TRCPU	UTILIZACION REAL CPU ERMF-WRMF	129.3	79.2	42.21	19.1	269.86
%U	PORCENTAJE DE UTILIZACION TRCPU/ERMF	67%	62.6%	42%	37%	57.3%
CAP	LIMITE DE CAPACIDAD P1 = .7 X ERMF P2,P3,W=.9 X ERMF	135	114	90.3	46.5	424.6
%C	CAPACIDAD UTILIZADA TRCPU/CAP	95.7%	70%	47%	41%	63.5
TR	TIEMPO RELOJ HRS. TURNO X DIAS/MES	231	147	126	56	560
%A	ERMF/TR	83%	86%	79%	92%	84%

FIGURA 6.6

nea disminuyen notablemente.

Del análisis como el realizado en la figura 6.6 se obtuvieron las figuras 6.7 y 6.8 de cuyo análisis se desprende lo siguiente:

1) la utilización del equipo en el turno 1 excede, para los meses de octubre y noviembre, los límites recomendables para mantener un buen tiempo de respuesta en el TSO y las dos horas planteadas como objetivo para el proceso en lote.
2) Suponiendo que no se pudiera redistribuir la carga, lo cual se analizará más adelante, se tiene un primer indicio de mayores requerimientos de equipo.
3) Aunque los promedios de disponibilidad no son malos, sería posible disponer de más capacidad si se pudiera aumentar el porcentaje a 92 ó 93 por ciento, lo cual es posible en algunas instalaciones.
4) Puede considerarse que el sistema tiene una utilización adecuada (mayor al 50%). El tener un porcentaje inferior significaría que existen cuellos de botella en los dispositivos de E/S o que el equipo está subutilizado y por lo tanto existe bastante capacidad disponible.

d) Comportamiento de las cargas de trabajo por aplicación.

Una vez que se ha analizado el comportamiento de la carga de trabajo en forma global, y que se han encontrado medios para ajustar la información obtenida por medio del SMF, es necesario desglosar la información sobre la utilización del sistema, para lo cual es necesario analizar los registros 5 y 35 grabados por medio del SMF.

PORCENTAJE DE CAPACIDAD UTILIZADA (EN %)
RESPECTO DE LA UTILIZACION MAXIMA RECOMENDADA

		PERIODO 1	PERIODO 2	PERIODO 3	PROMEDIO
1979	JULIO	59	42.3	33.9	45.1
	OCTUBRE	87.8	66.9	52	69
1980	ENERO	43.8	37.1	28.5	36.7
	MAYO	80.2	56.6	42.2	59.7
	JUNIO	95.7	53.7	41	63.5
	JULIO	83	64.56	50.4	66
	AGOSTO	81	67.8	57.1	67
	SEPTIEMBRE	92.1	86.3	70	82.7
	OCTUBRE	107.2	80.7	66.7	84.9
	NOVIEMBRE	105	82	65	83.7
	DICIEMBRE				

FIGURA 6.7

PORCENTAJE DE DISPONIBILIDAD

		PERIODO 1	PERIODO 2	PERIODO 3	PROMEDIO
1979	JULIO	92%	89%	84%	90%
	OCTUBRE	87%	91%	78%	88%
1980	ENERO	88%	85%	87%	86%
	MAYO	90%	93%	90%	91%
	JUNIO	83%	89%	79%	85%
	JULIO	89%	92%	88%	90%
	AGOSTO	87%	88%	84%	87%
	SEPTIEMBRE	85%	76%	72%	79%
	OCTUBRE	81%	86%	90%	84%
	NOVIEMBRE				
	DICIEMBRE				
PROMEDIO TOTAL..:		87%	88%	83%	87%

FIGURA 6.8

La herramienta SLR resume estos registros y proporciona reportes tan detallados como sea necesario. Por ejemplo, en este estudio se obtuvieron los resultados siguiendo los siguientes niveles de detalle:

- Departamento usuario
- Tipo de aplicación
  - En lote
  - TSO
- Objetivo del proceso
  - Producción
  - Desarrollo de aplicaciones
  - Soporte de producción

En las figuras 6.9, 6.10 y 6.11 se muestran los resultados obtenidos para el mes de junio/80. Es importante verificar para cada mes que:

.5 X TRCPU < total de producción + total de desarrollo + total de soporte = TSMF, en este caso:

TRCPU = 269.86 (figura 6.6)

Total de producción = 122.1 (figura 6.9)

Total de desarrollo = 51.6 (figura 6.10)

Total de soporte = 21.02 (figura 6.11)

entonces:

TSFM = 194. 72 > .5 X 269.86 = 134.93

si TSMF fuera menor que el 50% del TRCPU podría ser por alguna, o - varias, de las siguientes razones:

1) Se están corriendo en el sistema tareas de larga duración-

HORAS DE CPU OBTENIDAS POR SMF (PRODUCCION)

PRODUCCION	No. DE REGISTRO DE SMF	PERIODO 1	PERIODO 2	PERIODO 3	FIN DE SEMANA	TOTAL
EN LOTE	5					
GBG		9.89	10.81	15:49	6.72	42.91
PRIVADAS		6.35	4.56	4.94	1.39	17.24
SIST. ADMON.		11.43	24.11	10.09	4.44	50.07
OTROS		2.57	1.31	0	0	3.88
TOTAL EN LOTE		30.25	40.8	30.52	12.55	114.1
TSO	35					
GBG		1.01	.28	0	0	1.29
FSO GOBIERNO		1.07	.18	0	0	1.25
PRIVADAS		3.24	2.08	.1	.04	5.46
TOTAL TSO		5.32	2.54	.1	.04	8.00
TOTAL PRODUCCION		35.57	43.34	30.62	12.59	122.1

FIGURA 6.9

HORAS DE CPU OBTENIDAS POR SMF (DESARROLLO)

DESARROLLO	No. DE REGISTRO SMF	PERIODO 1	PERIODO 2	PERIODO 3	FIN DE SEMANA	TOTAL
EN LOTE	5					
EDUCACION		3.35	.57	0	0	3.92
SIST. INF.		15.50	10.61	3.96	1.45	31.52
TOTAL EN LOTE		18.85	11.18	3.96	1.45	35.44
TSO	35					
PRIVADAS		1.50	.58	.6	.20	2.88
SIST. INF.		10.47	1.97	0	.21	12.65
OTROS		.98	.18	0	.0	1.16
TOTAL TSO		12.96	2.73	.6	.41	16.69
TOTAL DESARROLLO		31.81	13.91	4.02	1.86	51.6

FIGURA 6.10

HORAS DE CPU OBTENIDAS POR SMF (SOPORTE DE PRODUCCION)

SOPORTE	No. DE REGISTRO DE SMF	PERIODO 1	PERIODO 2	PERIODO 3	FIN DE SEMANA	TOTAL
EN LOTE	5					
PRODUCCION		2.30	.73	1.34	1.41	5.78
SOPORTE TECNICO		3.14	1.56	1.12	.26	6.08
MANTENIMIENTO		3.06	2.59	1.23	.18	7.06
TOTAL EN LOTE		8.50	4.88	3.69	1.85	18.92
TSO	35					
TOTAL TSO		1.53	.25	.23	.09	2.1
TOTAL SOPORTE		10.03	5.13	3.92	1.94	21.02

FIGURA 6.11

que tienen en las tarjetas de control (JCL) el parámetro - TIME=1440. La codificación de este parámetro en las tarjetas de control de un trabajo (JOB) indica al sistema operativo que no debe registrar el tiempo de CPU consumido para ese trabajo. En esta situación podrían encontrarse algunos de los paquetes que controlan las comunicaciones cuando se tiene servicio en línea, por ejemplo: IMS y CICS. - (El TSO no se encuentra en esta situacion).

En este caso es necesario estimar el tiempo de CPU mediante otros medios, por ejemplo otras herramientas de medición.

2) Es una instalación con servicios de TSO básicamente. En - este caso, si los CR's escogidos son adecuados se remediará la situación.

3) Puede ser indicio de que la memoria principal no es suficiente y que por lo tanto hay una paginación excesiva. En este caso deberá corroborarse con reportes sobre la paginación durante el periodo. (Ver el ejemplo del reporte en la figura 6.12)

En este punto es necesario, al mismo tiempo que se hace el análisis, verificar personalmente los estándares de operación y cómo los siguen los usuarios del centro de cómputo, ya que se puede llegar a conclusiones erróneas si solamente se basa uno en los reportes obtenidos.

La razón es que muchas veces se pasan por alto los estándares de operación. Por ejemplo, por displicencia o para ob-

SLR/USAGE

PAGE: 0001
DATE: 80 JUL 04
TIME: 10:22:34

MVS PAGING ACTIVITY

MVS SYSTEM PAGING

MONTH	SYSID	WP	PAGING RATE PAGES/SEC
JUNT	HI58	PI	6.68
		P2	I.I6
		P3	0.2I
		WEEKEND	0.24
		TOTAL	3.II

YEAR = 80
MONTH = JUN
SYSID = H158

FIGURA 6.12

tener mayor prioridad de selección, se utilizan tarjetas de control que indican "prueba" para correr trabajos de "producción"; los nombres de los trabajos no siguen los estándares, y por lo tanto no se puede hacer una separación adecuada de utilización por departamento o aplicación; tiempos de trabajos que corren en forma continua durante dos o más turnos de producción son cargados únicamente al turno en el que terminó el trabajo, etc.

Como resultado de este análisis se deberán ajustar manualmente los reportes obtenidos.

e) Ajuste de las cifras obtenidas por aplicación.

En este paso el objetivo es calcular la utilización real de CPU por aplicación, aplicándole el CR correspondiente.

Esto se logra con:

$$TRCPUDU = \frac{TSMFCPU}{CR}$$

En las figuras 6.13, 6.14 y 6.15 se muestran los resultados de este ajuste.

f) Estimar el tiempo de CPU dedicado a paginación.

Este tiempo, como se dijo anteriormente, no se registra por SMF, por lo que hay que estimarlo de la manera siguiente:

$$PH_i = \% P_i \times ERMF_i$$

con $i = 1,2,...4$

y $\% P_i$ = promedio de páginas/seg en el periodo iX.004

En la figura 6.16 se muestran los resultados.

HORAS REALES DE CPU POR APLICACION (PRODUCCION)

PRODUCCION	CR	PERIODO 1	PERIODO 2	PERIODO 3	FIN DE SEMANA	T O T A L
EN LOTE	.89					
GBG		11.11	12.14	17.40	7.55	48.21
PRIVADAS		7.13	5.12	5.55	1.56	19.37
SIST. ADMON.		12.8	27.08	11.33	4.98	56.25
OTROS		2.8	1.47	0	0	4.35
TOTAL EN LOTE		33.98	45.84	34.29	14.10	128.20
TSO	.50					
GBG		2.02	.56	0	0	2.58
FSO GOBIERNO		2.14	.36	0	0	2.50
PRIVADAS		6.4B	4.16	.2	.08	10.92
TOTAL TSO		10.64	5.08	.2	.08	16.00
TOTAL PRO-DUCCION		44.62	50.92	34.49	14.18	144.20

FIGURA 6.13

HORAS REALES DE CPU POR APLICACION (DESARROLLO)

DESARROLLO	CR	PERIODO 1	PERIODO 2	PERIODO 3	FIN DE SEMANA	T O T A L
EN LOTE	.81					
EDUCACION		4.13	.70	0	0	4.83
SIST. INF.		19.13	13.09	4.88	1.79	38.91
TOTAL EN LOTE		23.27	13.80	4.88	1.79	43.75
TSO	.41					
PRIVADAS		3.65	1.41	1.46	.48	7.02
SIST. INF.		25.53	4.80	0	0	30.85
OTROS		2.39	.43	0	.51	2.82
TOTAL TSO		31.60	6.65	1.46	1.00	40.70
TOTAL OESARROLLO.		54.87	20.45	6.34	2.79	84.45

FIGURA 6.14

HORAS REALES DE CPU POR APLICACION (SOPORTE)

SOPORTE	CR	PERIODO 1	PERIODO 2	PERIODO 3	FIN DE SEMANA	T O T A L
EN LOTE	.62					
PRODUCCION		3.70	1.17	2.16	2.27	9.32
SOPORTE TECNICO		5.06	2.51	1.80	.41	9.80
MANTENIMIENTO		4.93	4.17	1.98	.29	11.38
TOTAL EN LOTE		13.70	7.87	5.95	2.98	30.51
TSO	.48					
TOTAL TSO		3.18	.52	.47	.18	4.37
TOTAL SOPORTE		16.88	8.39	6.42	3.16	34.88

FIGURA 6.15

g) Sumarización y análisis.

En las figuras 6.16, 6.18, 6.19, 6.20, 6.21 y 6.22 se muestra un resumen de resultados que permite completar el análisis sobre el medio ambiente de operación de la instalación y sobre el comportamiento de las cargas de trabajo. Del análisis de estos cuadros se desprende lo siguiente:

1.- Se puede tener confianza en los CR's utilizados para ajustar las cifras de SMF, ya que estos se consideran adecuados cuando:

/ TRCPUSMF-TRCPU/ $\leq$ .1 C TRCPU

(VALOR ABSOLUTO)

Si no sucede así, podría ser por alguna de las siguientes razones:

- Pudo haber algún error al calcular la utilización de CPU, debido a la paginación. Los promedios de pag/seg se deben tomar por periodo de producción y no globales en el mes porque esto puede distorsionar las cifras.
- Durante el mes en que no concuerdan las cifras pudieron haberse ejecutado trabajos eventuales que no siguen los patrones normales (altos volúmenes de impresión, utilización del concepto de SHARED SPOOL, etc.)
- El TSO pudo haber sido utilizado en forma poco usual para alimentar trabajos en lote o para trabajos orientados a mucha utilización de CPU. En estos casos el CR para el TSO debería ser mucho más alto.

HORAS ESTIMADAS DE CPU DEDICADAS A PAGINACION

		PERIODO 1	PERIODO 2	PERIODO 3	FIN DE SEMANA	
ERMF	ELAPSED TIME	192.99	126.75	100.40	471.86	
P/S	PROMEDIO DE PAGINAS POR SEGUNDO.	6.68	1.16	.21	.24	
%P	PORCENTAJE DE UTILIZACION DE CPU DEBIDO A PAGINACION. P/S X .4	2.6	.4	0	0	TOTAL
PH	HORAS DE CPU AL MES POR PAGINACION. ERMF X %P	5.01	.5	0	0	5.51

FIGURA 6.16

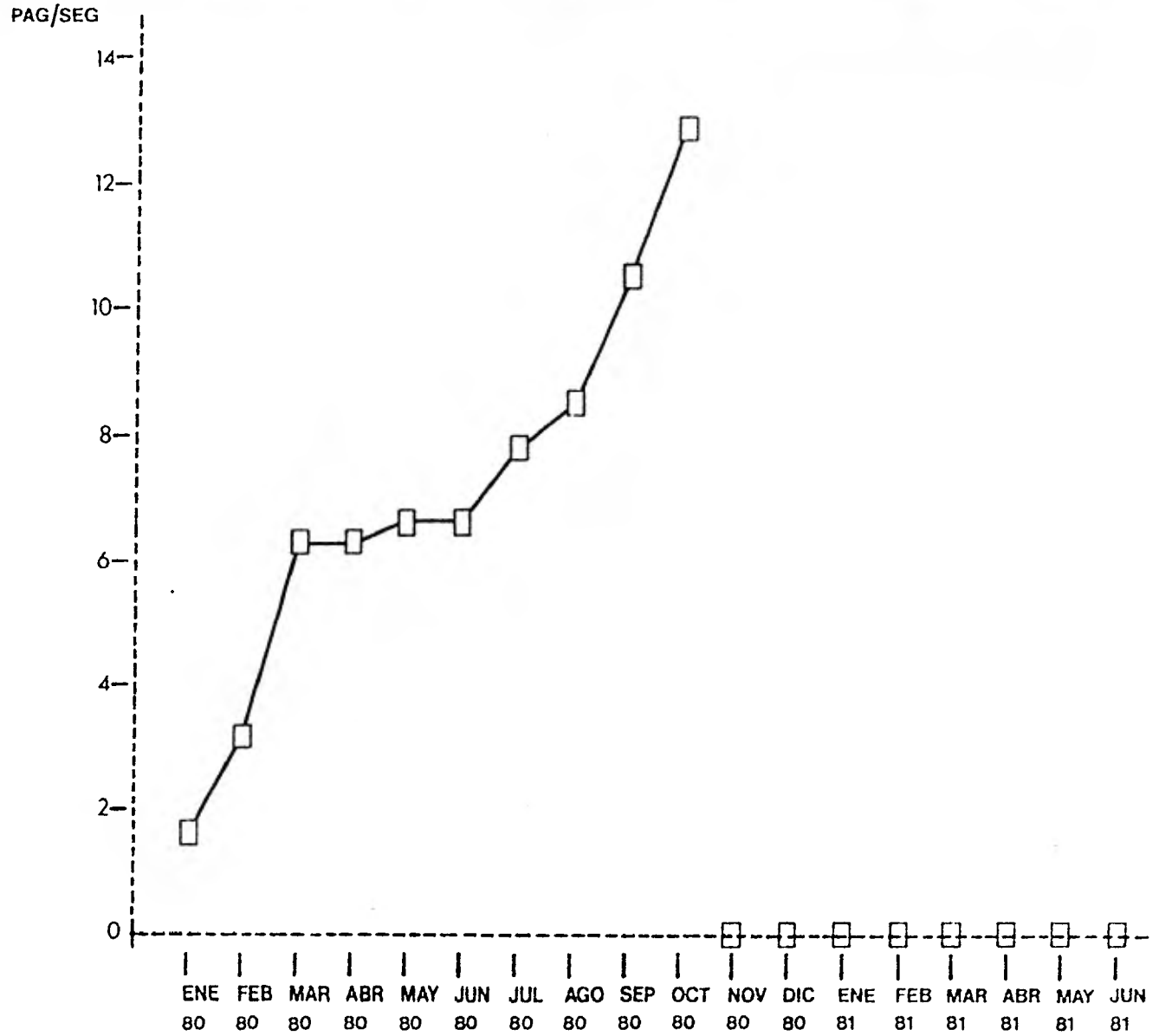
RESUMEN DE RESULTADOS

		PERIODO 1	PERIODO 2	PERIODO 3	FIN DE SEMANA	TOTAL
TRCPUP	UTILIZACION DE CPU EN PRODUCCION	44.62	50.92	34.49	14.18	144.20
TRCPUD	UTILIZACION DE CPU DESARROLLADO	54.87	20.45	6.34	2.79	84.45
TRCPUS	UTILIZACION DE CPU SOPORTE DE PRODUCCION	16.88	8.39	6.42	3.16	34.88
PH	UTILIZACION DE CPU PAGINACION	5.01	.5	0	0	5.51
TRCPUSMF	UTILIZACION REAL DE CPU CALCULADA A PARTIR DE SMF	121.38	80.26	47.25	20.13	269.02
TRCPU	UTILIZACION REAL DE CPU OBTENIDA POR RMF	129.3	79.2	42.21	19.1	269.86
%D	PORCENTAJE DE DIFERENCIA ENTRE LAS DOS MEDIDAS	6.2	1.4	10.7%	5.2%	0%
BE	PUNTO DE EQUILIBRIO (P+T) / TRCPUSMF	82%	89%	86%	84%	85%
TRCPUTP	UTILIZACION REAL DE CPU PARA APLICACIONES EN LINEA	45.42	12.25	2.13	1.26	61.07
%UTP	PORCENTAJE DE UTILIZACION DE CPU POR TP	23.5%	9.6%	2.1%	2.4%	12.9
%UTR	PORCENTAJE DE UTILIZACION PARA SOPORTE A LA PRODUCCION (REPROCESOS, RESPALDOS, RECUPERACION, ETC).	3%	1%	4.5%	11.%	3%

FIGURA 6.17

		PERIODO 1	PERIODO 2	PERIODO 3	FIN DE SEMANA
1 9 8 0	ENERO	1.51	1.20	--	--
	MARZO	6.37	0.97	0.16	0.24
	ABRIL	6.29	1.48	0.33	0.96
	MAYO	6.72	0.67	0.10	0.21
	JUNIO	6.68	1.16	0.21	0.24
	JULIO	8.09	2.04	0.42	0.44
	AGOSTO	8.82	2.08	0.30	0.42
	SEPTIEMBRE	10.71	2.58	1.31	0.88
	OCTUBRE	13.35	2.35	0.76	0.62
	NOVIEMBRE				
	DICIEMBRE				

FIGURA 6.18


PAG/SEG
14
12
10
8
6
4
2
0
ENE 80
FEB 80
MAR 80
ABR 80
MAY 80
JUN 80
JUL 80
AGO 80
SEP 80
OCT 80
NOV 80
DIC 80
ENE 81
FEB 81
MAR 81
ABR 81
MAY 81
JUN 81

Grafica 6.19

UTILIZACION DE CPU EN 1979 Y 1980

PERIODO 1 Y TOTAL

	1979						1980						
	Total horas de CPU Periodo 1	Porcentaje Utilización CPU Periodo 1	Horas de CPU TSO Periodo 1	Porcentaje Utilización de CPU TSO	Total horas de CPU	Porcentaje de utilización CPU	Total horas de CPU Periodo 1	Porcentaje utilización CPU Periodo 1	Horas de CPU TSO Periodo 1	Porcentaje de utilización CPU - TSO.	Total horas de CPU	Porcentaje de utilización - CPU	Porcentaje de crecimiento de utilización en los años 1979-1980
	71	35.3	10	4.8	191	39.7	59	29	17	8.3	158	33	- 16.7
	61	11.8	10	4.8	232	48.3	96	47.7	28	13.9	258	53.7	5.6
	52	42.1	12	6	245	51	110	54.6	33	16.4	298	61	11.5
	63	46	13	6.4	252	52.5	126	63	37	18.4	326	67.9	29.3
	74	37	10	4.8	203	43.3	111	53.1	41	19.6	269	53.8	15.1
	64	32	8	3.9	175	36.4	121	62.6	45	23.3	269	56.9	34.1
	56	33.9	16	4.6	203	40.6	114	55.3	50	24.2	297	59.4	46
	50	17	22	10.9	211	50.2	109	54	48	23.7	298	60.6	41.8
	123	61.1	32	15.9	278	63.1	127	64.1	55	27.7	327	74.4	30.5
	115	58.4	35	17.3	303	62.9	134	71.2	55	29.2	356	76.3	12.5
	107	53.2	30	14.9	279	58.1	137	72	58	28.8	354	74.1	19.6
	1 000		192		2 624		1 221		467		3 231		13.6

FIGURA 6.20

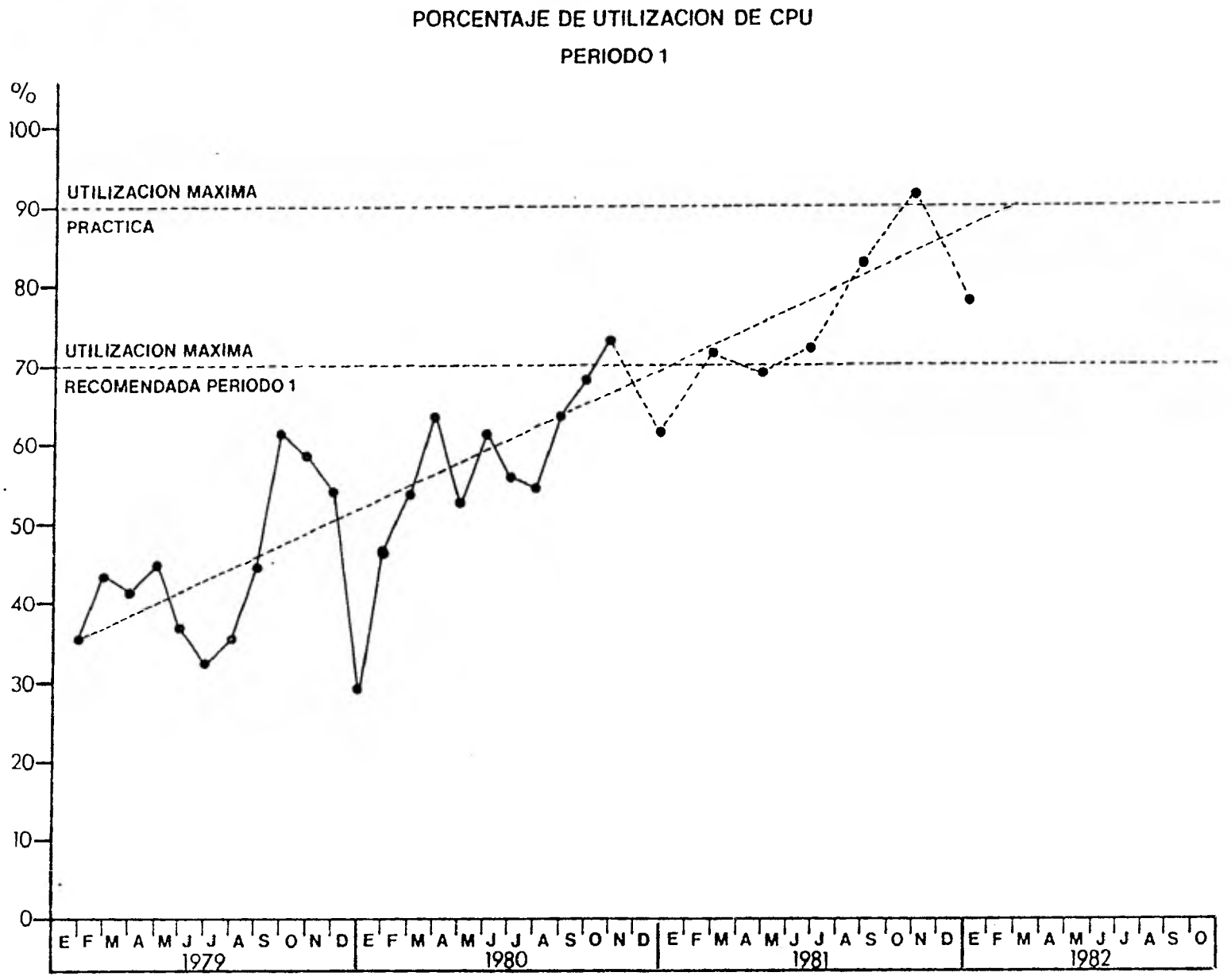

PORCENTAJE DE UTILIZACION DE CPU
PERIODO 1
%
100
90
80
70
60
50
40
30
20
10
0
UTILIZACION MAXIMA
PRACTICA
UTILIZACION MAXIMA
RECOMENDADA PERIODO 1
E F M A M J J A S O N D
1979
E F M A M J J A S O N D
1980
E F M A M J J A S O N D
1981
E F M A M J J A S O
1982

Figura 6.21

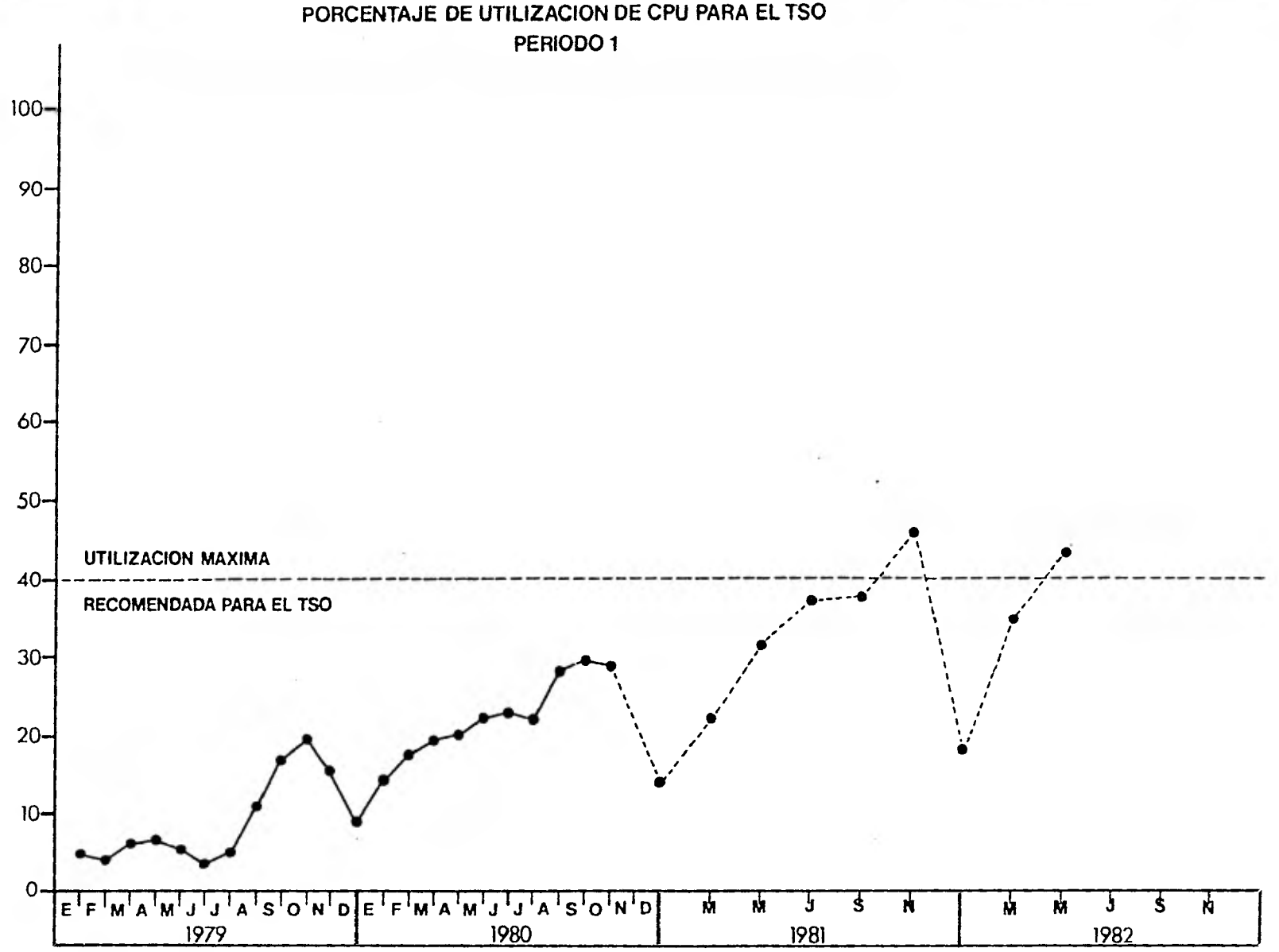

PORCENTAJE DE UTILIZACION DE CPU PARA EL TSO
PERIODO 1
100
90
80
70
60
50
40
30
20
10
0
UTILIZACION MAXIMA
RECOMENDADA PARA EL TSO
E F M A M J J A S O N D
1979
E F M A M J J A S O N D
1980
M M J S N
1981
M M J S N
1982

Figura 6:22

2.- Hay un balance adecuado entre las tres categorías de proceso, ya que:

PRODUCCION: 54%

DESARROLLO: 31%

MANTENIMIENTO: 10%

PROCESOS Y RESPALDOS: 3%

Los porcentajes recomendables para cada una de estas categorías son:

$\% \ 60 \leq$ PRODUCCION $\leq 70\%$

$\% \ 20 \leq$ DESARROLLO $\leq 30\%$

$\% \ 8 \leq$ MANTENIMIENTO $\leq 13\%$

PAGINACION $\leq 6\%$

PROCESOS Y RESPALDOS $\leq 5\%$

Obviamente esto dependerá del tipo de instalación. Por ejemplo, en la instalación estudiada es un poco bajo el porcentaje de producción con respecto al desarrollo, aunque en este caso no es preocupante porque la instalación tiene un área "educación" que incrementa las cifras de desarrollo, área que normalmente no se presenta en las demás instalaciones.

3. En el caso de la paginación, aunque el promedio de páginas/seg obtenido en ese mes no representaba problema en los cuadros anuales (figura 6.18 y 6.19), se observa que la memoria podría ser un cuello de botella a principios de 1981, suponiendo el mismo patrón de crecimiento. Ya que para el mes de octubre el porcentaje de utilización de CPU para pa

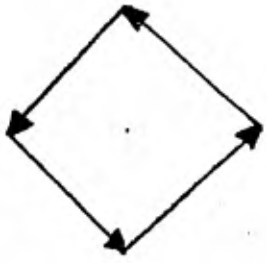
ginación sería de 10.31%. Por lo tanto éste fue un factor considerado en el análisis de alternativas.

4. En una primera instancia se puede asumir que el computador es rentable, ya que el punto de equilibrio es de 85%. Esta es una medida muy general que indica el porcentaje de utilización de CPU para trabajo productivo; se aconseja que sea mayor o igual al 55%. En este caso se presupone que para cada aplicación se ha hecho un análisis de costo/beneficio que justifique su existencia o su necesidad de desarrollo.
5. La utilización de CPU para la aplicación en línea (TSO) en el periodo 1 está dentro del límite recomendable, ya que si el porcentaje de utilización fuera mayor o igual a 40%, la CPU no sería capaz de absorber adecuadamente los periodos de cargas pico que caracterizan a las aplicaciones en línea. En muchos casos estas cargas pico llegan a ser hasta 2.5 veces la carga promedio, por lo que en esos periodos, si se tiene una utilización promedio de CPU mayor al 40%, el incremento en la carga repercute en el tiempo de respuesta o en el nivel de servicio.
6. Aunque el porcentaje de utilización del TSO está dentro de los límites recomendables, el porcentaje de utilización total para el periodo 1 excedía del límite recomendable (70%) para los meses de septiembre y octubre, situación que repercutió en la satisfacción de los usuarios, particularmente en los programas de aplicaciones y de sistemas, que en horas

de carga pico recibían un mal tiempo de respuesta, y por lo tanto bajaba mucho su productividad. Cada vez que daba menos tiempo para prueba y optimización de aplicaciones y sistema operativo, por lo que la instalación entró en un peligroso círculo vicioso.

MENOS HORAS DE CPU DISPONIBLES

UTILIZACION DE CPU CRITICA



MAYOR PROBLEMATICA EN MATERIA DE SOFWARE Y MAYOR INCIDENCIA EN - SOPORTE A LA PRODUCCION

MENOS TIEMPO PARA PRUEBAS Y MEJORAS

7. Otra cifra que abunda en lo anterior es la relación que se supone debe haber en la utilización de CPU para desarrollo entre el primero y los demás turnos. El primer turno debería representar el 75% de la utilización de CPU; que no sea así, significa que los programadores se están quedando horas extra para poder desarrollar su trabajo, lo cual no es conveniente por ningún motivo.

   En nuestro ejemplo es necesario tomar las cifras de la figura 6.10 correspondiente a los renglones en lote (sist.inf), ya que son las que representan el desarrollo de nuevas aplicaciones para la institución.

   Por tanto:

   TRCPUD 1 = 15.50 + 10.47 = 25.97 = 58%

TRCPUD23W = 10.61 + 3.96 + 1.97 + .21 = 18.2 = 41%

La obtención de estas cifras nos hace ver que si no hay un tiempo de respuesta adecuado para el desarrollo de aplicaciones en el primer turno, los programadores usarán cada vez más el sistema en los turnos siguientes y durante el fin de semana, por lo que también en este punto se observa la necesidad de encontrar una solución para tener mayor capacidad del sistema.

8. Los dos puntos anteriores sugirieron en una primera instancia la optimización en la programación de trabajos con objeto de liberar a la CPU en el primer turno del proceso en lote que no requiriera correr en ese momento. Desgraciadamente, al entrar al detalle de las aplicaciones, se encontró que solamente se podía reprogramar un 10% de la carga de trabajo, concretamente los sistemas de producción de soporte a los departamentos administrativos, ya que el resto de la carga provenía de departamentos usuarios a los cuales se les tenía que proporcionar servicio en el periodo 1 con un tiempo promedio de respuesta de dos horas.

Proyección del crecimiento y análisis de alternativas.

Una vez definida la necesidad de hacer algo que permitiera mantener los niveles de servicio en el periodo 1, pues los tiempos de respuesta empezaban a degradarse sobre todo para los periodos pico de carga de trabajo (10 a 13 y 15 a 17.30 h), se hizo un análisis de alternativas que permitieran satisfacer adecuadamente los requerimientos de servicio para los años de 1981, 1982 y 1983. Para esto se proyectaron los requerimientos de cómputo en esos años con base en varias fuentes que permitieron estimar el crecimiento futuro. Primeramente se consutó el plan operativo de la empresa, en donde se estimaba un crecimiento del 30% anual de las operaciones; posteriormente se consultó a la gerencia de DP para conocer los planes específicos en cuanto a desarrollo e implantación de nuevas aplicaciones y se proyectó la utilización de la CPU con base en la información de los años 1979 y 1980.

La gerencia de DP estimó que el incremento de la carga de trabajo por nuevas aplicaciones o por el incremento en los volúmenes de las aplicaciones en lote ya instaladas seguiría el mismo patrón de crecimiento que el de los años anteriores, esto es 15% anual. (véase Fig. 6.20)

En el caso del TSO se consideró un porcentaje mucho mayor de crecimiento (60% el primer año, 35% el segundo, 20%, el tercero y el cuarto) debido a que esta era una herramienta que no se utilizaba todavía en la medida adecuada porque había sido instalada recientemente (finales de 1978) y porque en un principio (los primeros 9 meses)

fue orientada únicamente al apoyo del grupo técnico. Por lo tanto, - había una demanda latente en los posibles usuarios que en el momento- del estudio no la utilizaban, ya sea porque no la conocían adecuada-- mente o porque no había terminales suficientes, pero que por las ca-- racterísticas de sus funciones podía serles útil.

Por otro lado, también se indicó que el Departamento de ventas tenía intenciones de cambiar el sistema 370/158 por un modelo 4341 - con tecnología más moderna y mejor precio/rendimiento que el siste- ma 370, pero con menor poder de cómputo. El poder de cómputo se -- puede estimar utilizando la capcidad de MIPS (millones de instruc-- ciones por segundo) de la CPU. (En la fig. 6.23 se presentan los - MIPS para cada CPU mencionada en este estudio). Este cambio en el- computador que apoyaba al Departamento de ventas e investigación re percutió en la carga de trabajo del sistema estudiado, pues el APL- (A PROGRAM LANGUAGE), paquete con los mismos principios de un siste ma de tiempo compartido pero orientado a la solución de problemas - matemáticos y de investigación, pasaría del computador de ventas al de producción.

PODER DE COMPUTO EN MIPS (MILLONES DE INSTRUCCIONES POR SEGUNDO)

MODELO	MIPS
370/158 (UP)	.91
370/158 (AP)	1.4
4341	.75
3032	2.5
3033	3

FIGURA 6.23

El APL representaba el 40% de la carga de trabajo en el sistema de soporte a ventas, el cual tenía un porcentaje de utilización del 60%, por lo que se consideraba que a partir de septiembre de 1981 (fecha planeada de instalación de la 4341) se agregaría un 24% (considerando una 370/158) a la máquina de producción en el periodo 1. Por otro lado se indicó que el crecimiento del APL era del 7% anual.

En la figura 6.24 se presenta el crecimiento estimado en las cargas de trabajo del periodo 1. Se escogió el periodo 1 por ser el crítico en cuanto a tiempo de respuesta y utilización de CPU.

Los crecimientos especificados son los promedios de crecimiento encontrados en el análisis de los años de 79 y 80, excepto para el caso del TSO en el que se utilizaron los criterios expuestos por la gerencia DP en cuanto a demanda latente.

Los porcentajes de utilización se obtuvieron de la siguiente forma:

$$\% \text{ UBATCH} = \frac{\% \text{ U PERIODO 1} - \% \text{ UTSO (FIGURA 6.20 NOVIEMBRE 80)} \times .9}{100}$$

El .9 se debe a que como se dijo antes, el 10% de la carga en lote podría ser transferida a otros turnos.

% UTSO: (figura 6.20, noviembre 80)

% UAPL: Se obtuvo de la entrevista con la gerencia de DP que indicó que el APL ocupaba un 40% de toda la carga del sistema de ventas, el cual era utilizado en un 60%, por tanto:

$$\% \text{ UAPL} = \frac{60 \times 40}{100} = .20$$

La razón de que se hayan utilizado MIPS fue establecer una me-

PROYECCION DE LA CAPACIDAD EN MIPS REQUERIDA PARA EL PERIODO 1

	1980	1981	1982	1983
CAPACIDAD EN MIPS	.91			
CAPACIDAD UTILIZADA EN MIPS PARA BATCH (PERIODO 1) .91X%U BATCH X% DE CRECIMIENTO.	.3537	.4068	.4678	.5380
PORCENTAJE DE CRECIMIENTO		15%	15%	15%
CAPACIDAD UTILIZADA EN MIPS PARA TSO (PERIODO 1) .91X%U TSO X% DE CRECIMIENTO	.26208	.4193	.5660	.6793
PORCENTAJE DE CRECIMIENTO		60%	35%	20%
CAPACIDAD UTILIZADA EN MIPS PARA APL (PERIODO 1) .91X% U APL X% DE CRECIMIENTO		.2184	.2336	.2500
PORCENTAJE DE CRECIMIENTO		7%	7%	7%
CAPACIDAD UTILIZADA EN PERIODO 1 (TOTAL)	.6157	1.0445	1.2674	1.4673

FIGURA 6.24

dida común que permitiera comparar la capacidad utilizada para diferentes modelos de computadoras, ya que las horas de CPU y el porcentaje de utilización varían conforme a la velocidad del proceso de la CPU.

En las figuras 6.25 y 6.26 se graficó la cantidad de MIPS requerida para los años de 1981, 1982 y 1983, indicando también los límites de utilización recomendables para cada una de las alternativas de solución.

En este caso la evaluación de las alternativas se efectuó según los siguientes parámetros:

1) Precio/rendimiento
2) Capacidad de cómputo
3) Fecha de entrega
4) Posibilidad de crecimiento
5) Tecnología

Todos los sistemas propuestos utilizaban el mismo sistema operativo instalado, de modo que éste no fue un factor de decisión.

Las alternativas analizadas fueron:

A) Agregar al computador actual una facilidad que permite tener un segundo procesados conectado a la 370/158 (ATTACH PROCESSOR). Esta solución incrementa la capacidad de cómputo de la 370/158 de .91 a 1.4 MIPS, además de que tiene la ventaja de mejorar la disponibilicad, ya que si falla una CPU la otra sigue funcionado.

Otra posible ventaja de esta alternativa es que esta facilidad puede instalarse directamente en el sitio en que se encuentra el computador y que el proveedor puede entregarla inmediatamente.

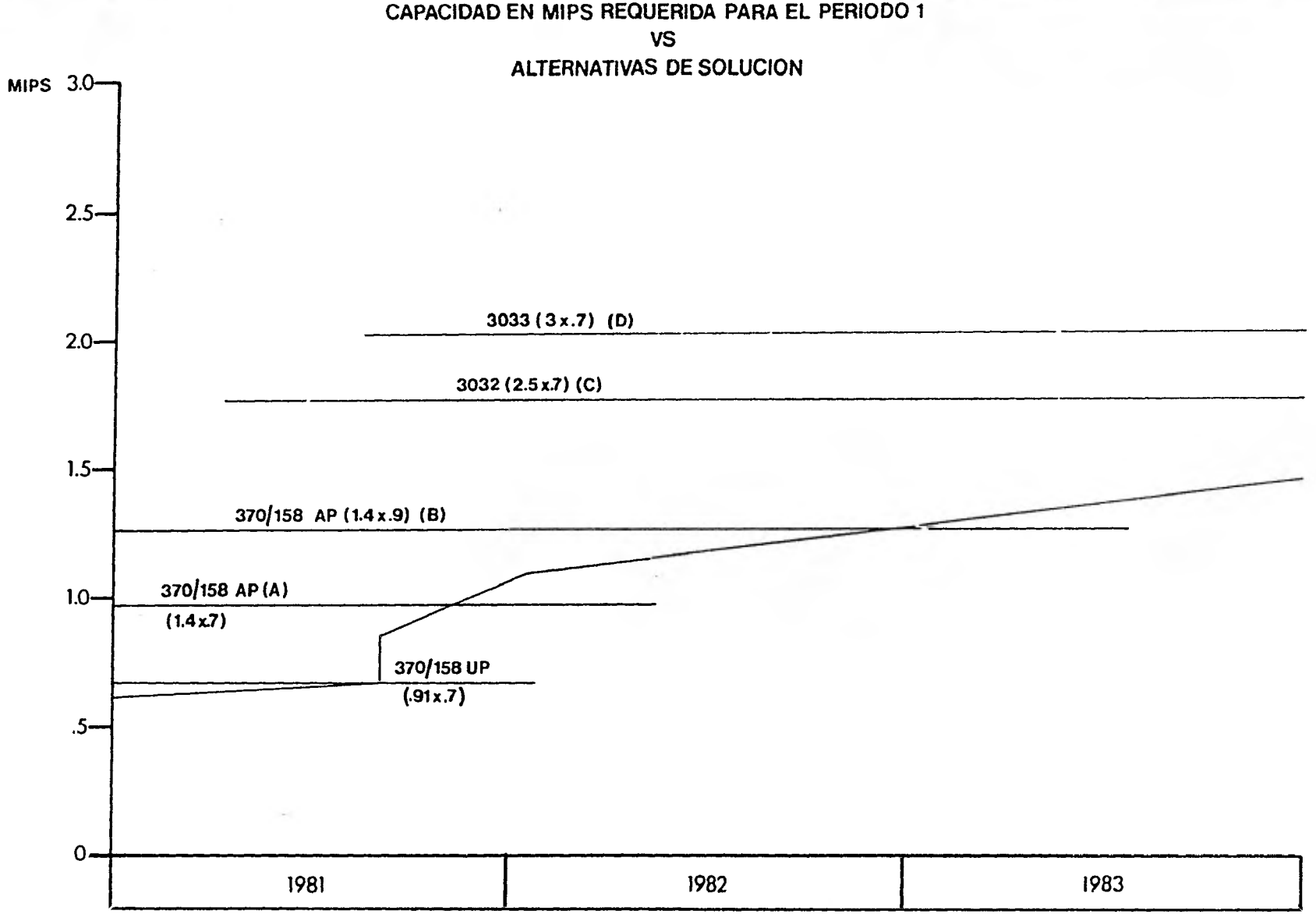

CAPACIDAD EN MIPS REQUERIDA PARA EL PERIODO 1
VS
ALTERNATIVAS DE SOLUCION
MIPS
3.0
2.5
2.0
1.5
1.0
.5
0
3033 (3 x .7) (D)
3032 (2.5 x.7) (C)
370/158 AP (1.4 x .9) (B)
370/158 AP (A)
(1.4 x.7)
370/158 UP
(.91 x .7)
1981
1982
1983

Figura 6.25

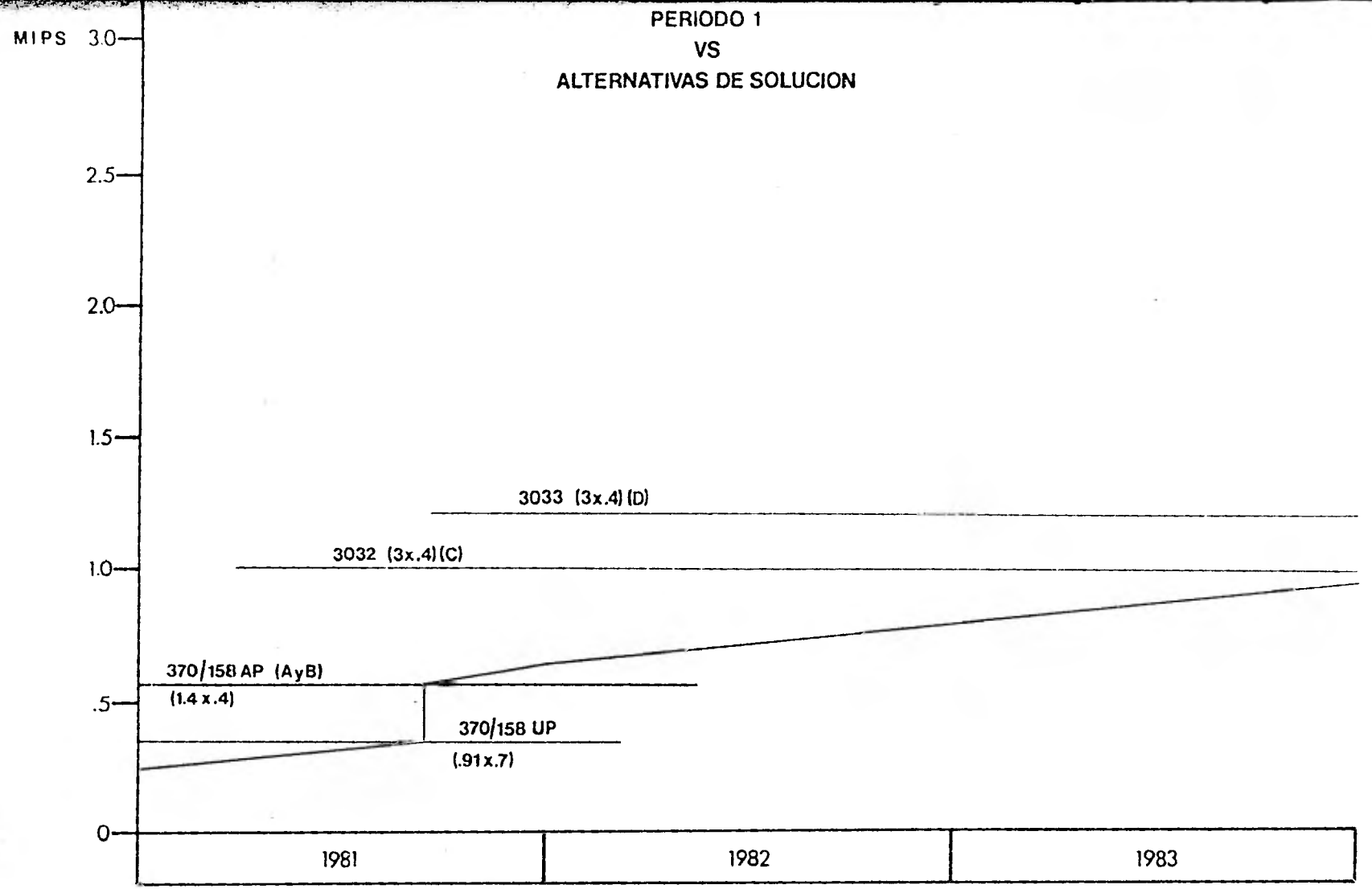

PERIODO 1
VS
ALTERNATIVAS DE SOLUCION
MIPS
3.0
2.5
2.0
1.5
1.0
.5
0
3033 (3x.4) (D)
3032 (3x.4) (C)
370/158 AP (AyB)
(1.4 x .4)
370/158 UP
(.91x.7)
1981
1982
1983

Figura 6.26

Después del análisis del crecimiento en relación con la capa-- cidad máxima recomendada (fig. 6.25 y 6.26), no se recomendó esta - alternativa, a pesar de sus ventajas porque:

1) La fecha de saturación estaba muy cercana (septiembre u octubre de 1981)
2) Su capacidad de cómputo no podría seguir creciendo.
3) Había computadores con tecnología más moderna.

B) Otra alternativa analizada fue la de incrementar el poder de cómputo de acuerdo con el inciso A) y cambiar las políticas de - servicio para el turno 1 dando prioridad a los sistemas en línea sobre los procesos en lote. Por un lado esto elevaría la capacidad - máxima recomendada de 70% a 90% y por el otro, incrementaría el tiempor de respuesta para los procesos en lote que tuvieran que correren el periodo 1.

En el análisis de la figura 6.27 se observa que la limitante - en esta alternativa es la capacidad requerida para poder mantener - el tiempo de respuesta de las aplicaciones en línea en los periodos pico. Situación que nos lleva a tener una fecha de saturación a finales de 1981 lo que implica descartar esta alternativa.

C) En esta alternativa se analizaba la posibilidad de contar con un sistema 3032 con las siguientes ventajas sobre las alternativas anteriores.

1) Una tecnología más moderna que la 370/158.
2) Capacidad de cómputo suficiente para cubrir los requeri

mientos previstos hasta 1984, lo cual justificaba la inversión.

3) Fecha de entrega bastante atractiva, ya que el proveedor se comprometía a entregarlo en 3 meses, lo que permitiría estabilizar el servicio antes de que la 370/158 llegara a la fecha de saturación.

Esta alternativa podría haber sido la solución, pero tenía las siguientes desventajas:

1) El proveedor ya había anunciado que este modelo dejaría de fabricarse.

2) La literatura especializada hablaba de problemas de -- disponibilidad en el director de canales de este computador.

3) Como ya no se produciría, tenía muy poco poder de crecimiento.

D) Esta fue la alternativa sugerida ya que el modelo 3033 analizado tenía las siguientes ventajas.

1) Era el último computador anunciado por el proveedor, - lo cual aseguraba el soporte adecuado por lo menos hasta la fecha de saturación.

2) Un amplio poder de crecimiento tanto en memoria como - en poder de cómputo, lo cual permitiría satisfacer los requerimientos de los próximos 5 años.

3) Tecnología reciente.

Esta solución planteaba el problema de la fecha de entrega, ya

que ésta coincidía con la fecha prevista de saturación de la 370/158; esto definitivamente era un riesgo, pero la gerencia consideró conveniente correrlo.

R E F E R E N C I A S

REFERENCIAS

Chandra 01 — S. Chandrasekhar, Observation must be confirmed by a theory, Univ. of Chicago Magazine, Chicago, Ill., Summer 1974, p. 16.

Rosen 01 — Saul Rosen, Lectures on the measurement and evaluation of the performance of computing systems, Purdue University, p. 25

Schulman 01 — Hardware measurement device for IBM System 360. Time sharing evaluation, Proc. 22nd ACM National Conference 1967, pp. 103-109.

Morris 01 — J. A. Morris, Hardware measurement. Past, present - and future, paper presented to SHARE, vol. 2, pp. - 308-332

Buchholz 01 — W. Buchholz, A Synthetic job for measuring system -- performance, IBM System Journal 9 (1969), pp. 309-318.

Cooper 01 — J. C. Cooper, A Capacity planning methodology, IBM - System Journal 19 (1980), pp. 32-34.

Sreen 01 — K. Sreenivasam and A. J. Kleinaman, On the construction of a representative synthetic workload, Comm. - ACM 17 (1974), pp. 127-133.

Saltzer 01 — J. H. Saltzer and J. W. Gintell, The instrumentation of multics, Comm. ACM 13 (1970), pp. 493-500.

TPNS 01 — Teleprocessing network simulator (TPNS), Release 3.0 General Information Manual, order no. GH20-1907.

Ferrari 01 — Ferrari, D., Workload characterization and selection on computer performance measurement, Computer, julio/ agosto, 1972.

Teichro 01 — C. Teichroew and J. F. Lubin, Computer simulation discussion of the techique and comparison of languages, comm. ACM, 9 (1966), pp. 723-741.

Stewart 01 — H. M. Stewart, Performance analysis of complex communication systems, IBM System Journal 18, no. 3, pp.- 356-373 (1979)

Nguyen 01 — H. C. Nguyen, A. Ockene, The role of detailed simulation in capacity planning, IBM System Journal 19,1, pp. 81-101.

Nonotza 01 — Revista de la difusión científica, tecnólogica y cultural editada por IBM de México, Libro décimoprimero, año V, Segundo trimestre de 1981, p. 475-476.

Heller 01 — H. Hellerman and T. F. Conrroy, Computer system performance, McGraw-Hill Computer Sciences Series, pp.-169.

Sackman 01 — Time-sharing vs. batch processing: the experimental - evidence, vol. 32, 1968, pp. 1-10.

UNAM 01 — Informe 1977, vol II, UNAM.

Cheng 01 — P. S. Cheng, Trace-driven system modeling, IBM System Journal, 8 (1969), pp. 280-289.

Joslin 01 — O. E. Joslin, Aplication bench-marks: the key to meaningful computer evaluation, Proc. 20th ACM National Conference, 1965.

Huesmann 01 — L. R. Huesmann and R. P. Goldberg, Evaluating computer systems through simulation, Computer Journal 10 (1967), pp. 150-156

Kendall 01 — D. G. Kendall, Stochastic processes occurring in the theory of queues and the analysis by the method of the imbedded mark or chain, Annals of Mathematical Statistics, vol 24, 1953, pp. 338-350.

Bell 01 — T. E. Bell, Objetives and problems in simulating computers, AFIPS, Conf. Proc., 41 (1272)

Grenander 01 — U. Grenander and R. F. Tsao, Quantitative methods for evaluating computer system performance. A review and proposals, statistical computer performance evaluation, Academic Press, New York, 1972.

Nielsen 01 — N. R. Nielsen, The analysis purpose computer time-sharing systems, Document 40-10-1, Stanford Computation Center, Stanford, Calif., 1966.

A N E X O S

ABRIL 1981

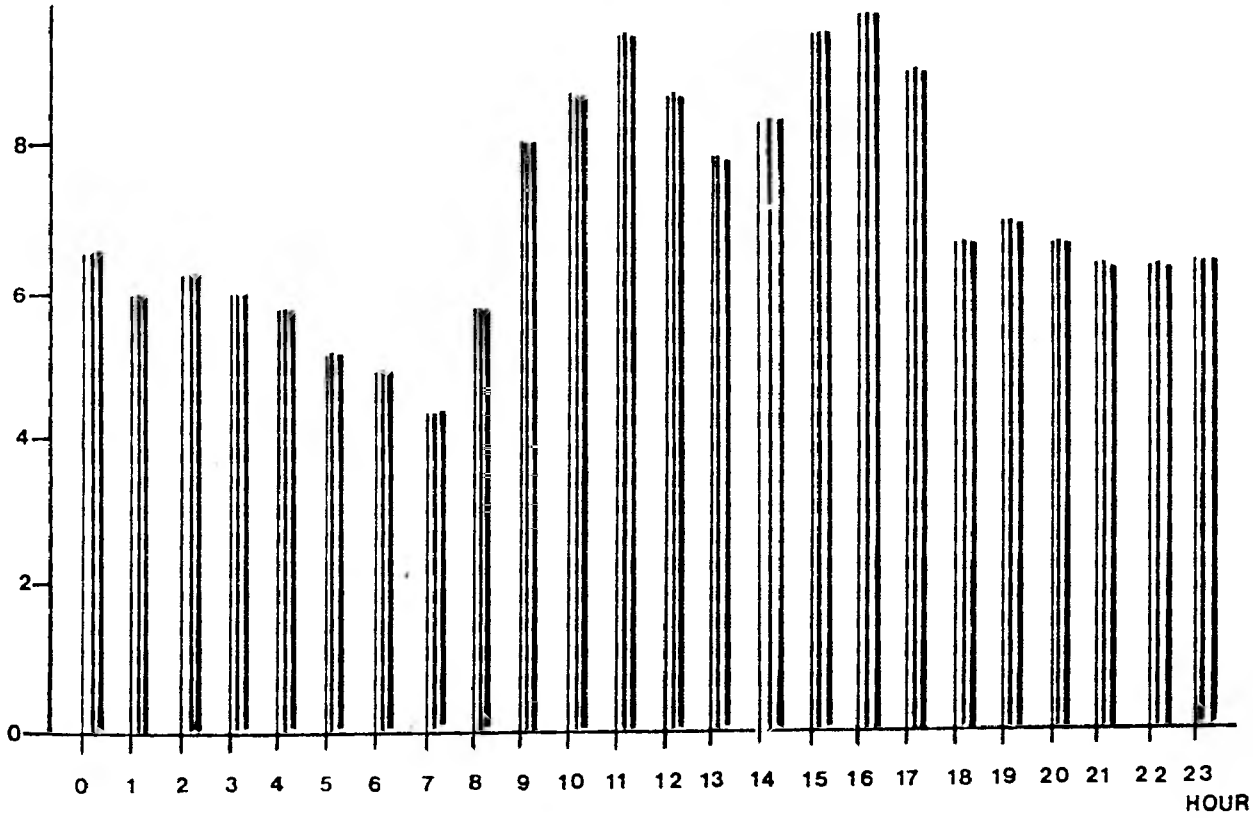
REPORTES DE UTILIZACION DE CPU

NIVEL DE MULTIPROGRAMACION – PROMEDIO DIARIO

PAGE : 0001
DATE : 81 MAY 20
TIME : 12 22 51

YEAR : 81
MONTH : APR
DAY : TOT
HOUR : TOT
SYSID : H158

MULTI PROGRAMMING LEVEL

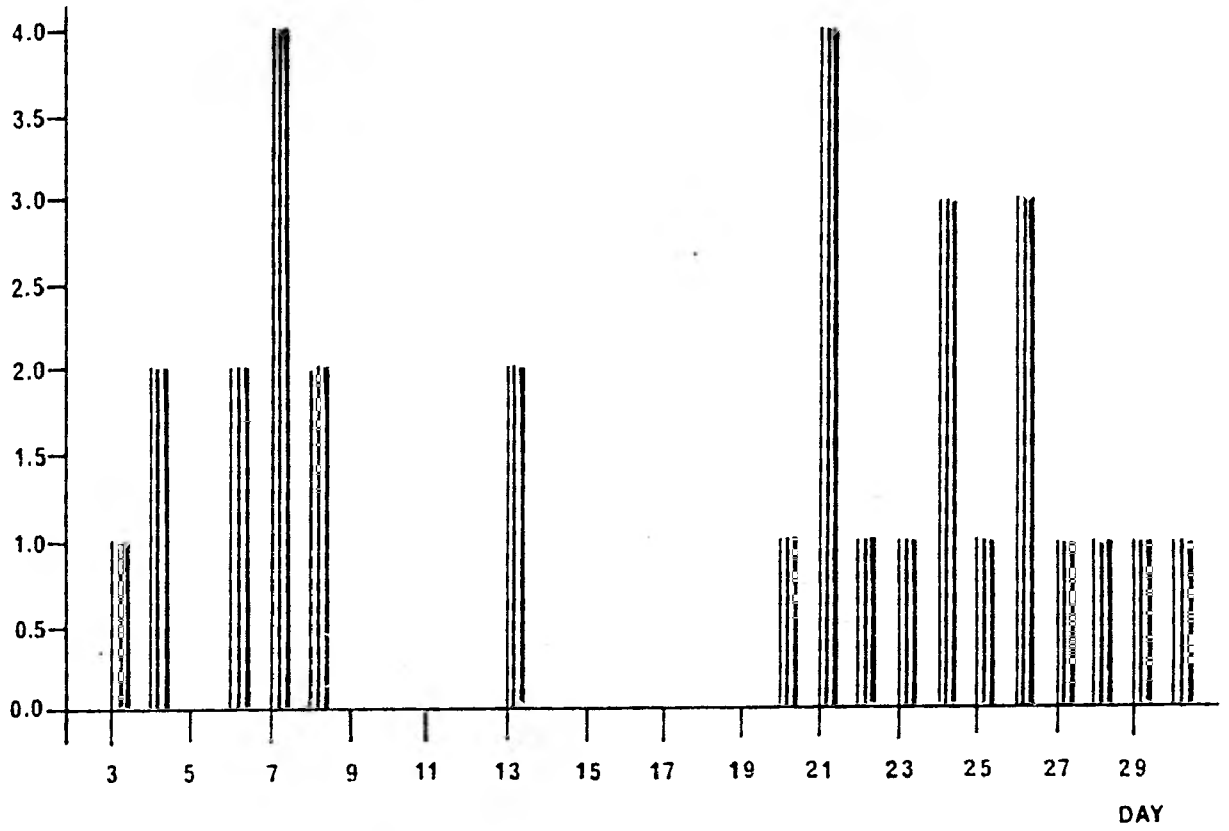


ABRIL 1981
REPORTES DE IPL
RESUMEN DE IPL'S

PAGE :0001
DATE :81 MAY 20
TIME :12:23:29

YEAR :81
MONTH :APR
SYSID :H158
PERIODO :TOTAL

NUMBER OF IPL'S.



ABRIL 1981

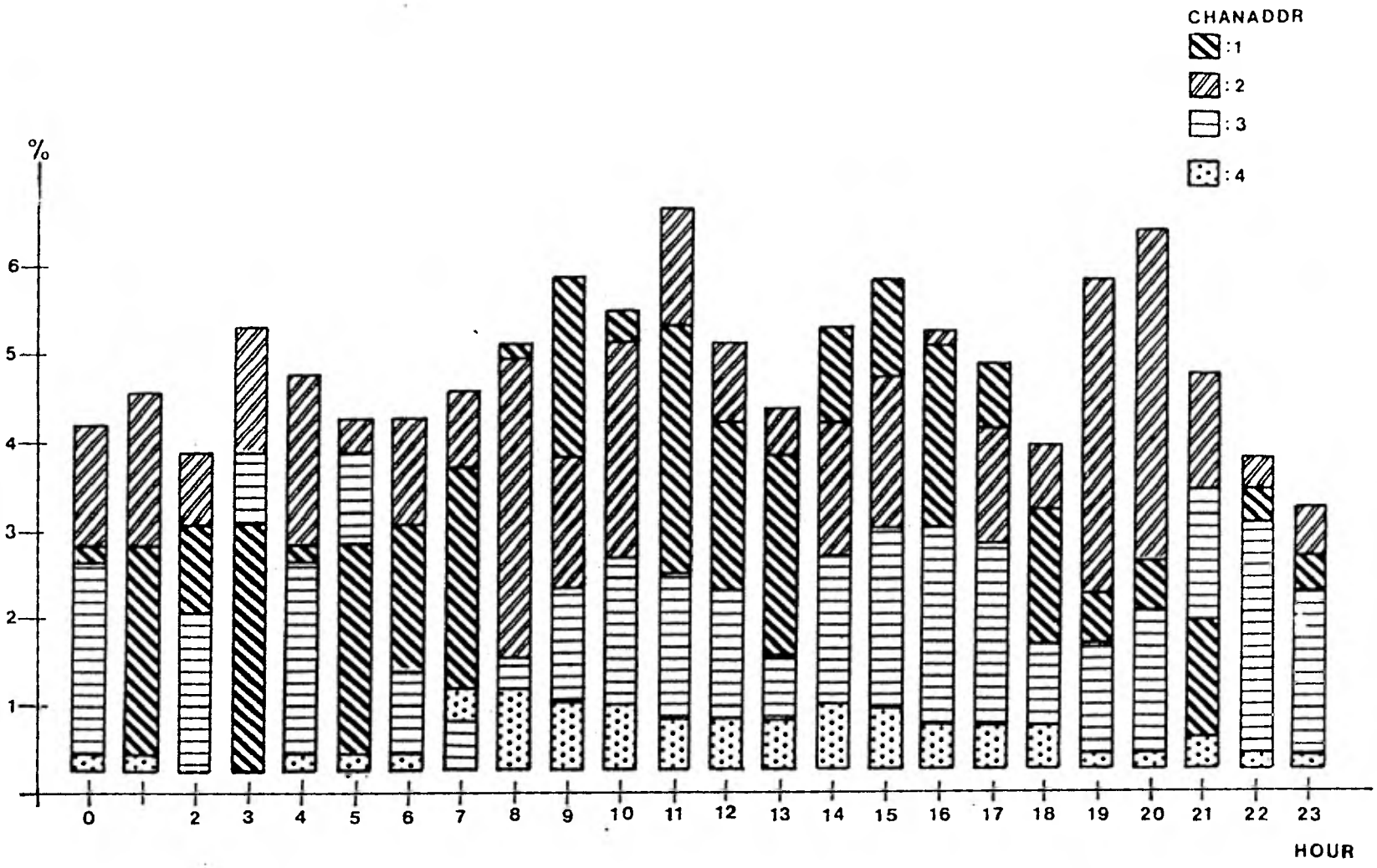
UTILIZACION DE CANALES

CARGA DE CANALES CUANDO C P U EN WAIT— CADA DIA DEL MES

DATE : 81 MAY 20
TIME : 12:23:21

YEAR : 81
MONTH : APR
DAY : TOT
HOUR : TOT
SYSID : H158
CPUNUM : 0

AVG LOAD WHEN C P U IN WAIT



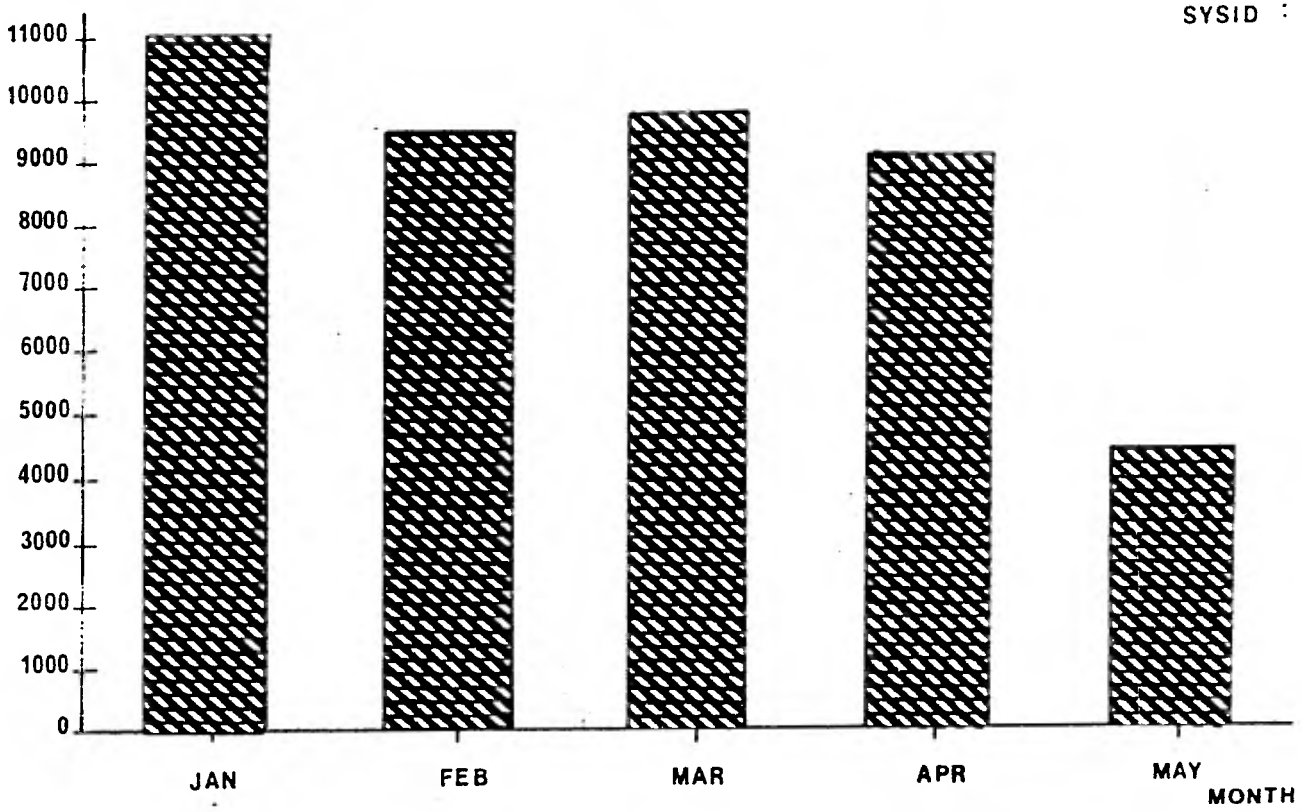
REPORTE DE JOBS

TODOS LOS JOBS-CARGA ANUAL

PAGE : 0001
DATE : 81 MAY 20
TIME : 12:20:19

YEAR : 81
MONTH : TOT
SYSID : H158

TOTAL NUMBER OF JOBS



ABRIL 1981

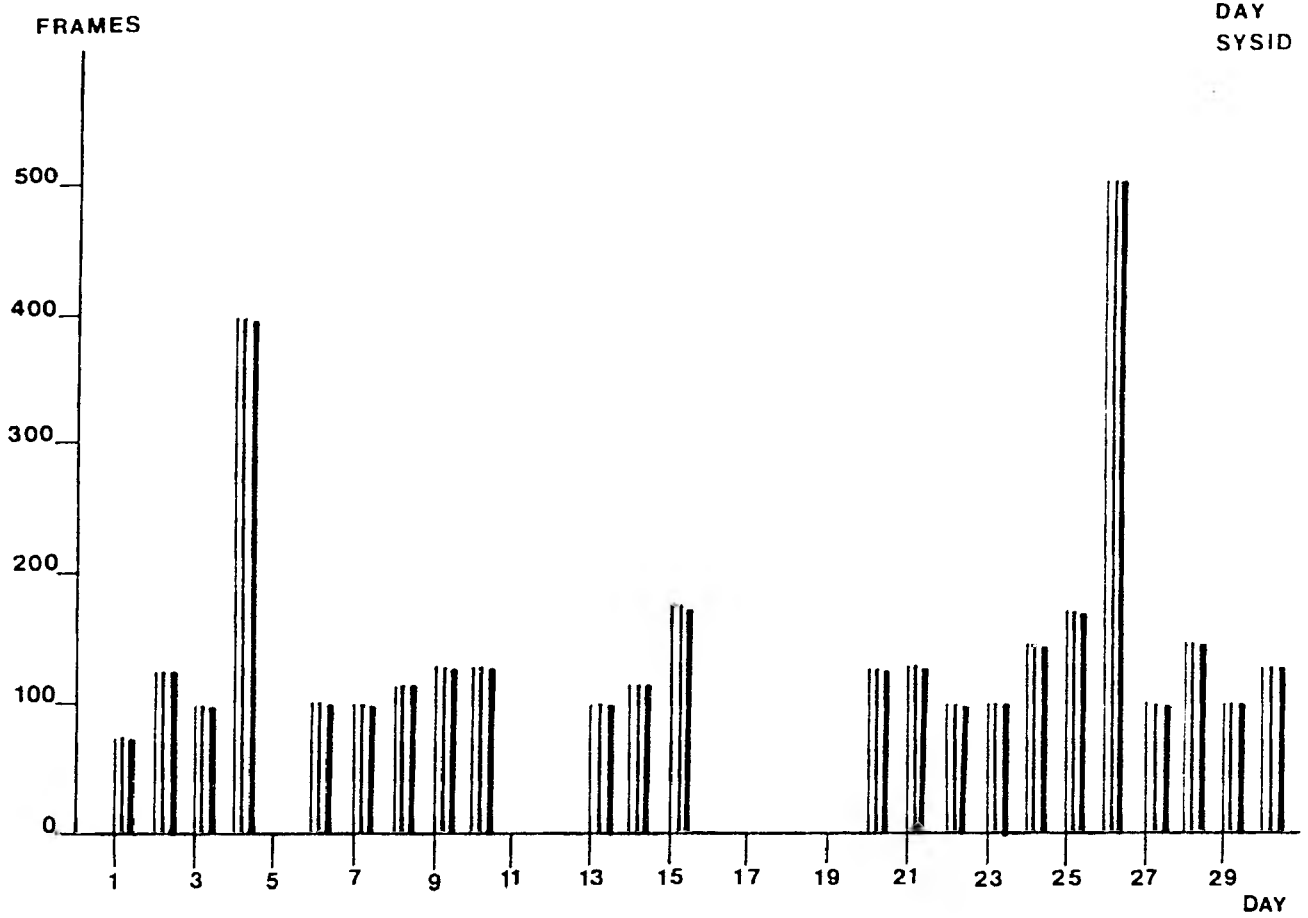
REPORTES DE PAGINACION

PAGINAS DISPONIBLES - PROMEDIO POR DIA

PAGE : 0001
DATE : 81 MAY 20
TIME : 12:24:04

AVERAGE NUMBER UNUSED FRAMES

YEAR : 81
MONTH : APR
DAY : TOT
SYSID : H 158

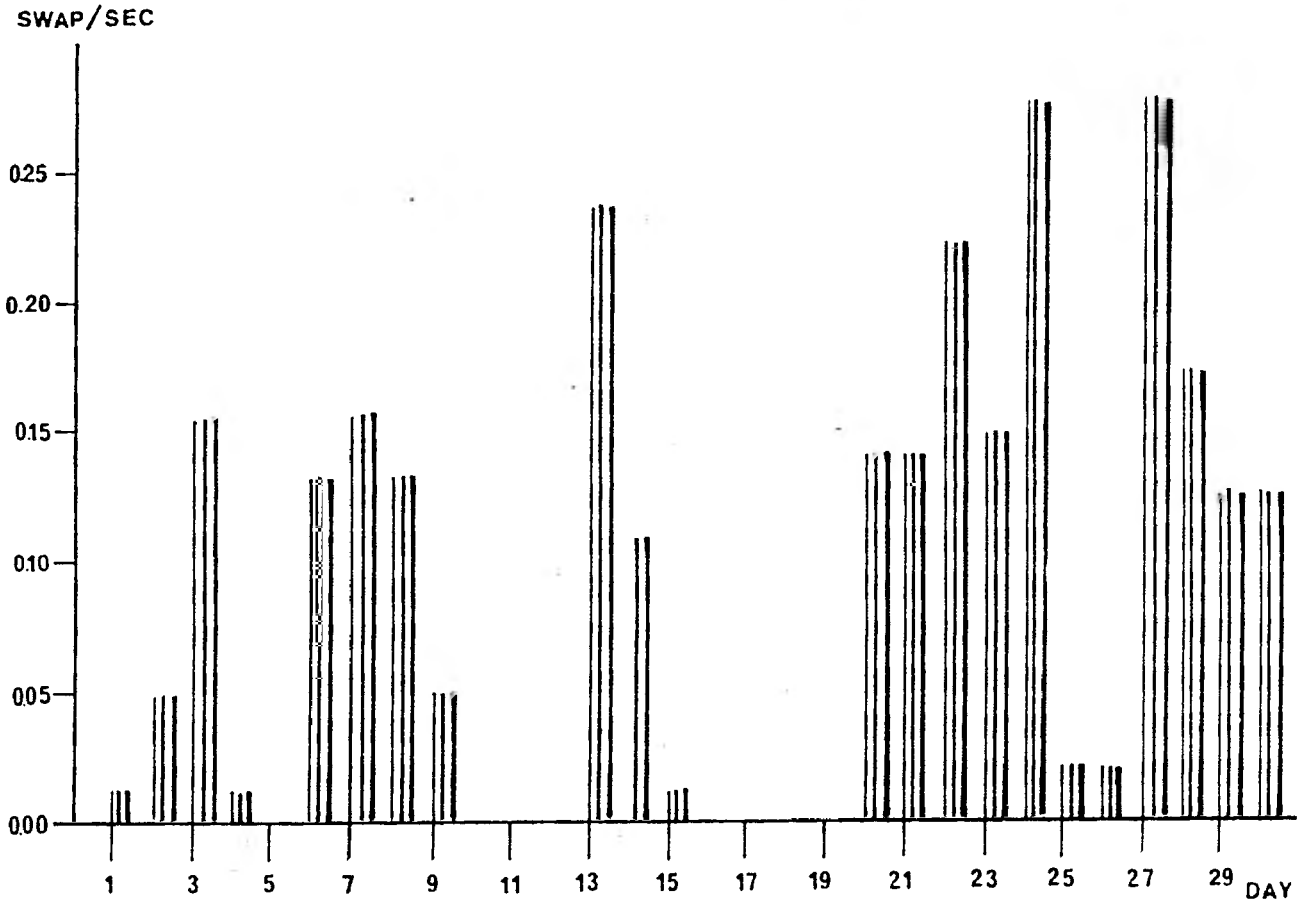


ABRIL 1981
REPORTES DE PAGINACION
SWAP RATE - PROMEDIO POR DIA

YEAR: 81
MONTH: APR
DAY: TOT
SYSID: H 158

PAGE : 0001
DATE : 81 MAY 20
TIME : 12:24:00

SWAP SEQUENCE RATE



Impresiones
artes al instante, s.a. de c.v.
REP. DE COLOMBIA No. 6, 1er. PISO
(CASI ESQ. CON BRASIL)
MEXICO 1, D. F.
526-04-72 529-11-19