



Capítulo 3: Manage Work

Componentes

- Los programas (modulos) se agrupan en componentes, los cuales dan diferentes servicios
- **Servicios transparentes para el usuario**
 - Ejecución de instrucciones
 - Manejo de memoria
 - Multiprogramacion/Multiprocesamiento
- **Servicios disponibles para el usuario**
 - Manejo de archivos
 - Lenguajes de programacion
 - Seguridad
- El z/OS usa bloques de control para describir toda la informacion sobre el sistema, los recursos y las tareas que maneja

Componentes

- Un grupo de instrucciones relacionadas se denomina rutina o módulo.
- Un conjunto de módulos relacionados para hacer posible algo particular se llama componente del sistema.
- Las secuencias de instrucciones que realizan funciones del sistema frecuentemente utilizadas, pueden ser invocadas con instrucciones macro ejecutables o macros.
 - Por ejemplo las macros de z/OS existen para funciones como abrir y cerrar archivos de datos, cargar y eliminar programas y enviar mensajes a consola.
 - La PSW (palabra de estado del programa), es un área de datos de 64 bits en el que el procesador, junto con registros de control (control registers), registros de temporización (control registers) y registros de prefijo (prefix registers), proporciona detalles cruciales tanto para el hardware y el software.

-La PSW actual incluye la dirección de la próxima instrucción del programa a ser ejecutada e información de control del programa que se está ejecutando.

Cada procesador tiene solamente una PSW actual.

-Por lo tanto, sólo se puede ejecutar una tarea sobre un procesador a la vez.

-z/OS es capaz de multiprogramar o ejecutar muchos programas simultáneamente, de distintos usuarios.

-En multiprogramación, cuando un trabajo no puede utilizar el procesador, el sistema puede suspender o interrumpir el trabajo, liberando al procesador para atender otro trabajo (job).

-z/OS también puede realizar multiprocesamiento, que es la operación simultánea de dos o más procesadores que comparten los distintos recursos de hardware, como memoria y dispositivos de almacenamiento en disco.

-A medida que las instrucciones ejecutan el trabajo, guardan el seguimiento (tracking) de este trabajo en áreas de almacenamiento conocidas como bloques de control.

-En términos generales, existen tres tipos de bloques de control de z/OS:

- Bloques de control relacionados con el sistema (System-related control blocks).
- Bloques de control relacionados con recursos (Resource-related control blocks)
- Bloques de control relacionados con tareas (Task-related control blocks).

-Cada bloque de control relacionado con el sistema (system control blocks) representa un sistema z/OS y contiene información de todo el sistema, tal como cuántos procesadores están en uso.

-Cada bloque de control relacionado con los recursos (resource control blocks), representa un recurso, como un procesador o un dispositivo de almacenamiento.

-Cada bloque de control relacionado con la tarea (task control block) representa a una unidad de trabajo.

Los bloques de control sirven como vehículos para la comunicación en todo z/OS.

Esta comunicación es posible porque los programas que lo utilizan conocen la estructura de un bloque de control y, por lo tanto, estos programas pueden encontrar la información necesaria sobre la unidad de trabajo o recurso.

Los bloques de control que representan muchas unidades del mismo tipo se puede encadenar en colas, con cada bloque de control apuntando al siguiente en la cadena. El sistema operativo puede buscar en la cola información sobre una unidad de trabajo o recurso en particular, que podría ser:

*Una dirección de un bloque de control o una rutina requerida.

*Datos actuales, como un valor, una cantidad, un parámetro o un nombre.

Indicadores de estado (normalmente bits únicos en un byte, donde cada bit tiene un significado específico).

*z/OS utiliza una gran variedad de bloques de control, muchos de ellos con fines muy especializados.

Tres de los bloques de control más utilizados:

- *Bloque de control de tareas (TCB), que representa una unidad de trabajo o tarea.
- *Bloque de solicitud de servicio (SRB), que representa una solicitud de un servicio del sistema
- *Bloque de control del espacio de direcciones (ASCB), que representa un espacio de direcciones (address space).

-Conceptualmente, las mainframes y todas las demás computadoras tienen dos tipos de almacenamiento físico.

-El almacenamiento físico ubicado en el propio procesador del mainframe. Se llama almacenamiento en procesador o almacenamiento real (almacenamiento principal).

-El almacenamiento físico externo al mainframe, incluido el almacenamiento en dispositivos de acceso directo, como unidades de disco y de cinta. Este almacenamiento se denomina almacenamiento auxiliar.

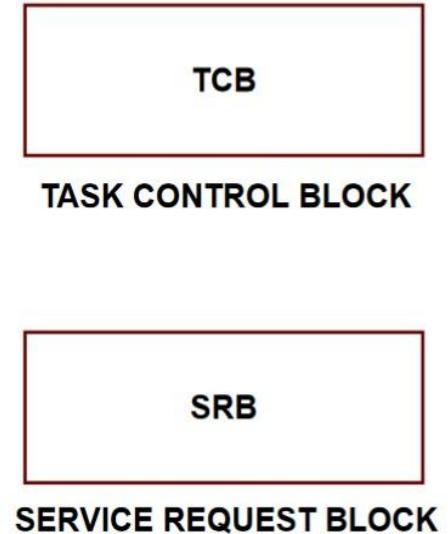
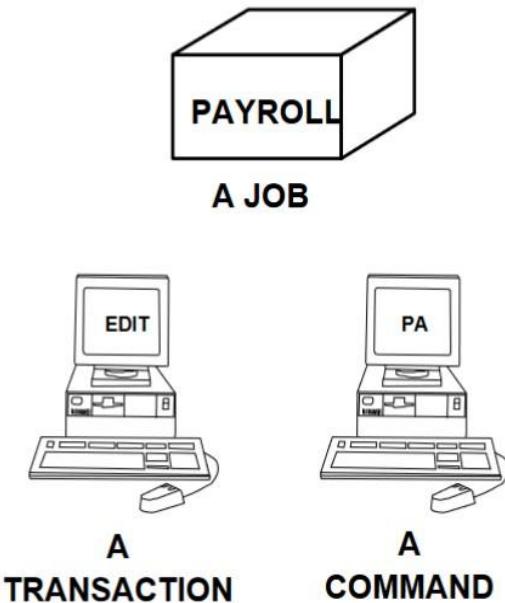
Módulos y macros

- z/OS se compone de instrucciones de programación que controlan el funcionamiento del sistema.
- Estas instrucciones garantizan que el hardware de la computadora esté siendo utilizada de manera eficiente y permite que se ejecuten los programas de aplicación.
- z/OS incluye conjuntos de instrucciones (set) que, por ejemplo, aceptan trabajo, convierten el trabajo a una forma que la computadora puede reconocer, realizar un seguimiento del trabajo, alocar recursos para el trabajo, ejecutar, trabajar, monitorear el trabajo y manejar la producción.
- Un grupo de instrucciones relacionadas se denomina rutina o módulo.
- Un conjunto de módulos relacionados que hacen posible una función determinada del sistema, se llama componente del sistema.
- El manejo de la carga de trabajo de z/OS (WLM), por ejemplo, controla los recursos del sistema, mientras que el recovery termination manager (RTM) maneja la recuperación del sistema.

Multiprocesamiento

- El funcionamiento simultáneo de dos o más procesadores que comparten los distintos recursos de hardware.
- Las secuencias de instrucciones que realizan funciones del sistema utilizadas con frecuencia se pueden invocar con instrucciones macro ejecutables o macros.
- Las macros de z/OS existen para funciones tales como abrir y cerrar archivos de datos, cargar y eliminar programas y enviar mensajes al operador de la computadora.

A Task

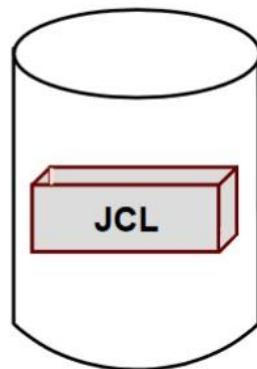
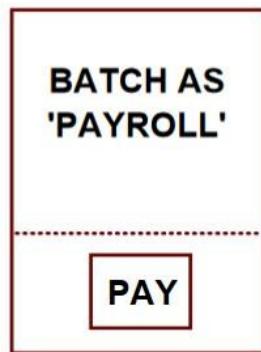


WE THINK

z/OS
THINKS

A Job

```
//PAYROLL JOB CLASS=A, PRTY=8  
//STEP1 EXEC PGM=PAY  
//DS1 DD DSN=PAY.D1, DISP=SHR
```

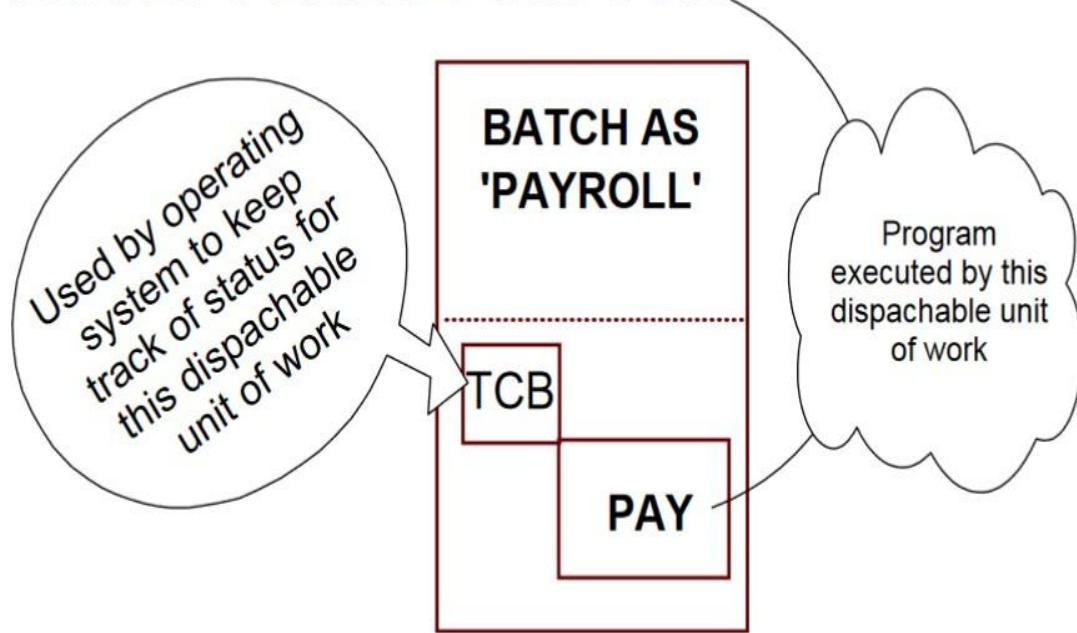


WE THINK

z/OS
THINKS

A Job Step

/STEP1 EXEC PGM=PAY



Services of the Supervisor - Program Linkage



Program Linkage

Dispatching

Interrupt Handling

Supervisor Calls

Inter-address Space Communication

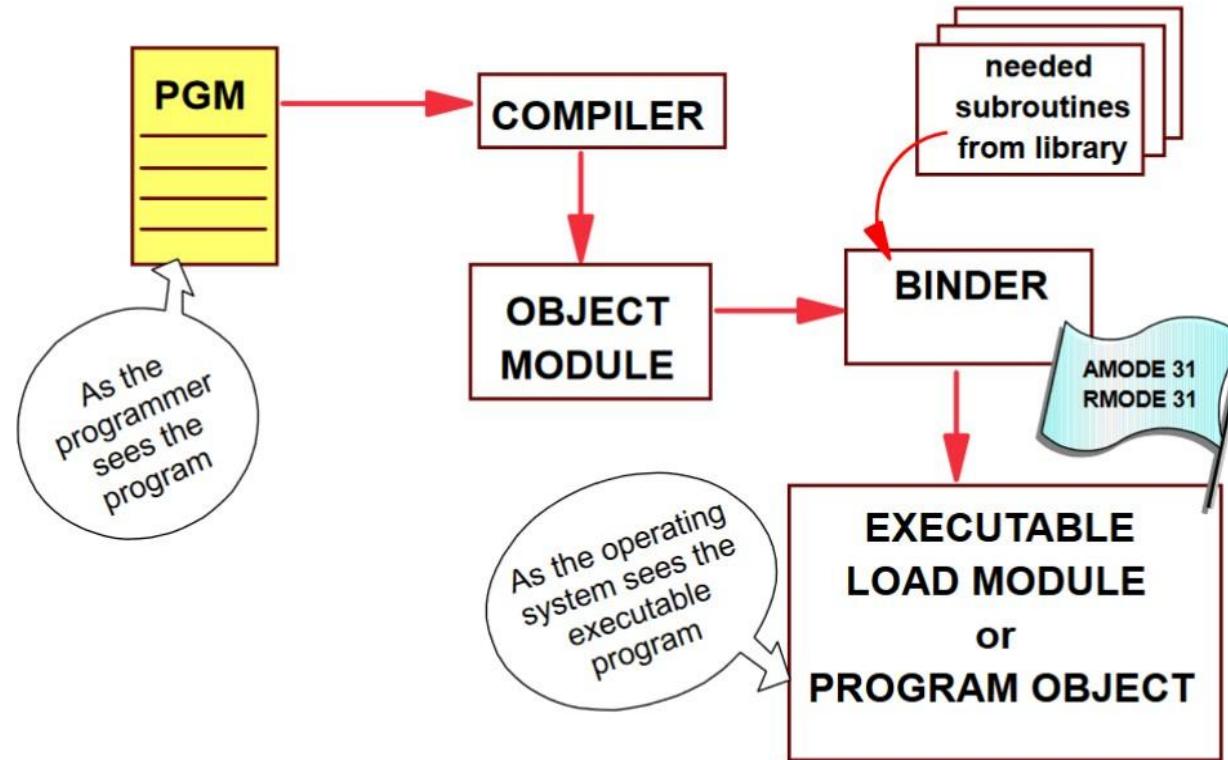
Serialization

UNIX System Services

Resource Balancing

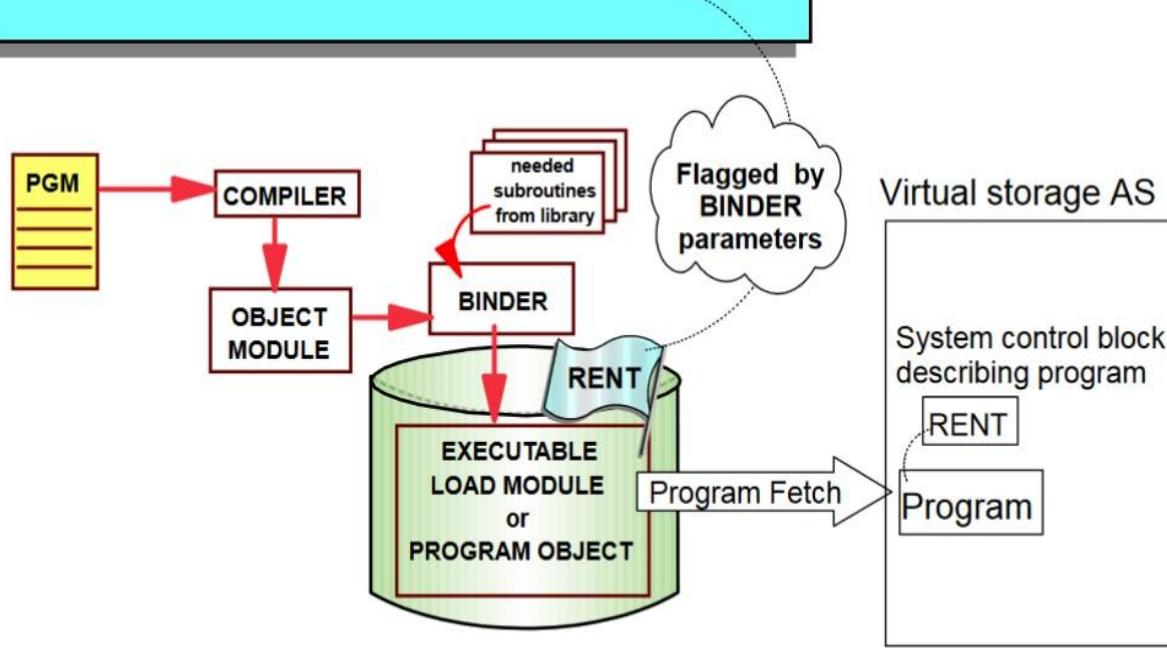
Measurement and Reporting

Programs, Load Modules and Program Objects

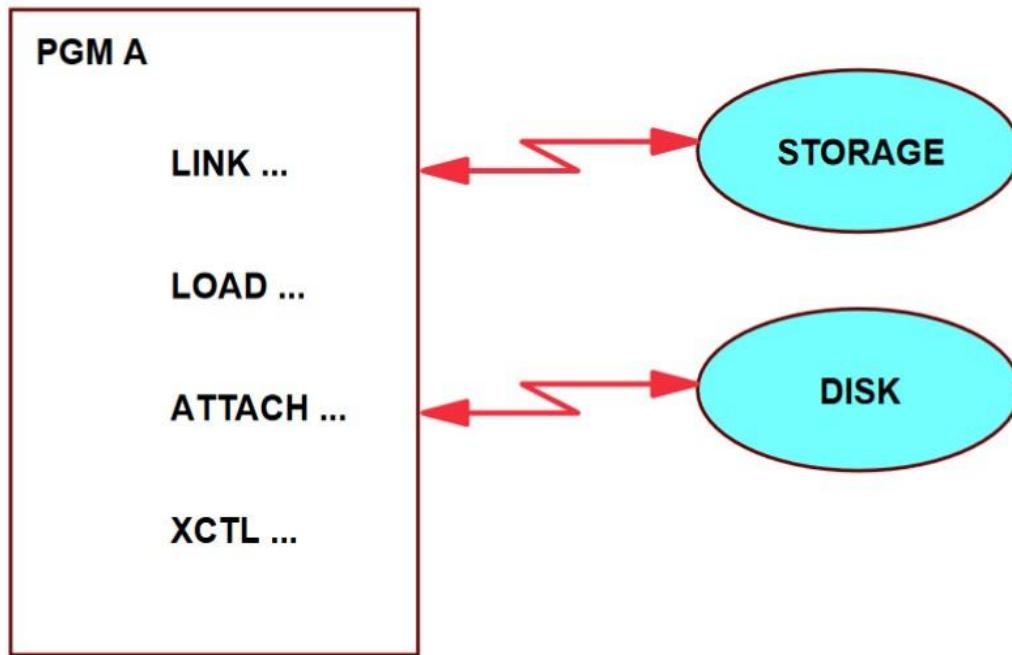


Module Attributes

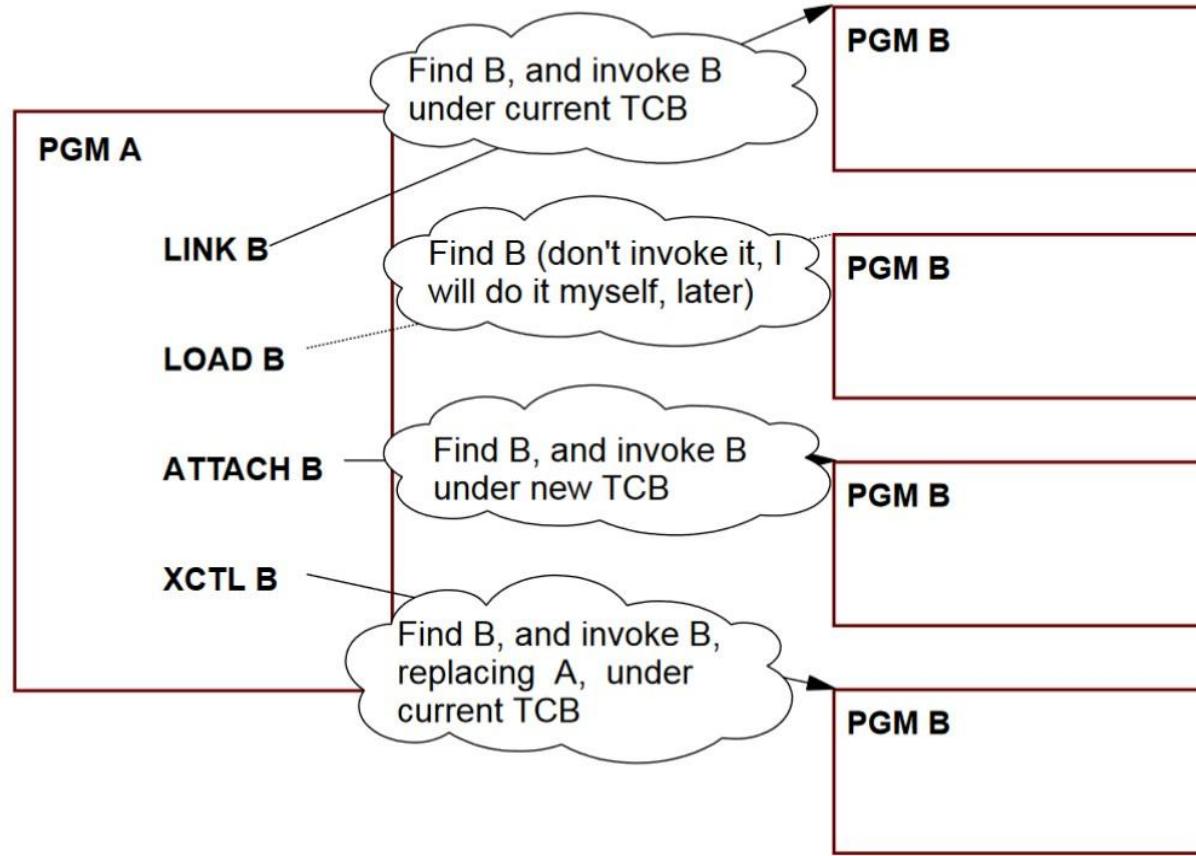
```
/LKED EXEC PGM=IEWL,  
PARM='REUS=(NONE|RENT|REUS|SERIAL),...'
```



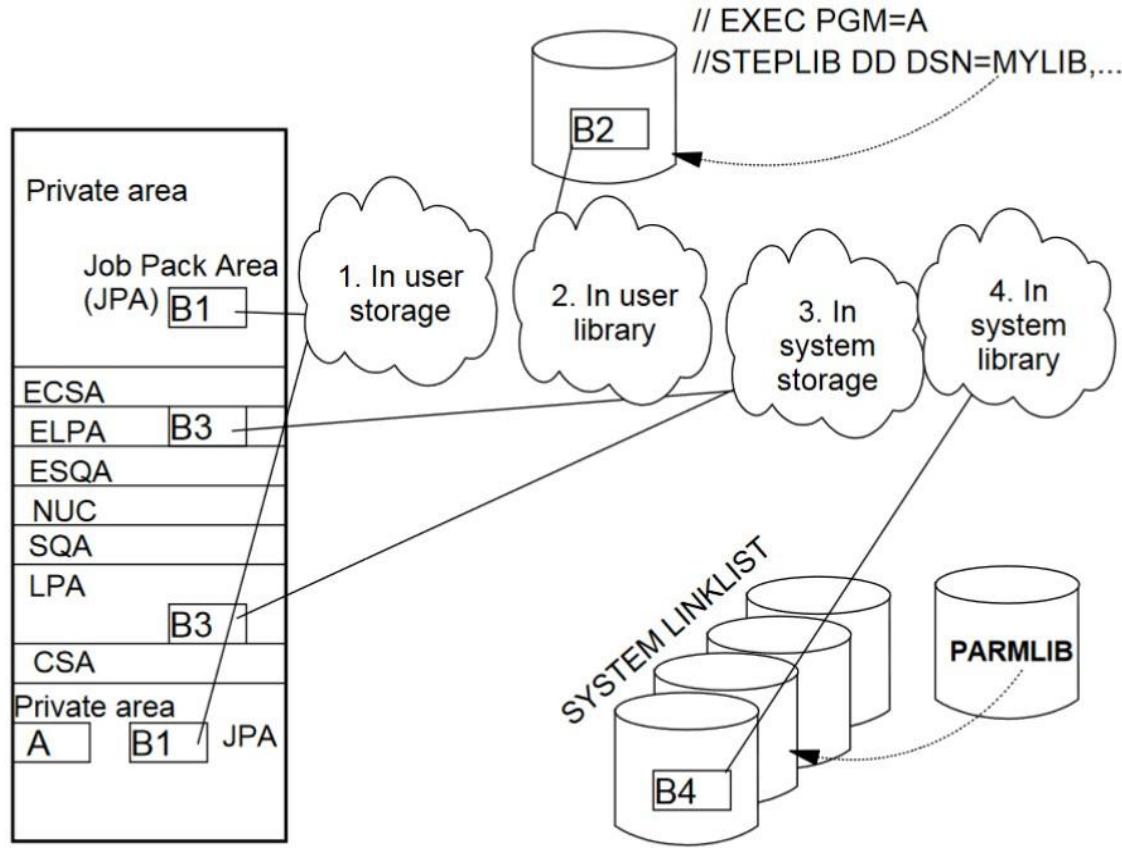
Dynamic Linkage



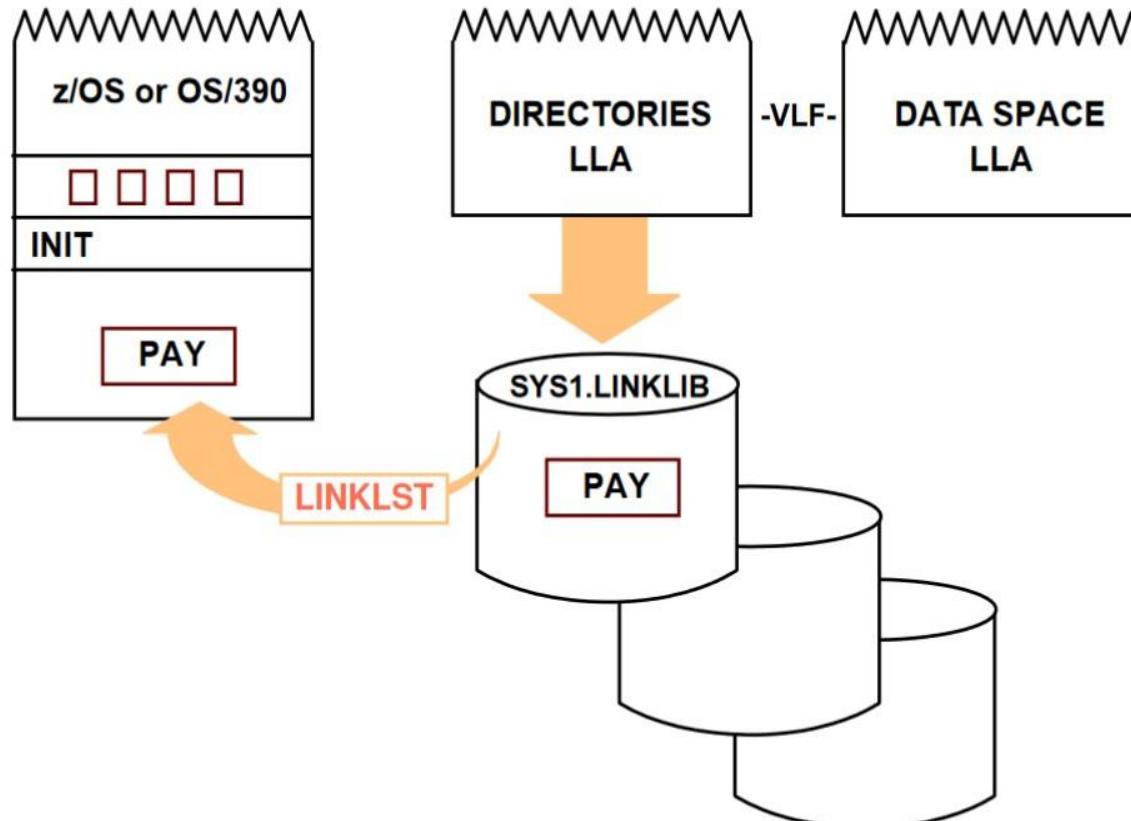
Dynamic Linkage - Overview



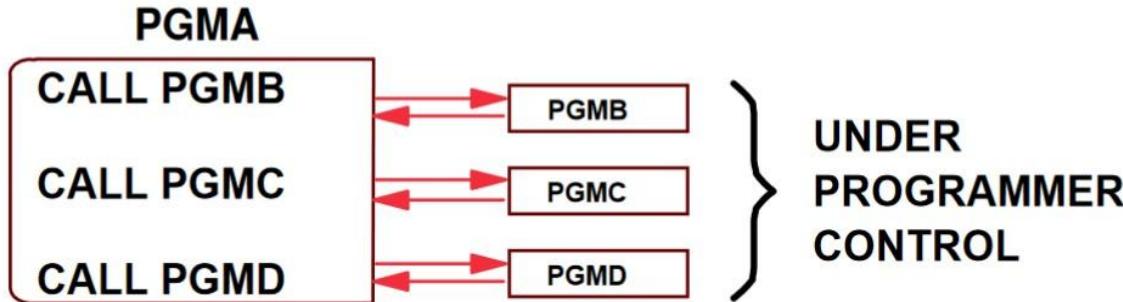
Finding a program



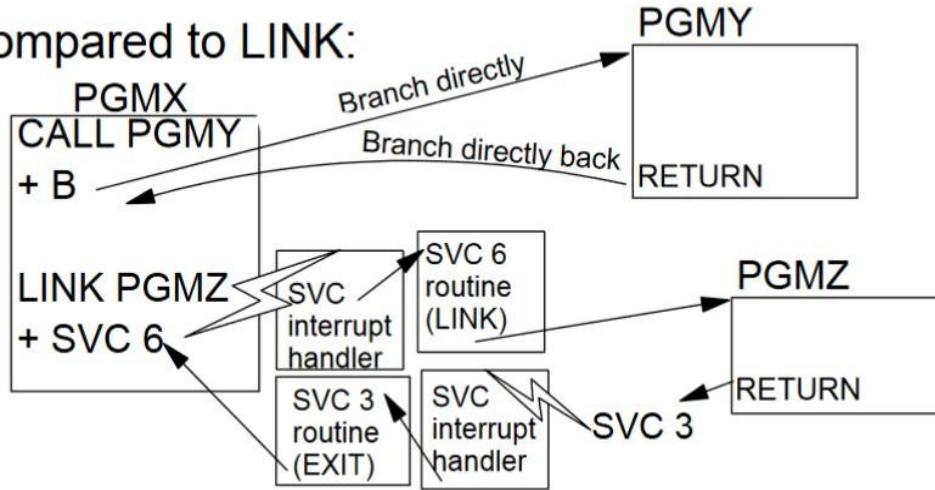
What is LINKLIST?



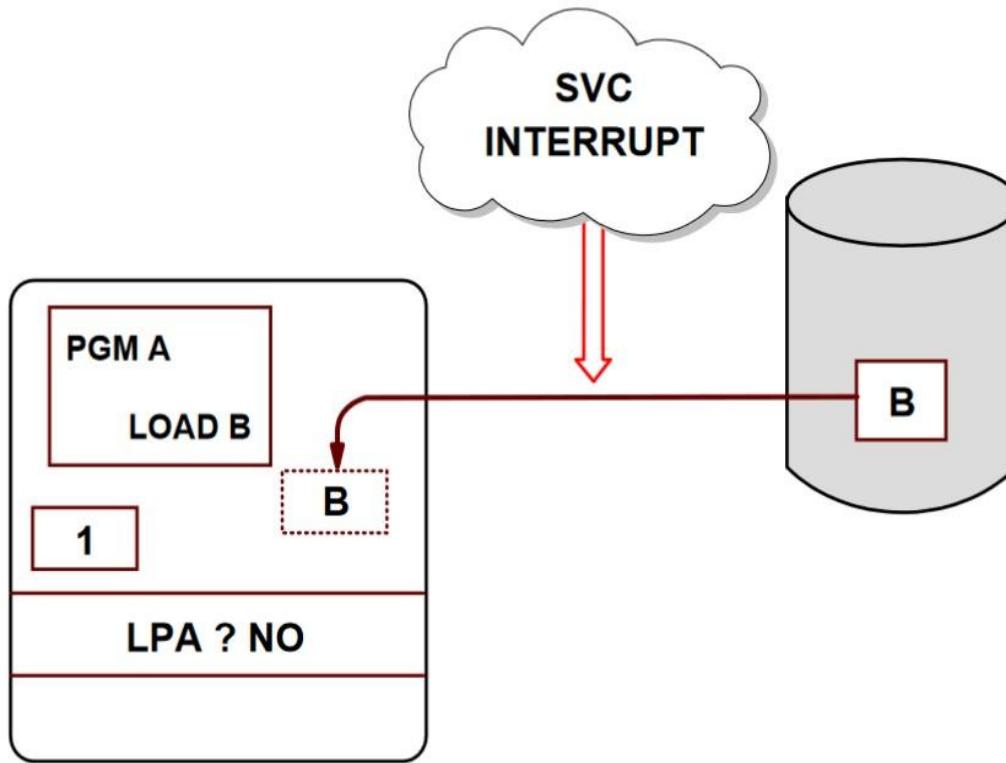
Call Macro



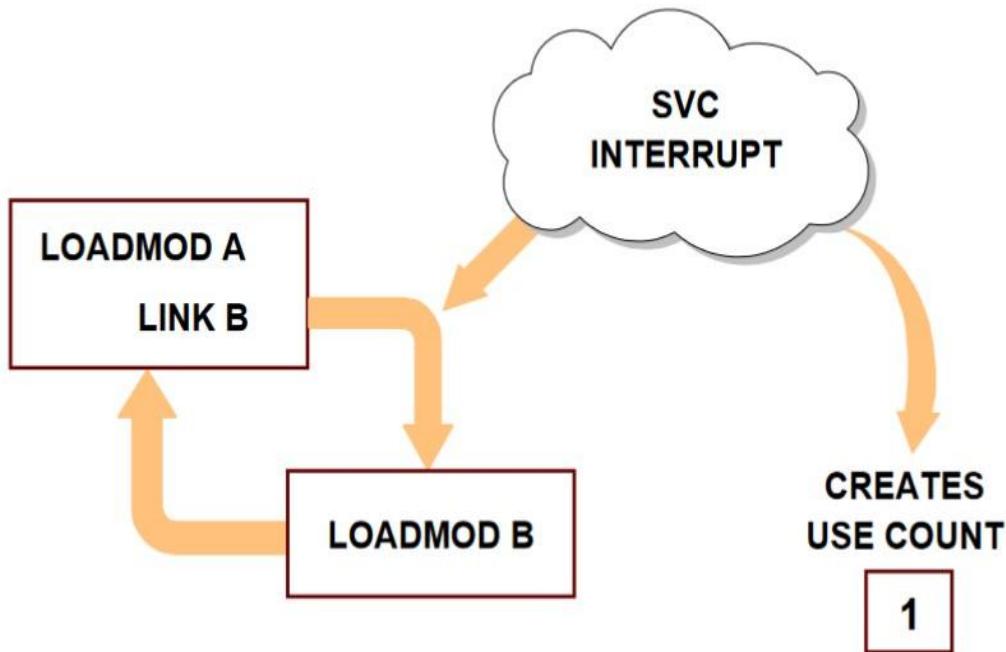
CALL compared to LINK:



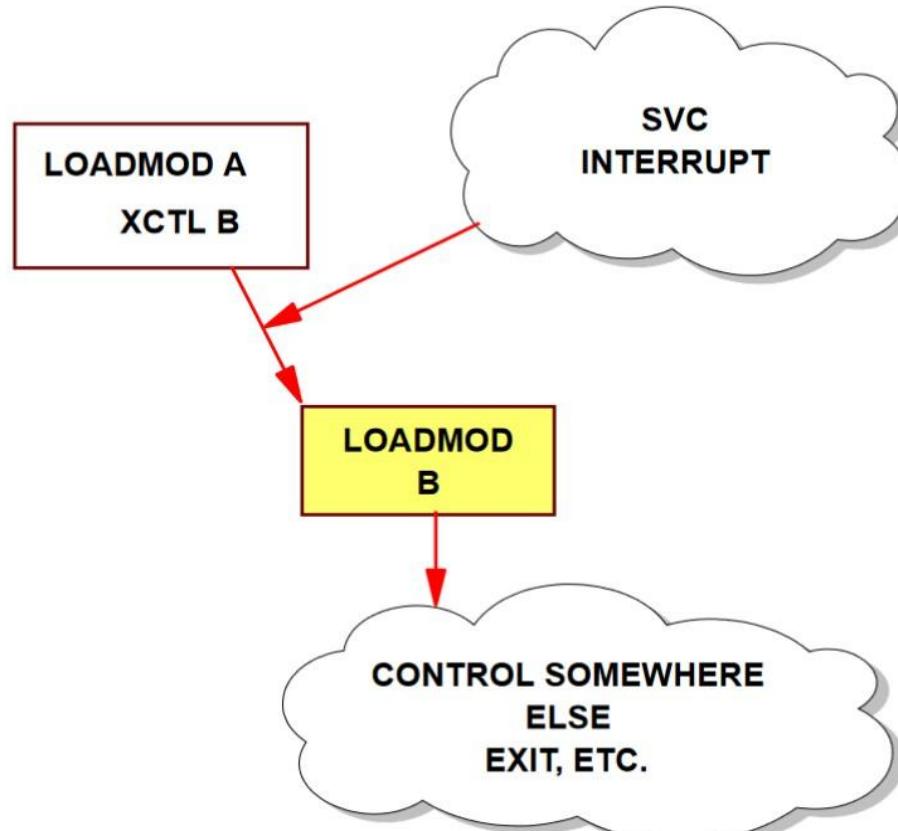
LOAD Macro Instruction



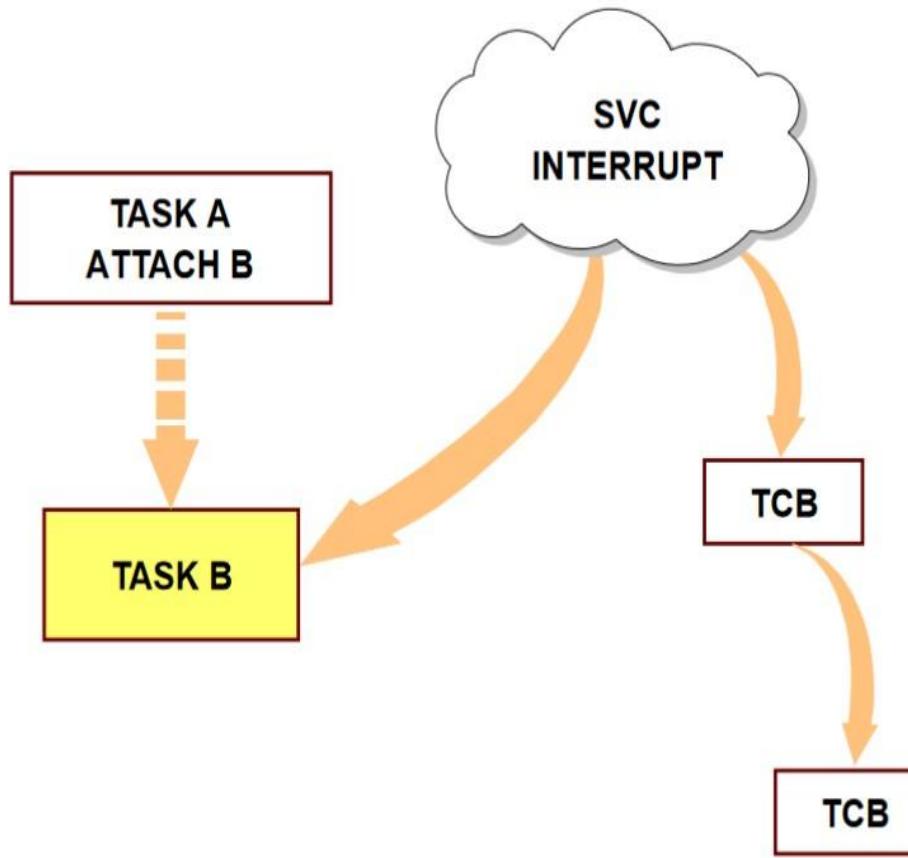
LINK Macro Instruction



XCTL Macro Instruction



ATTACH Macro Instruction



Services of the Supervisor - Dispatching



Program Linkage

Dispatching (of TCBs and SRBs)

Interrupt Handling

Supervisor Calls

Inter-address Space Communication

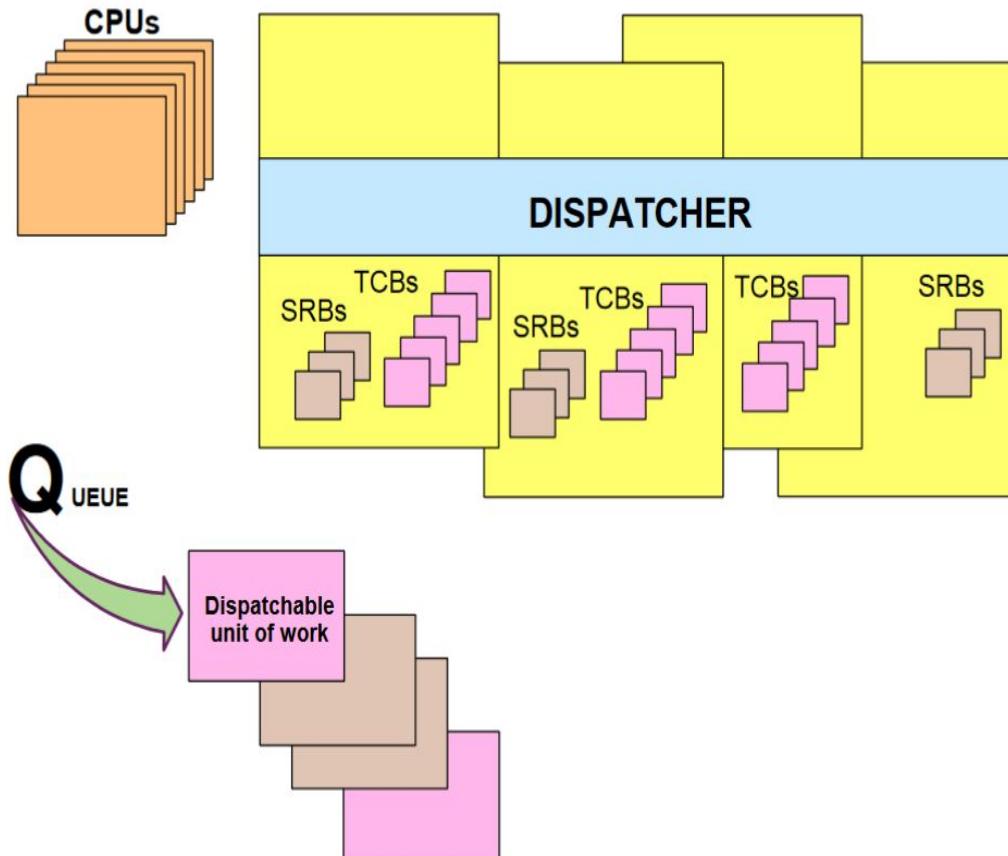
Serialization

UNIX System Services

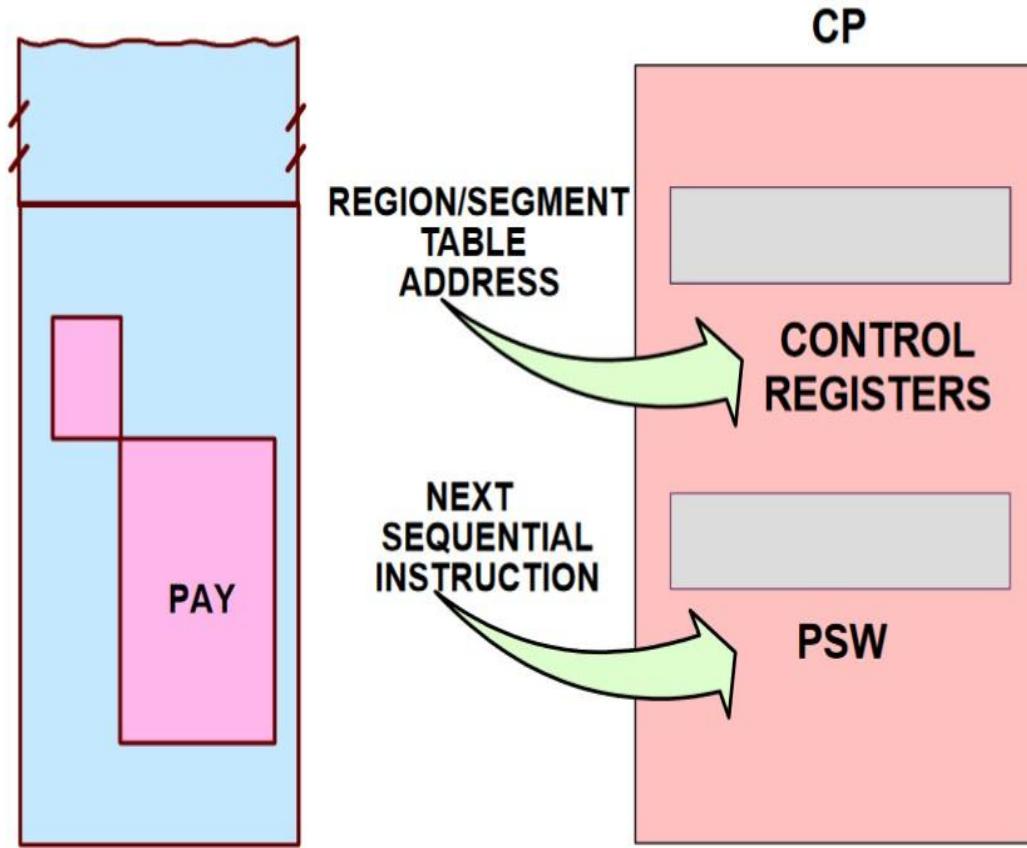
Resource Balancing

Measurement and Reporting

The Dispatcher



Dispatching a Task

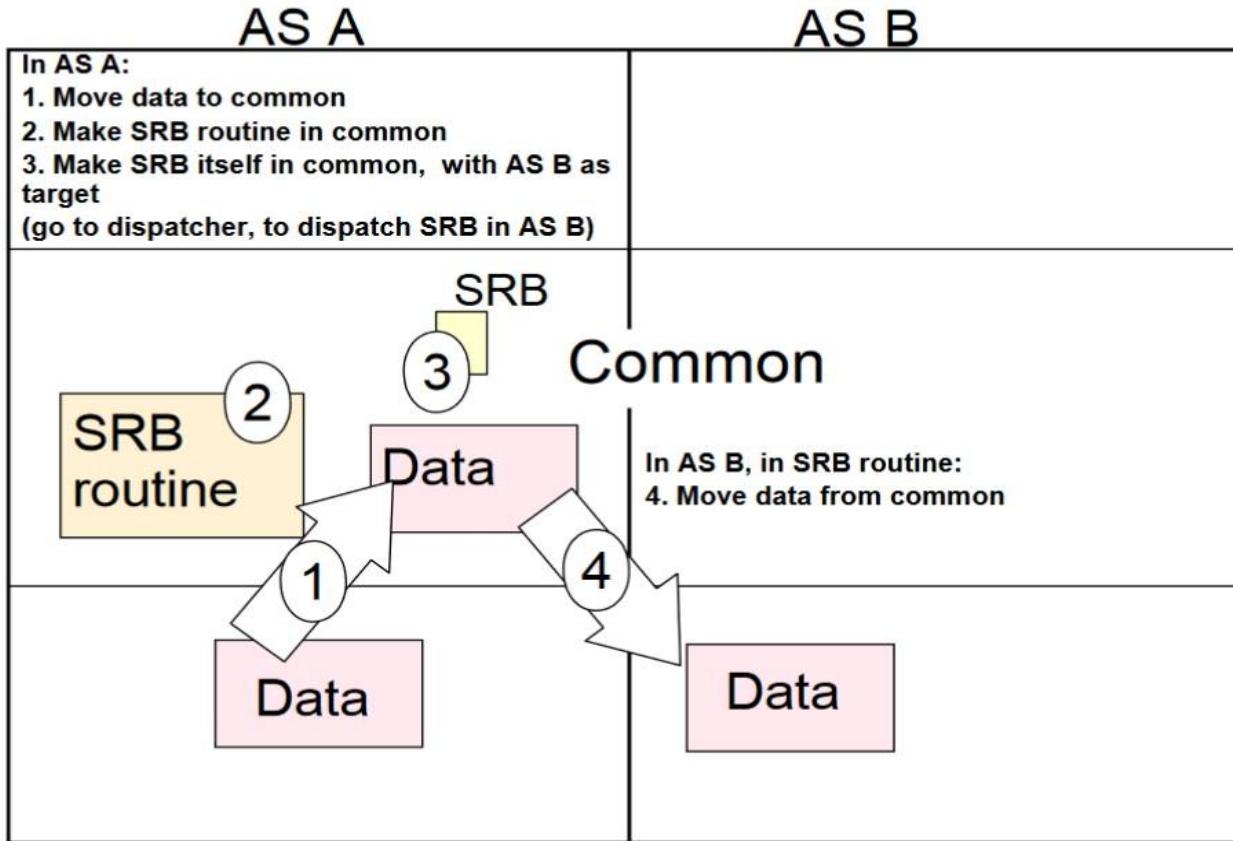


Dispatchable units of work

- TCB**: creado por la System Macro ATTACH desde un user o system program
- TCB** es borrado por el sistema al final de la tarea
- TCB** es colocado en LSQA
- Representa un normal work por ejemplo user-type of work
- El **TCB** mode no limita las funciones del sistema
- Prioridad acorde a las reglas estándar
- Pueden ser de ejecución larga o corta
- Se ejecuta sin autorización pero puede estar autorizado

- SRB**: creado por un código assembler específico (no es macro) por el system program
- SRB** debe ser explícitamente borrado por el código
- SRB** es almacenado en SQA
- Representa un normal work por ejemplo system-type of work
- Muchas funciones del sistema operativo no son permitidas en el SRB mode. No SVCs, LINK, I/O..
- Posee una prioridad especial alta
- Ejecuciones no preferentes
- Son de ejecución corta
- Son ejecuciones autorizadas

Ejemplo de SRB



Supervisión de la ejecución de los trabajos en el sistema.

En la multiprogramación, z/OS requiere el uso de varios supervisores de control, como ser:

- Procesamiento de interrupción (Interrupt processing).
- La multiprogramación requiere que exista alguna técnica para cambiar el control de una rutina, a otro modo que, por ejemplo, cuando la rutina “A” debe esperar que sea realizada una solicitud de I/O, se pueda ejecutar la rutina “B”.
- En z/OS, este modificador (o switch) es logrado mediante interrupciones, que son eventos que alteran la secuencia en la que el procesador ejecuta instrucciones.
- Cuando ocurre una interrupción, el sistema guarda el estado de ejecución de la rutina interrumpida y analiza y procesa la interrupción.
- Creación de unidades de trabajo despachables:
- Para identificar y realizar un seguimiento de su trabajo, el sistema operativo z/OS representa cada unidad de trabajo con un bloque de control.

- Dos tipos de bloques de control representan unidades de trabajo despachables: los bloques de control de tareas o TCB representan tareas ejecutando dentro de un address space; los bloques de solicitud de servicio o SRB representan servicios del sistema de mayor prioridad.

Despacho de trabajo (dispatching work):

Después de procesar las interrupciones, el sistema operativo determina qué unidad de trabajo (de todas las unidades de trabajo en el sistema) está listo para ejecutarse y tiene la más alta prioridad y pasa el control a esa unidad de trabajo.

Serializando el uso de recursos

- En un sistema multiprogramación, casi cualquier secuencia de instrucciones puede ser interrumpido para ser reanudado más tarde.
- Si ese conjunto de instrucciones manipula o modifica un recurso (por ejemplo, un bloque de control o un archivo de datos), el sistema operativo debe impedir que otros programas utilicen el recurso hasta que el programa interrumpido ha completado el procesamiento del recurso.
- Existen varias técnicas para serializar el uso de recursos; encolando y bloqueo (enqueueing and locking) son los más comunes (una tercera técnica se llama latching; que también es bloqueo).
- Todos los usuarios pueden utilizar encolamientos (enqueueing), pero sólo las rutinas autorizadas pueden utilizar locking para serializar el uso de recursos.

¿Qué es el procesamiento de interrupciones? (interrupt processing)

- Una interrupción es un evento que altera la secuencia en la que el procesador ejecuta instrucciones.
- Podría planificarse una interrupción (específicamente solicitada por el programa actualmente en ejecución) o no planeado (causado por un evento que podría o podría no estar relacionado con el programa que se está ejecutando actualmente).

z/OS utiliza seis tipos de interrupciones:

- Llamadas de supervisor o interrupciones de SVC (SVC interrupts):
 - Estos ocurren cuando el programa emite una SVC para solicitar un servicio en particular.
 - Un SVC interrumpe el programa que se está ejecutando y pasa el control al supervisor para que pueda realizar el servicio.
 - Los programas solicitan estos servicios a través de macros como OPEN (abrir un archivo), GETMAIN (obtener almacenamiento), o WTO (escribir un mensaje a consola).
- Interrupciones de E/S (I/O interrupts):
 - Estos ocurren cuando el subsistema de canales señala un cambio de estado, como una operación de E/S que se completa, o que se produce un error, o que un dispositivo de E/S, como una impresora está lista para funcionar.

- Interrupciones externas (External interrupts):
 - Estos pueden indicar cualquiera de varios eventos, como el vencimiento de un intervalo de tiempo, el operador presionando la tecla de interrupción en la consola, o el procesador recibiendo una señal de otro procesador.
- Reiniciar interrupciones (Restart interrupts):
 - Estos ocurren cuando el operador selecciona la función de reinicio en la consola o cuando se recibe una instrucción SIGP (procesador de señal) de reinicio de otro procesador.
- Interrupciones del programa (Program interrupts):
 - Estos son causados por errores del programa (por ejemplo, el programa intenta realizar una operación no válida), errores de página (el programa hace referencia a una página que no está en el almacenamiento central), o solicitudes para monitorear un evento.

- Interrupciones de verificación de la máquina (Machine check interrupts):
 - Estos son causados por mal funcionamiento de la máquina.
 - Cuando ocurre una interrupción, el hardware guarda la información pertinente sobre el programa que fue interrumpido y, si es posible, desactiva el procesador para continuar interrupciones del mismo tipo.
 - Luego, el hardware dirige el control a la dirección adecuada, a la rutina del controlador de interrupciones (interrupt handler routine).
 - La palabra de estado del programa o PSW es un recurso clave en este proceso.

¿Cómo se utiliza la palabra de estado del programa (PSW)?

- La palabra de estado de programa (PSW) es un área de datos de 128 bits en el procesador que, junto con una variedad de otros tipos de registros (registros de control, registros de sincronización, y registros de prefijo) proporciona detalles cruciales tanto para el hardware como para el software.
- La PSW actual incluye la dirección de la siguiente instrucción del programa. y controlar la información sobre el programa que se está ejecutando.
- Cada procesador tiene sólo un PSW actual.
- Por lo tanto, sólo se puede ejecutar una tarea en un procesador a la vez.
- La PSW controla el orden en que las instrucciones se envían al procesador e indica el estado del sistema en relación con el programa actualmente en ejecución.
- Aunque cada procesador tiene sólo un PSW, es útil pensar en tres tipos de PSW para comprender el procesamiento de interrupciones:
- PSW actual (current PSW)
- PSW nueva (new PSW)
- PSW anterior (old PSW)

- La PSW actual indica la siguiente instrucción a ejecutar, también indica si el procesador está habilitado o deshabilitado para interrupciones de E/S; interrupciones, interrupciones de verificación de la máquina y ciertas interrupciones de programas.
- Cuando el procesador está habilitado, estas interrupciones pueden ocurrir. Cuando el procesador está deshabilitadas, estas interrupciones se ignoran o permanecen pendientes.
- Hay una PSW nueva y un PSW antigua (old) asociados con cada uno de los seis tipos de interrupcion.
- La nueva PSW contiene la dirección de la rutina que puede procesar su interrupción asociada.

- Si el procesador está habilitado para interrupciones, cuando una interrupción ocurre, las PSW se cambian (switch) usando la siguiente técnica:
 1. Almacenar la PSW actual en la PSW antiguo (old) asociado con el tipo de interrupción que ocurrió
 2. Cargando el contenido de la nueva PSW para el tipo de interrupción que ocurrió en el PSW actual. La PSW actual, indica la siguiente instrucción a ejecutar, que ahora contiene la dirección de la rutina apropiada para manejar la interrupción. Este interruptor (switch) tiene el efecto de transferir el control al sistema, de rutinas del manejo de interrupciones apropiado.

Registros y PSW

- La arquitectura mainframe proporciona registros para realizar un seguimiento de las cosas.
- La PSW, por ejemplo, es un registro que se utiliza para contener información necesaria para la ejecución del programa actualmente activo. Los mainframes proporcionan otros registros, como:
- Los registros de acceso se utilizan para especificar el espacio de direcciones en el que se encuentran los datos.
- Los registros generales se utilizan para direccionar los datos almacenados y también para mantener datos.
- Los registros de coma flotante se utilizan para contener datos numéricos en forma de coma flotante.
- Los registros de control son utilizados por el propio sistema operativo, por ejemplo, como referencias a tablas de traducción o translación.

Unidades de trabajo despachables (dispatchable units of work)

- En z/OS, las unidades de trabajo despachables están representadas por dos tipos de control bloques:
- Bloques de control de tareas (TCB) Estos representan tareas que se ejecutan dentro de un espacio de direcciones, como usuario programas y programas del sistema que soportan los programas del usuario.
- Bloques de solicitud de servicio (SRB) Estos representan solicitudes para ejecutar una rutina de servicio del sistema. Proporcionan un mecanismo de comunicación entre espacios de direcciones.

¿Qué es una TCB ?

Una TCB es un bloque de control que representa una tarea, como su programa, mientras se ejecuta en un espacio de direcciones. Una TCB contiene información sobre la tarea en ejecución, como:

16 Registros de Acceso (32 bits)

16 registros de uso general (64 bits)

16 registros de coma flotante (64 bits)

¿Qué address space ?

dirección de datos datos, en virtual storage

(por e.j.) MVC B,A

MVC C,B

La instrucción Move (MVC): mueve el contenido del segundo operando a la ubicación del primer operando.

Hasta 5 niveles de tablas de traducción la dirección de cualquier área de almacenamiento que haya creado.

Las TCB se crean en respuesta a una macro ATTACH.

Al usar la macro ATTACH, un programa de usuario o rutina del sistema comienza la ejecución del programa especificado en la macro ATTACH, como una subtarea (subtaks).

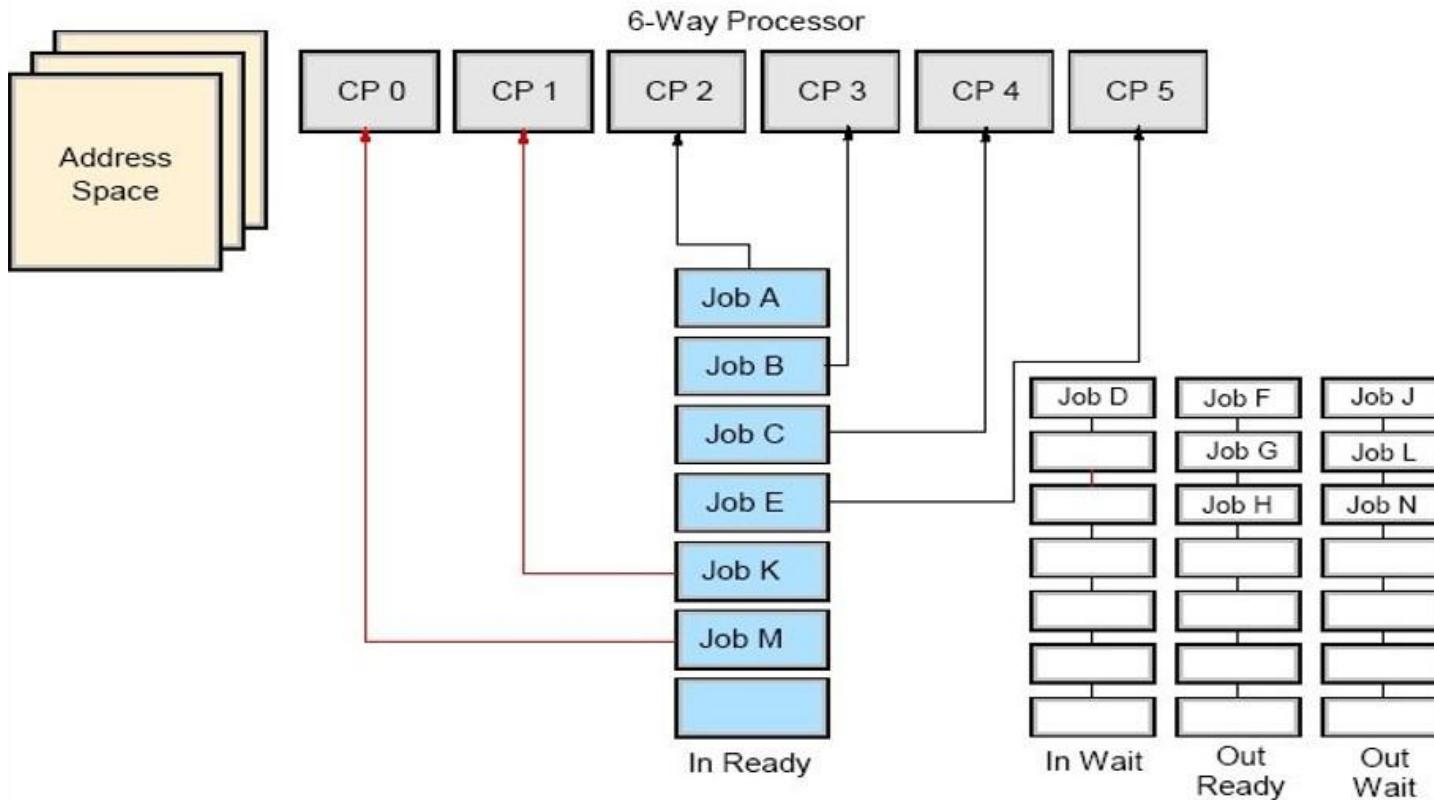
- Como una subtarea, el programa especificado puede competir por el tiempo del procesador y puede utilizar ciertos recursos ya asignados.
- La tarea de control de región (RCT Region Control Task), se encarga de preparar un address space para swap-in y swap-out, es la tarea de mayor prioridad en un address space.
- Todas las tareas dentro de un address space son subtareas (subtasks) de RCT.

¿Qué es un SRB ?

- Un SRB es un bloque de control que representa una rutina que realiza una determinada función o servicio en un address space.
- Normalmente, se crea una SRB cuando un address space se está ejecutando y ocurre un evento que afecta a otro address space.
- La rutina que realiza la función o servicio se denomina rutina SRB; iniciar el proceso se llama programar una SRB; la rutina SRB se ejecuta en el modo de funcionamiento conocido como modo SRB.
- Un SRB es similar a un TCB en que identifica una unidad de trabajo para el sistema. A diferencia de TCB, una SRB no puede ser “dueña” de áreas de almacenamiento. Las rutinas SRB pueden obtener, referencia, uso y libre almacenamiento, pero dichas áreas deben ser propiedad de un TCB.

- Como se mencionó anteriormente, un SRB proporciona un medio de comunicación asincrónica entre espacios de direcciones para programas que se ejecutan en z/OS. Sólo los programas que se ejecutan en un modo de autoridad superior llamado estado supervisor pueden crear una SRB. Estos programas autorizados obtienen almacenamiento e inicializan el bloque de control con cosas como la identidad del espacio de direcciones de destino y punteros al código que procesará la solicitud.
- El programa que crea la SRB, luego emite la macro SCHEDULE e indica si la SRB tiene prioridad global (system wide; en todo el sistema) o local (en el address space).
- El sistema coloca el SRB en la cola de despacho apropiada donde permanecerá hasta que se convierta en el trabajo de mayor prioridad en la cola.
- Los SRB con prioridad global tienen una prioridad más alta que la de cualquier espacio de direcciones, independientemente del espacio de direcciones real en el que se ejecutarán.
- SRB con una prioridad local tiene una prioridad igual a la del espacio de direcciones en el que ejecutarse, pero superior a cualquier TCB dentro de ese espacio de direcciones. La asignación de prioridad global o local depende de la “importancia” de la solicitud; por ejemplo, los SRB para interrupciones de E/S se programan con una prioridad global, para minimizar los retrasos de E/S.

Despacho de Tareas



¿Qué hace el Dispatcher?

- Selecciona un nuevo trabajo, por ejemplo, cuando una tarea se interrumpe o se vuelve no despachable, o después de que un SRB se complete o se suspende (es decir, un SRB es retrasado porque un recurso requerido no está disponible).
- En z/OS, el componente Dispatcher es responsable de enrutar el control a la unidad de trabajo de mayor prioridad que está lista para ejecutarse.
- El dispatcher procesa los trabajos en el siguiente orden:
 1. Special Exits:
- Estas son exits a rutinas que tienen una alta prioridad debido a motivos específicos. condiciones en el sistema.
- Por ejemplo, si un procesador en un SRB de multiprocesamiento puede ser interrumpido por el programa emisor, para permitir el trabajo con una prioridad igual o mayor, tener acceso al procesador.

- Además, los SRB de cliente y los SRB de enclave son “preemptable” (preferibles; estos temas están fuera del alcance de este curso).
- Un TCB no es interruptible cuando ejecuta un SVC, el sistema falla, se invoca una recuperación alternativa de la CPU mediante una exit especial para recuperar el trabajo que se estaba ejecutando en el procesador defectuoso.

2. SRB que tienen una prioridad global

3. Address space ready en orden de prioridad

Un address space está listo para ejecutarse si está swapped-in y no en espera de algún evento para completar.

La prioridad de un address space está determinada por la prioridad de despacho especificada por el usuario o la instalación. Después de seleccionar el address space de mayor prioridad, z/OS (a través del despachador) despacha los SRB con prioridad local que están programados para ese address space y luego los TCB en ese address space. Si no hay ningún trabajo listo en el sistema, z/OS asume un estado llamado enable-wait hasta que entre nuevo trabajo en el sistema.

Los diferentes modelos de hardware z/Series pueden tener varios procesadores centrales (CP).

Todos y cada uno de los CP pueden estar ejecutando instrucciones al mismo tiempo. Dispatching priorities, determinan cuándo un address space esta: ready-to-execute consigue ser despachado.

La figura anterior representa: Trabajo de despacho, en un procesador de 6 vías (6-ways)

In-ready address space

CP 0 CP 1 CP 2 CP 3 CP 4 CP 5

Trabajo A Trabajo B Trabajo C Trabajo E Trabajo K Trabajo M

In-wait:

Trabajo D Trabajo F Trabajo G Trabajo H Trabajo J Trabajo L Trabajo N

Out-wait

Out-ready

Un address space puede estar en cualquiera de las cuatro colas:

IN-READY: en almacenamiento central y esperando ser despachado

IN-WAIT: en almacenamiento central pero esperando a que se complete algún evento.

OUT-READY: Listo para ejecutar pero swapeado.

OUT-WAIT: swapeado y esperando a que se complete algún evento.

Sólo se pueden seleccionar trabajos IN-READY para su despacho.

Services of the Supervisor - Interrupt Handling

Program Linkage

Dispatching

Interrupt Handling

Supervisor Calls

Inter-address Space Communication

Serialization

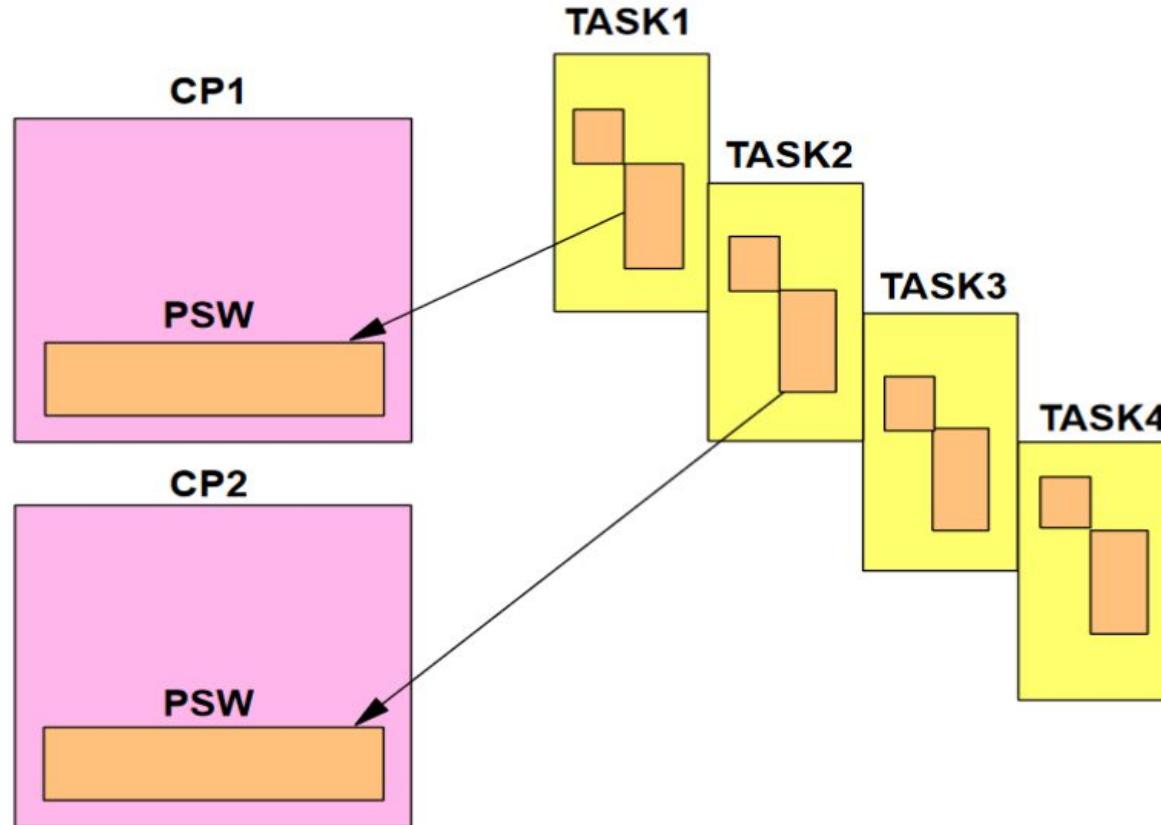
UNIX System Services

Resource Balancing

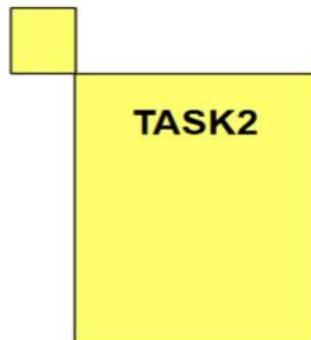
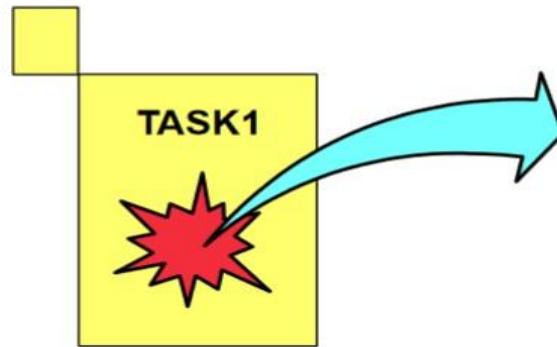
Measurement and Reporting



Función de las Interrupts



Tipos de Interrupts



SUPERVISOR CALL

PROGRAM CHECK

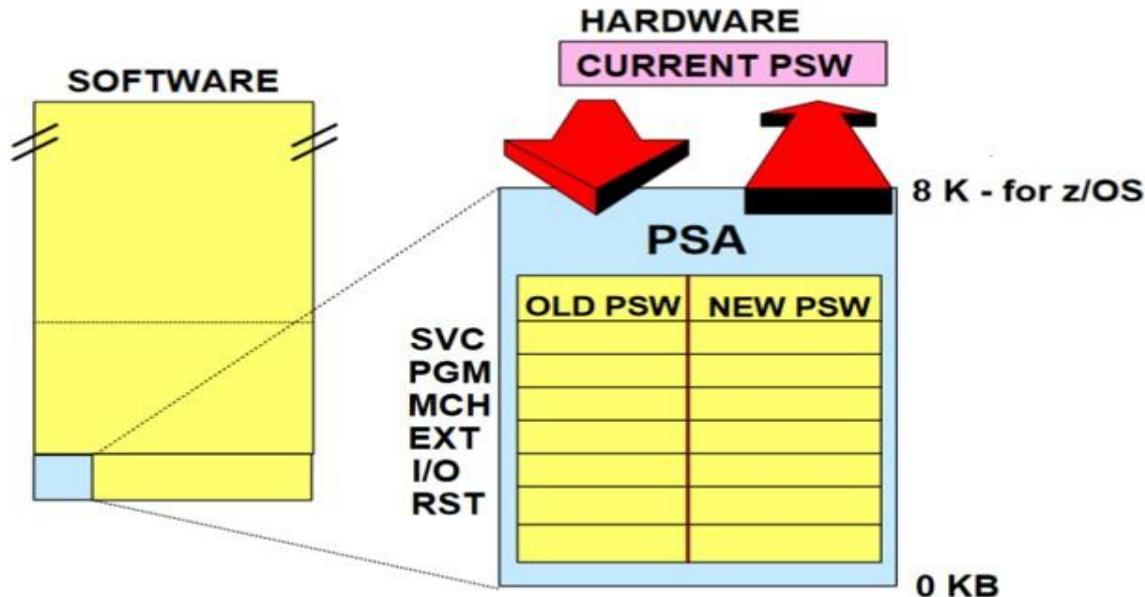
MACHINE CHECK

EXTERNAL

I/O

RESTART

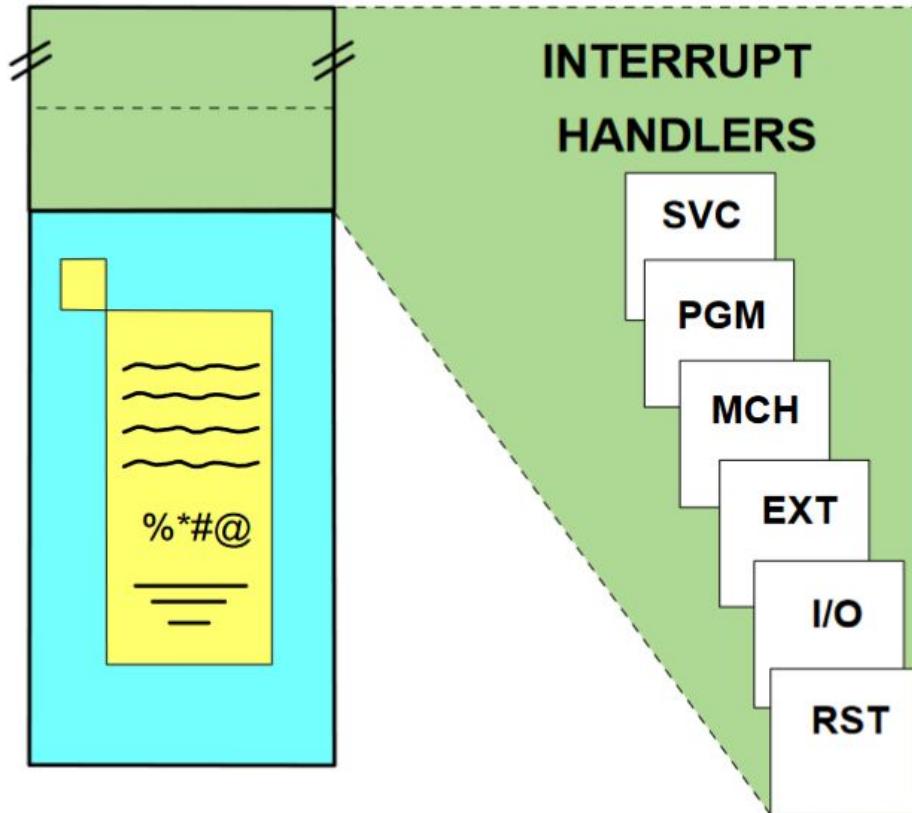
PSW Switch



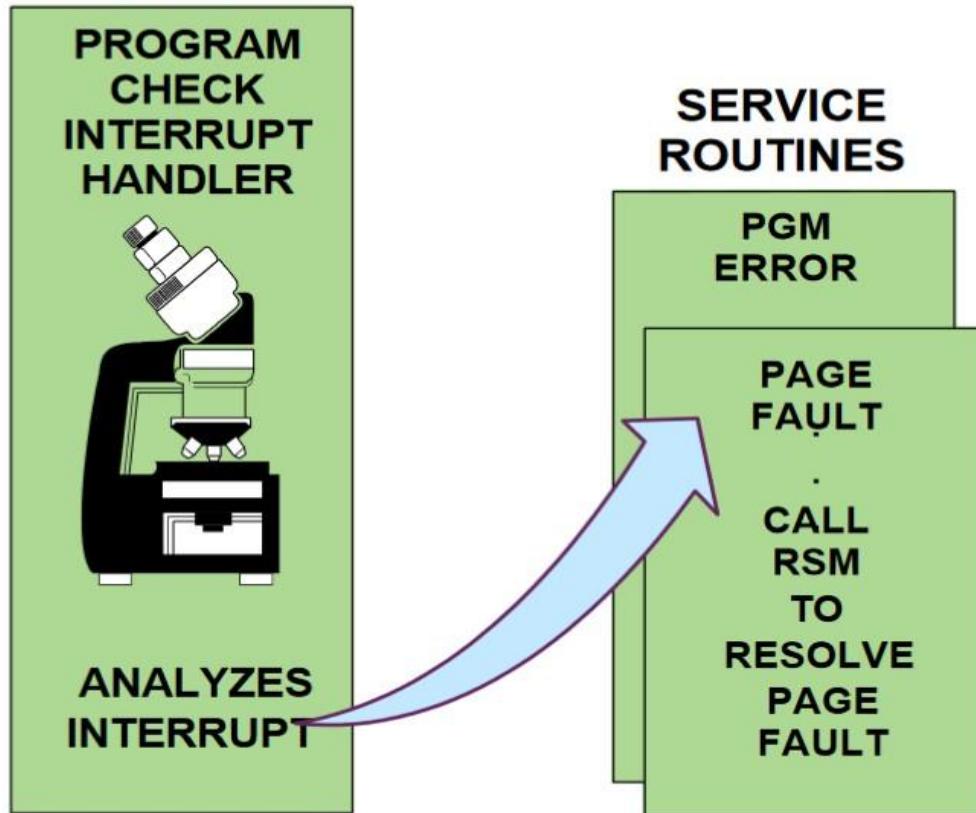
64-bit z/Architecture PSA
- New/Old PSWs Moved
Restart New PSW now 8 bytes at 0-7

Machine Check and Store Status areas remapped
z/Architecture Old/New PSW area (288-511=X'120'-X'1FF')

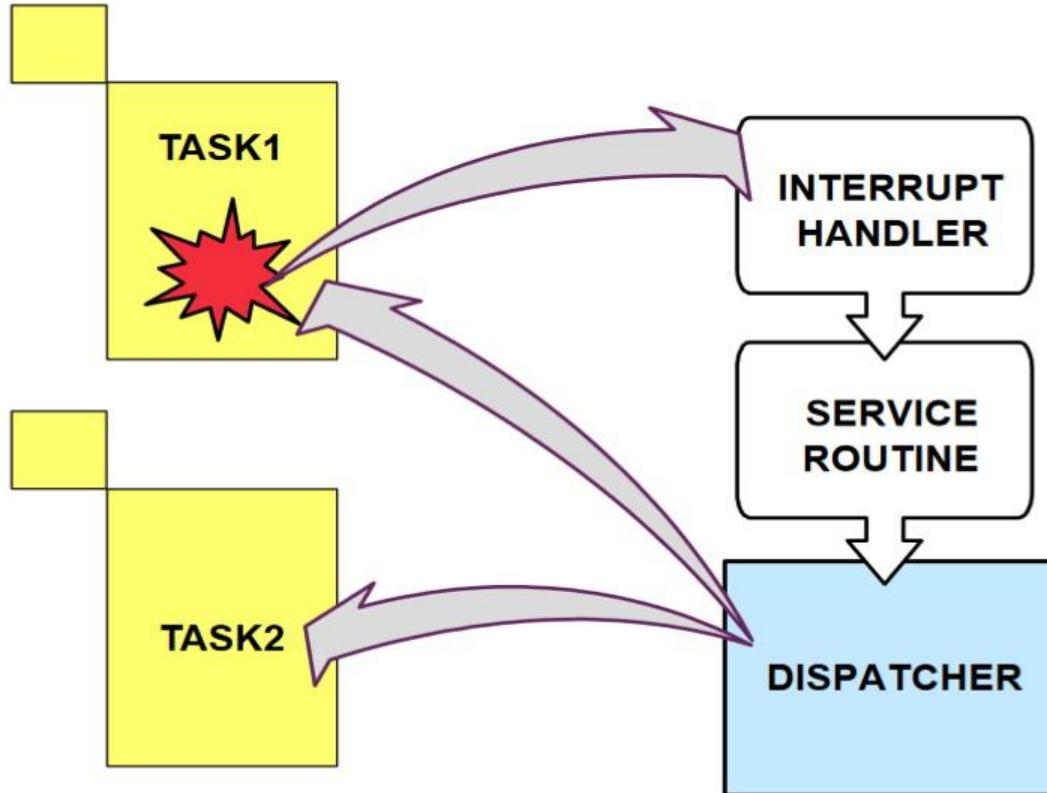
Interrupt Handling



Interrupt Analysis



Post Interrupt Processing



Services of the Supervisor - Supervisor Calls

- Program Linkage
- Dispatching
- Interrupt Handling
-  Supervisor Calls
- Inter-address Space Communication
- Serialization
- UNIX System Services
- Resource Balancing
- Measurement and Reporting

- Una solicitud de programa para un servicio de supervisión
 - Código Complejo
 - Funciones Comunes
 - Actividades limitadas al estado supervisor

Issuing SVCs

MY PROGRAM

.

.

.

MACRO

+

+

+

SVC

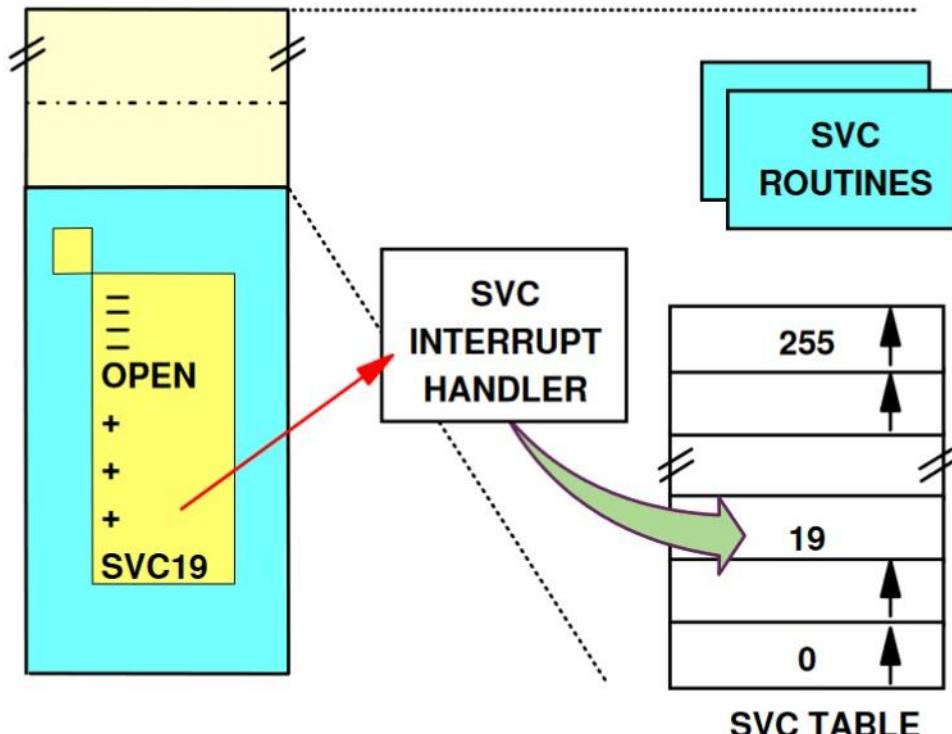
.

.

.

.

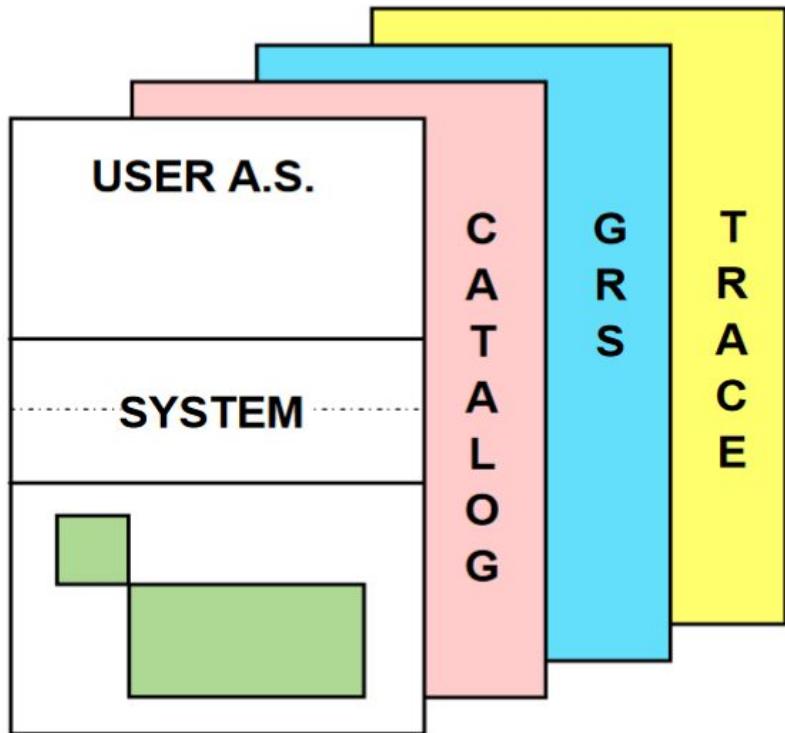
SVC Interrupt Processing



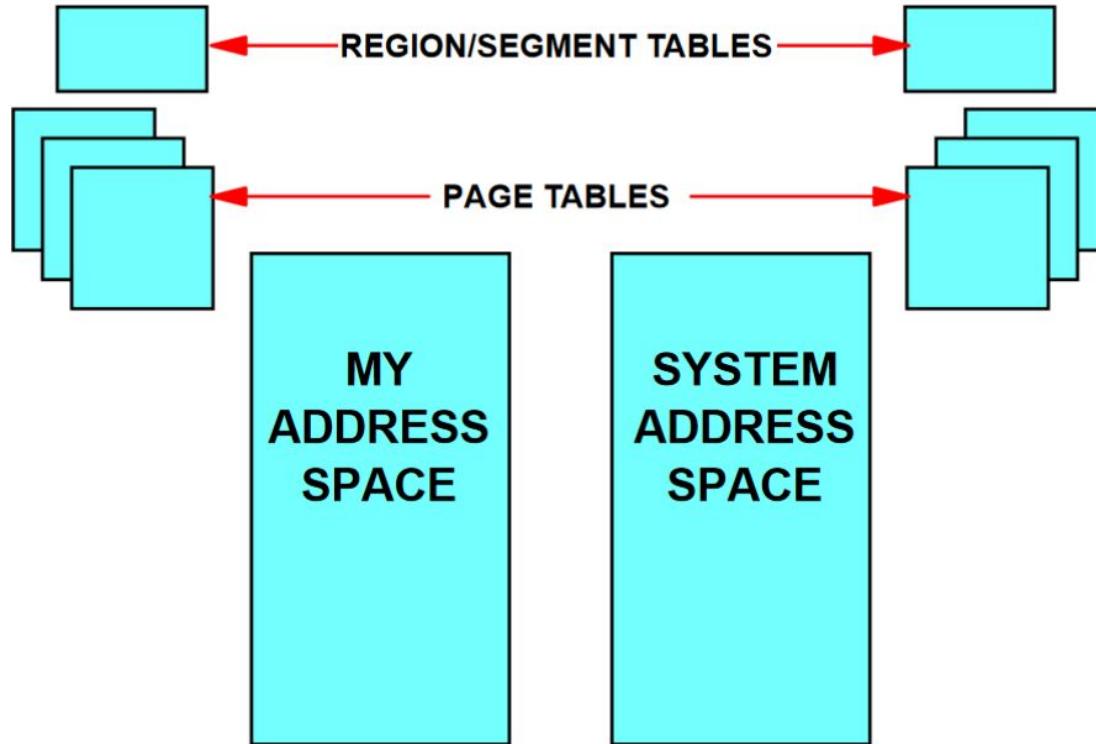
Services of the Supervisor - Inter AS Comm.

- Program Linkage
- Dispatching
- Interrupt Handling
- Supervisor Calls
-  Inter-address Space Communication
- Serialization
- UNIX System Services
- Resource Balancing
- Measurement and Reporting

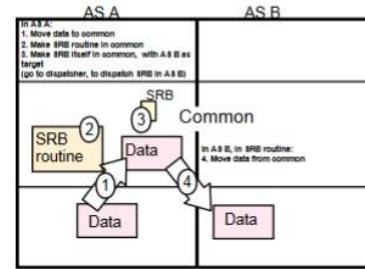
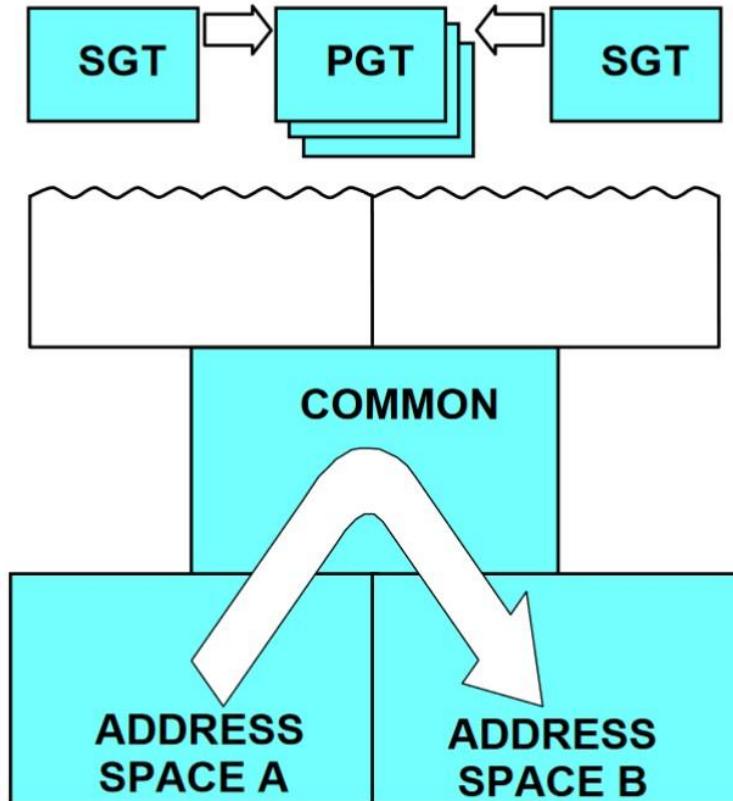
¿Por qué Inter-Address Space Communications?



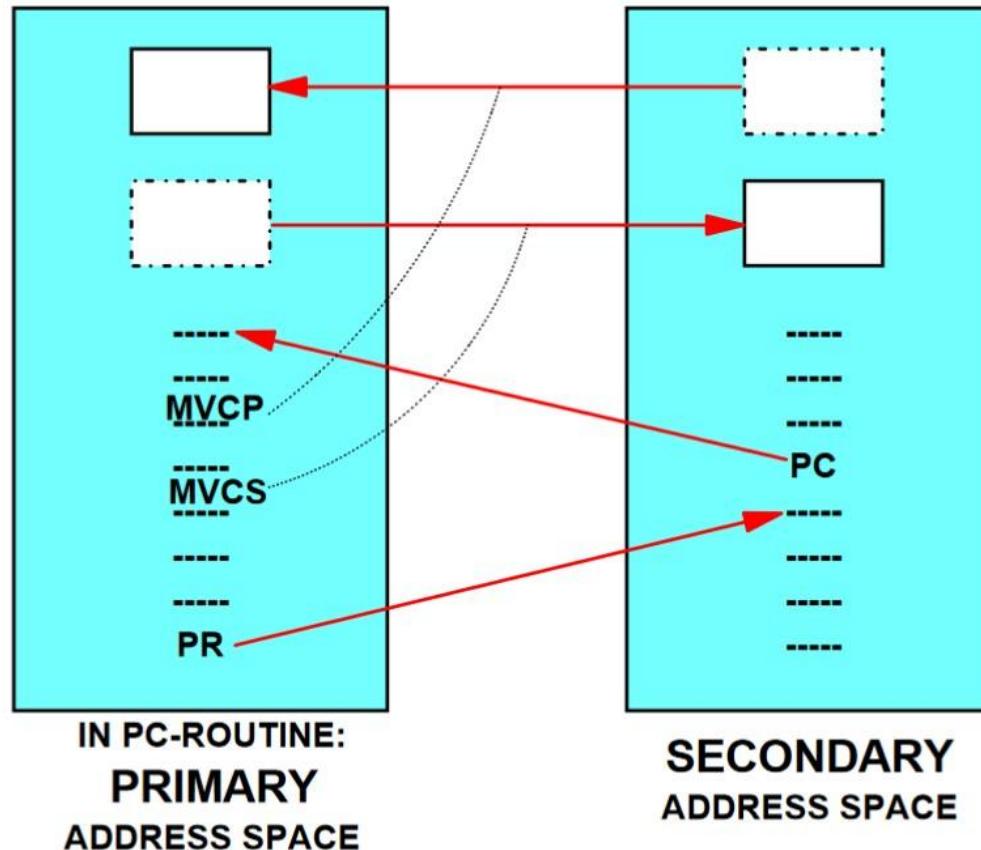
Multiple Address Space Architecture



La Comunicación a través del Common



Cross Memory Services Instructions



Services of the Supervisor - Serialization

Program Linkage

Dispatching

Interrupt Handling

Supervisor Calls

Inter-address Space Communication



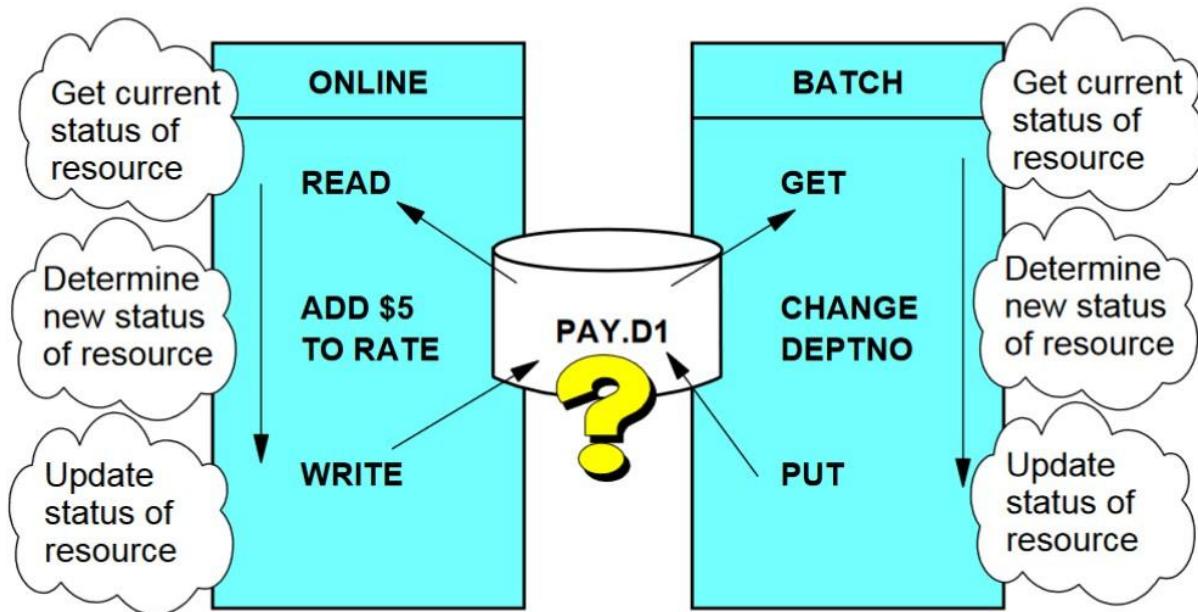
Serialization

UNIX System Services

Resource Balancing

Measurement and Reporting

¿Por Qué Serializar?



Serialización en Z/os

- Used in application programs
- (Almost) only used in system code
- Only used in system code
- Only used in system code
- (Almost) only used in system code
- Used in HLL application programs
- Only used in system code

ENQ and DEQ macros

To serialize resources (such as data sets) within a single system or across multiple systems.

WAIT and POST macros

Control the parallel execution of tasks (TCBs)

SUSPEND and RESUME macros

Control the parallel execution of SRBs

PAUSE and RELEASE macros

Control the parallel execution of TCBs and SRBs across several address spaces

RESERVE macro

Allow multiple systems to share a DASD

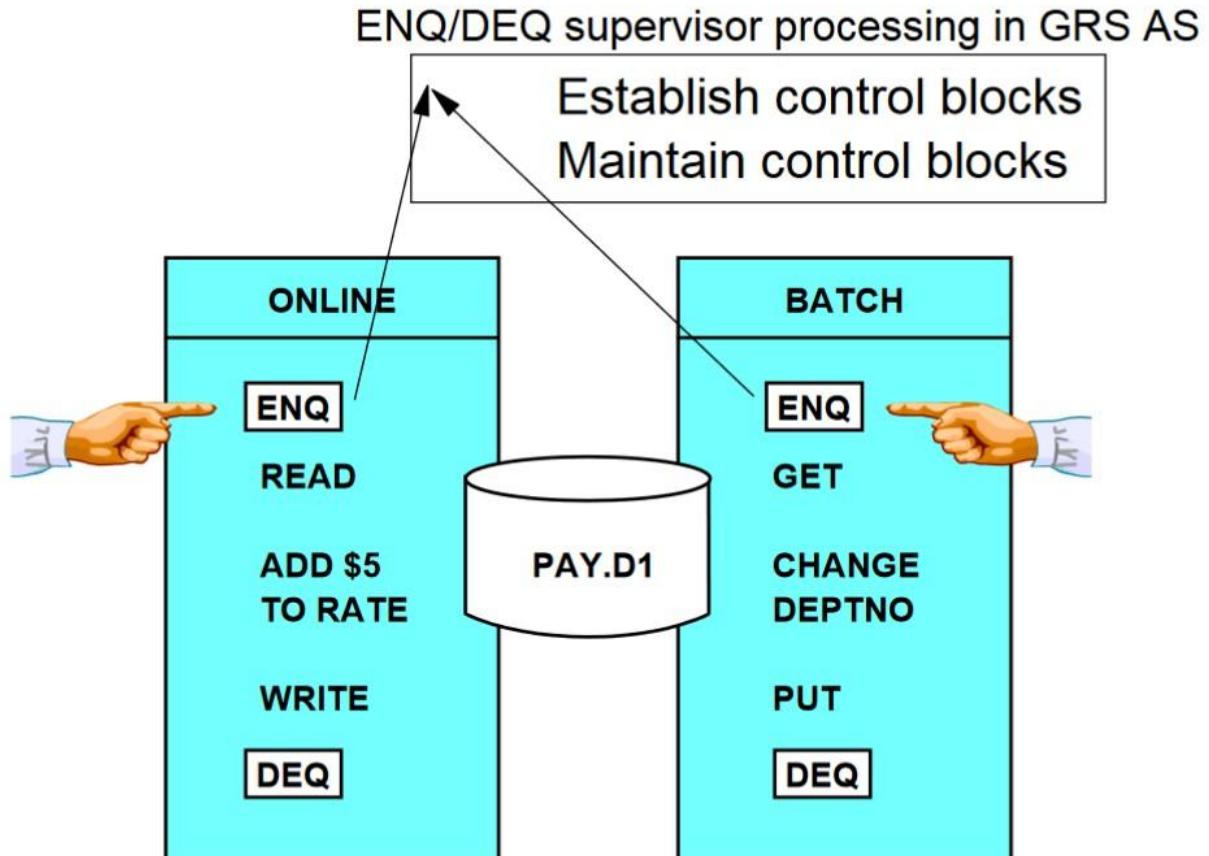
Latch manager callable services

Serialize resources (such as control blocks or data sets) and synchronize the execution of programs

Locking (SETLOCK macro)

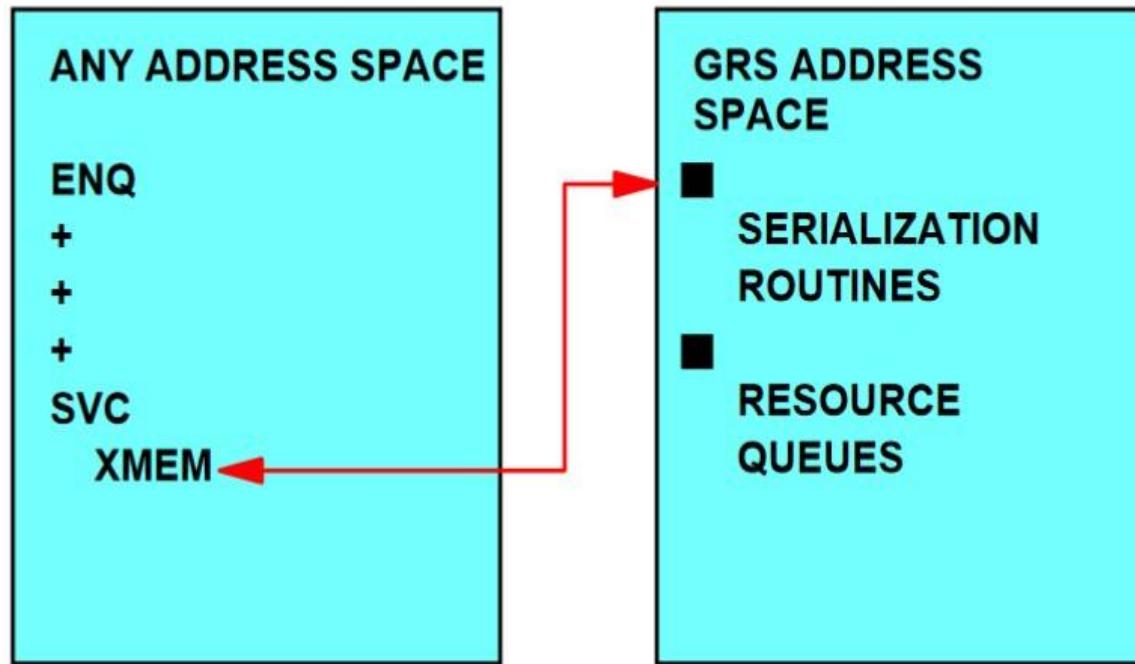
Serialize system resources such as control blocks

Serialización de Data Set compartidos

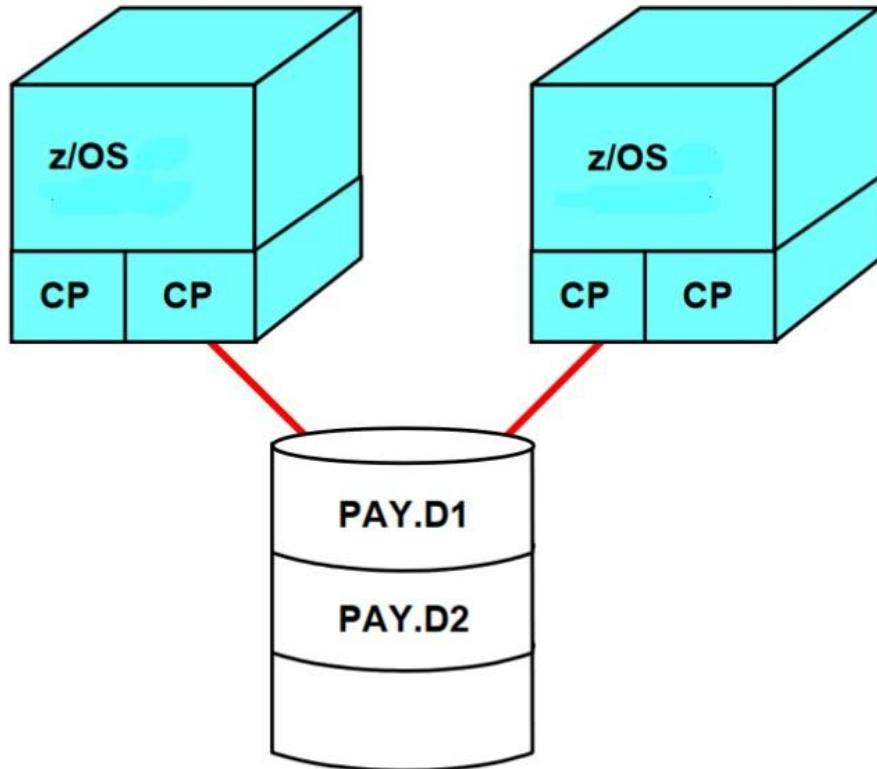


Global Resource Serialization (GRS)

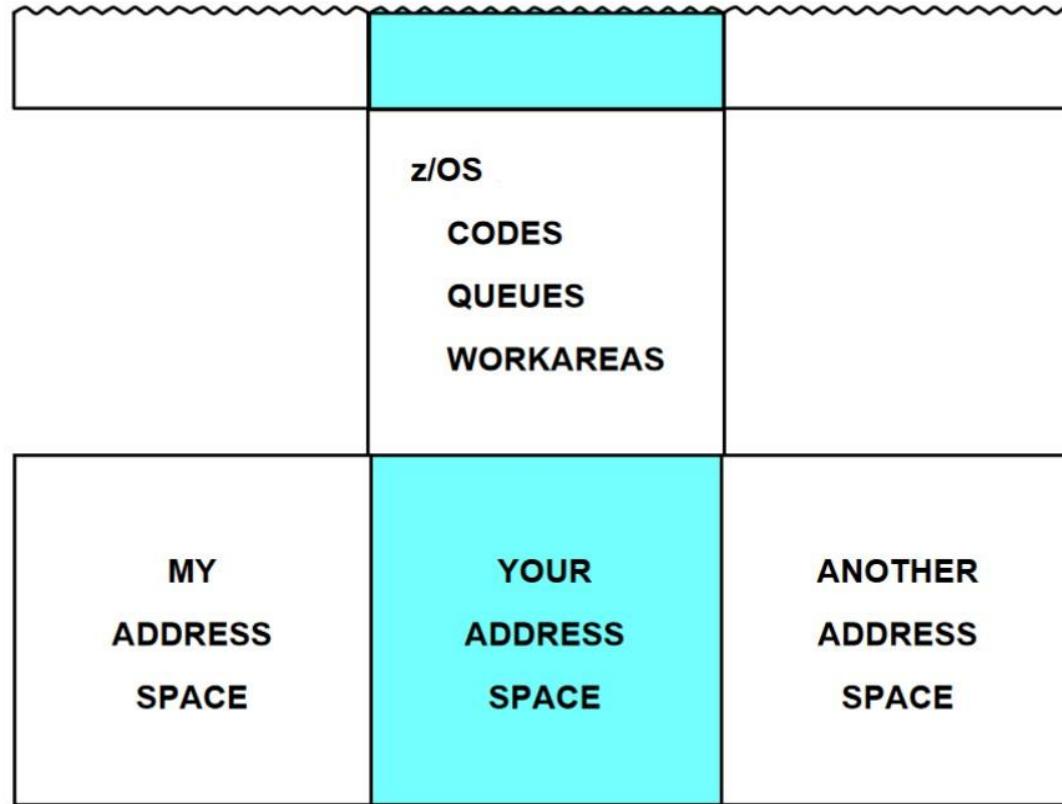
// DS1 DD DSN=PAY.D1,DISP=OLD



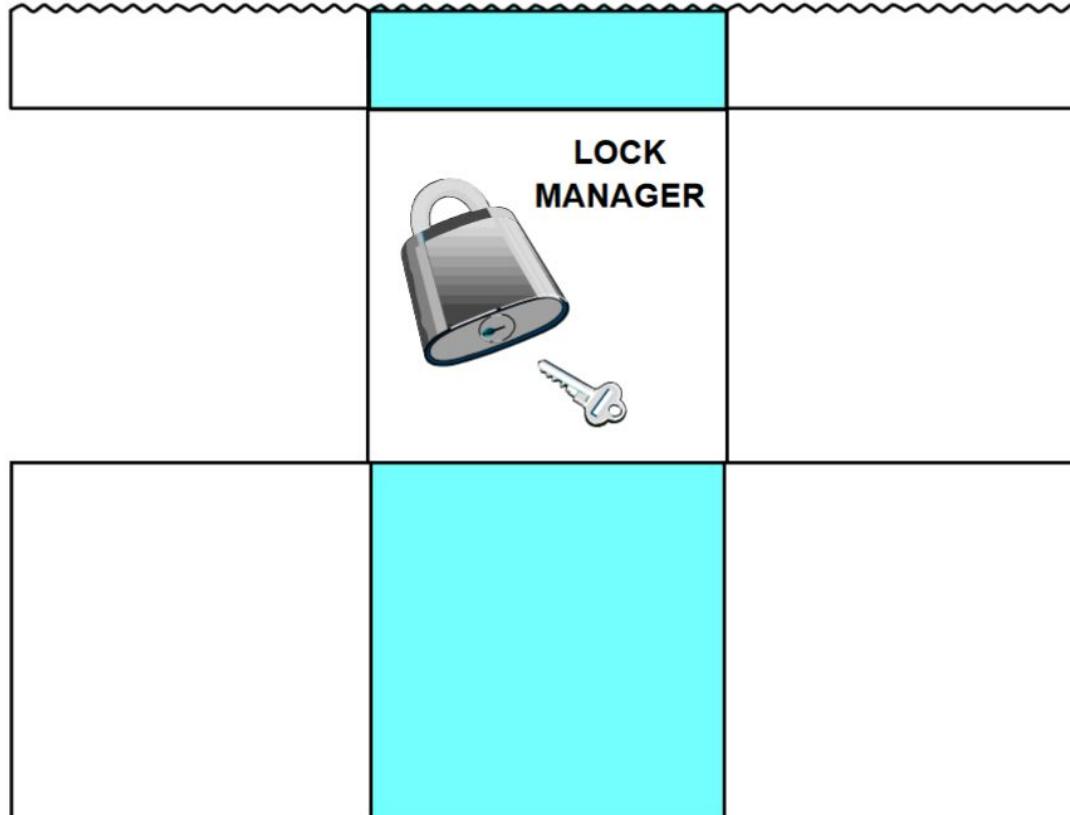
Serialización de Hardware



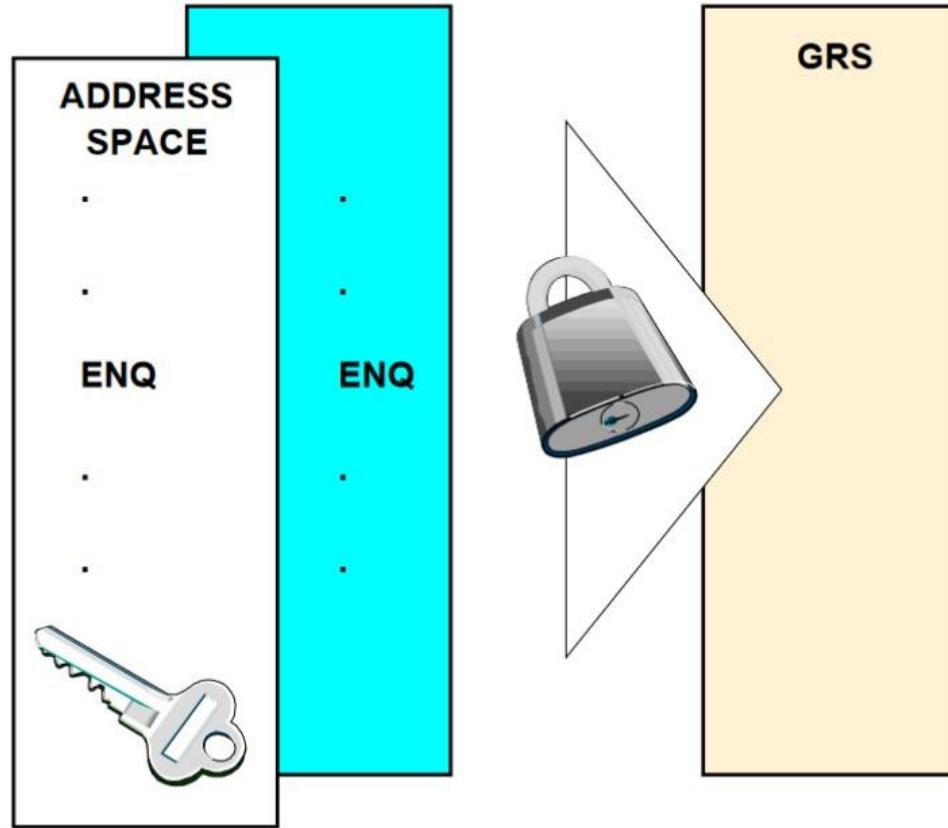
Serialización del Sistema de Recursos



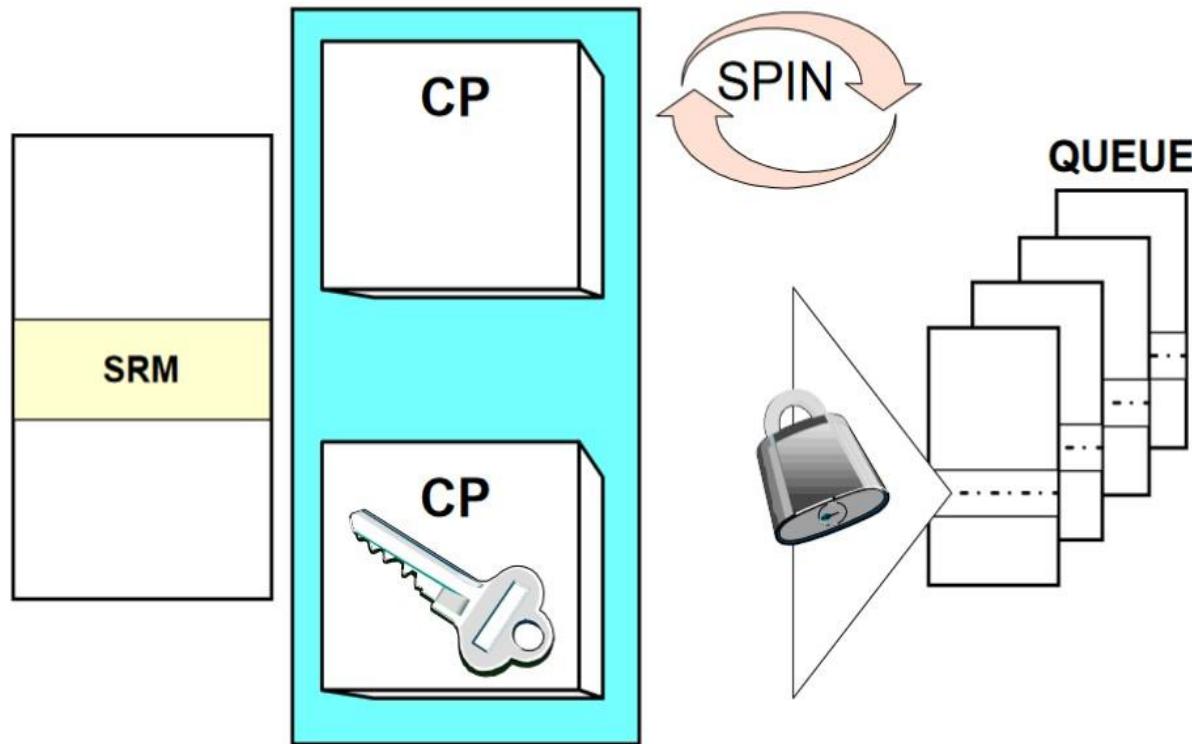
Locking



Suspend Lock



Spin Lock

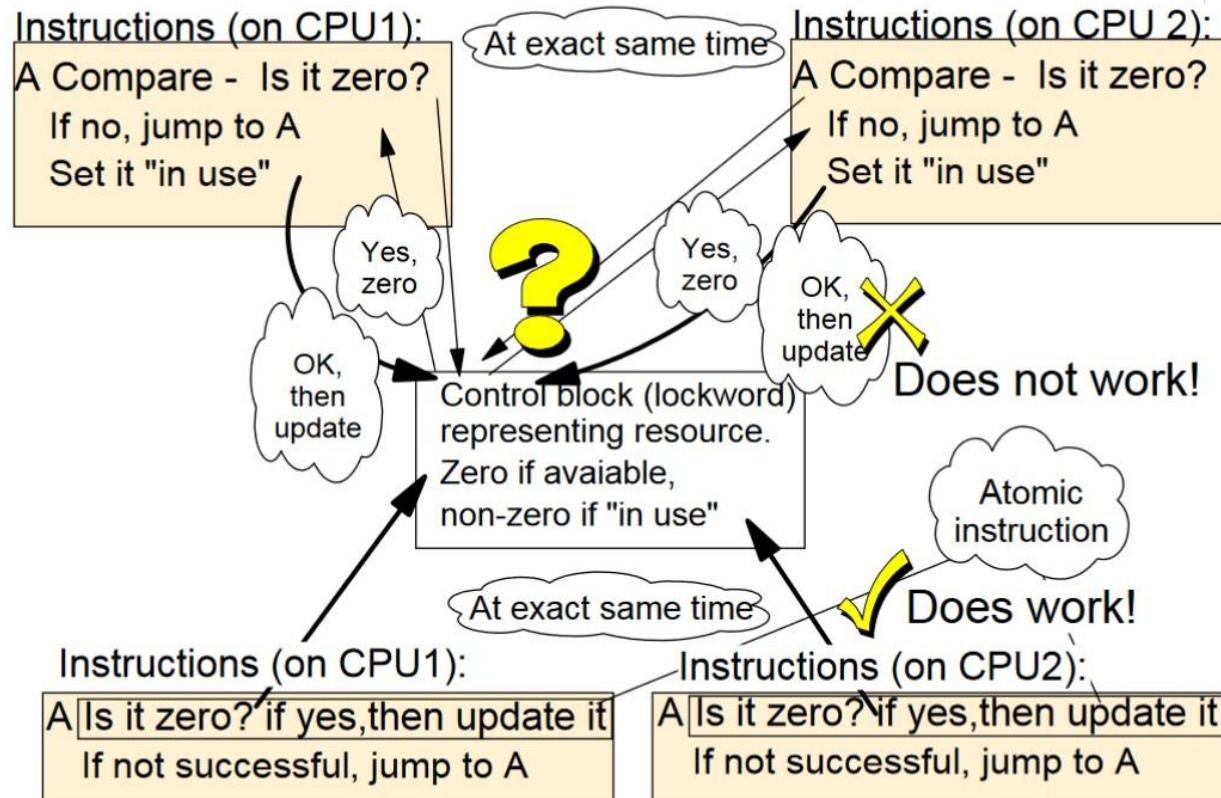


Spin Lock

En z/OS, existen dos tipos de bloqueos:

- Los Bloqueos Globales, para recursos relacionados con más de un espacio de direcciones. Los bloqueos globales se proporcionan para rutinas no reutilizables o no compatibles y varios recursos.
- Los Bloqueos Locales, para recursos asignados a un espacio de direcciones concreto.

¿Qué es primero en la Serialización?



¿Qué es primero en la Serialización?

ENQ/DEQ

- Sistema Simbólico de Reservas
- Uno o Múltiples sistemas
- Serializar Recursos de Usuario

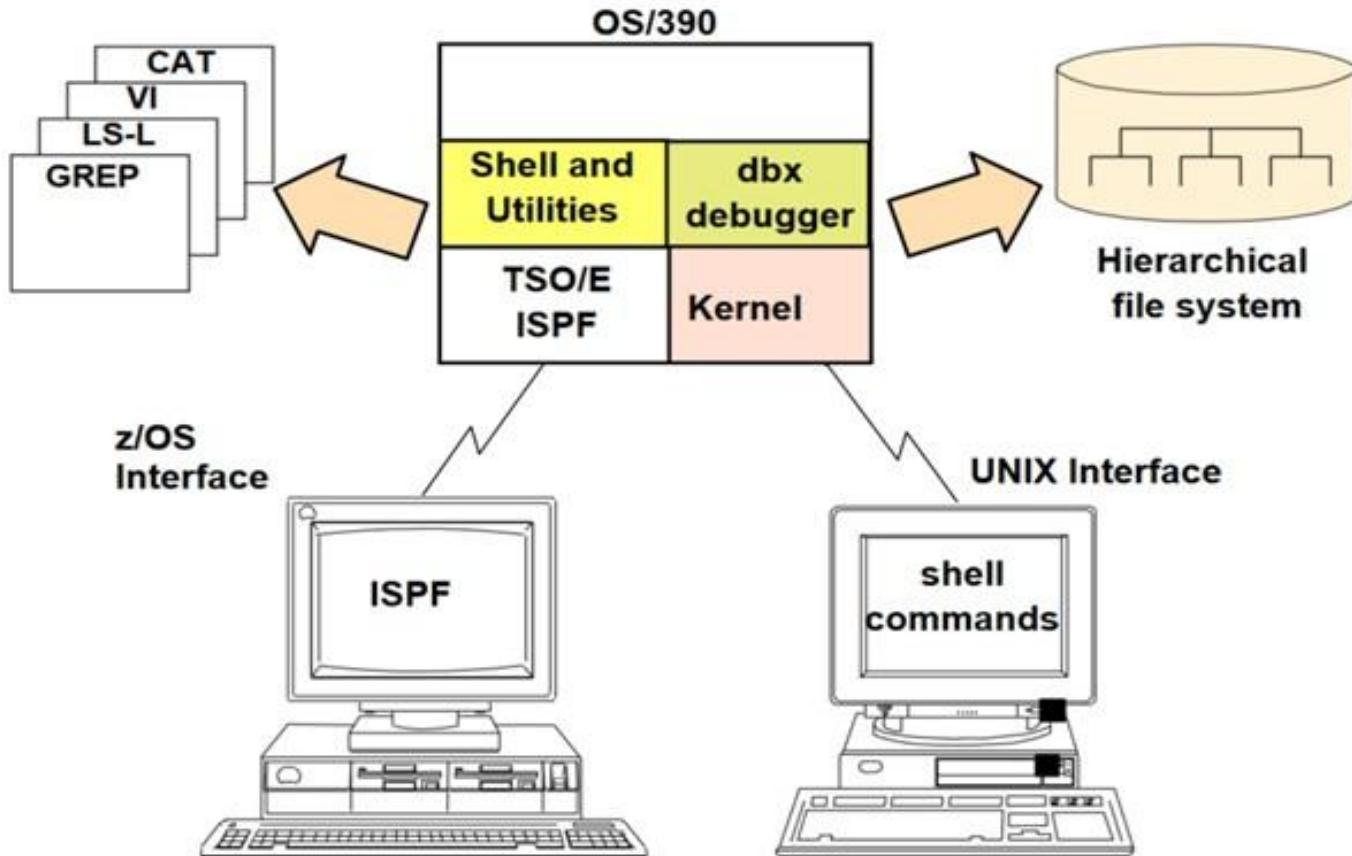
LOCKS

- LOCKWORDS en Almacenamiento Predefinido
- Sistema Único
- Serializar Recursos del Sistema

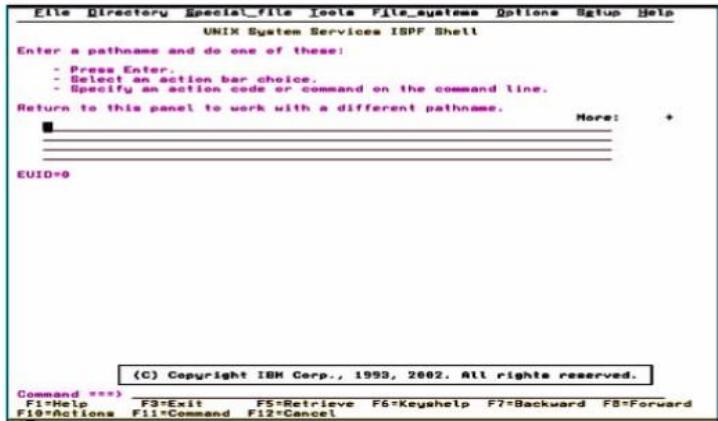
Services of the Supervisor - UNIX Syst. Serv.

- Program Linkage
- Dispatching
- Interrupt Handling
- Supervisor Calls
- Inter-address Space Communication
- Serialization
-  UNIX System Services
- Resource Balancing
- Measurement and Reporting

UNIX System Services (USS)



Two USS interfaces

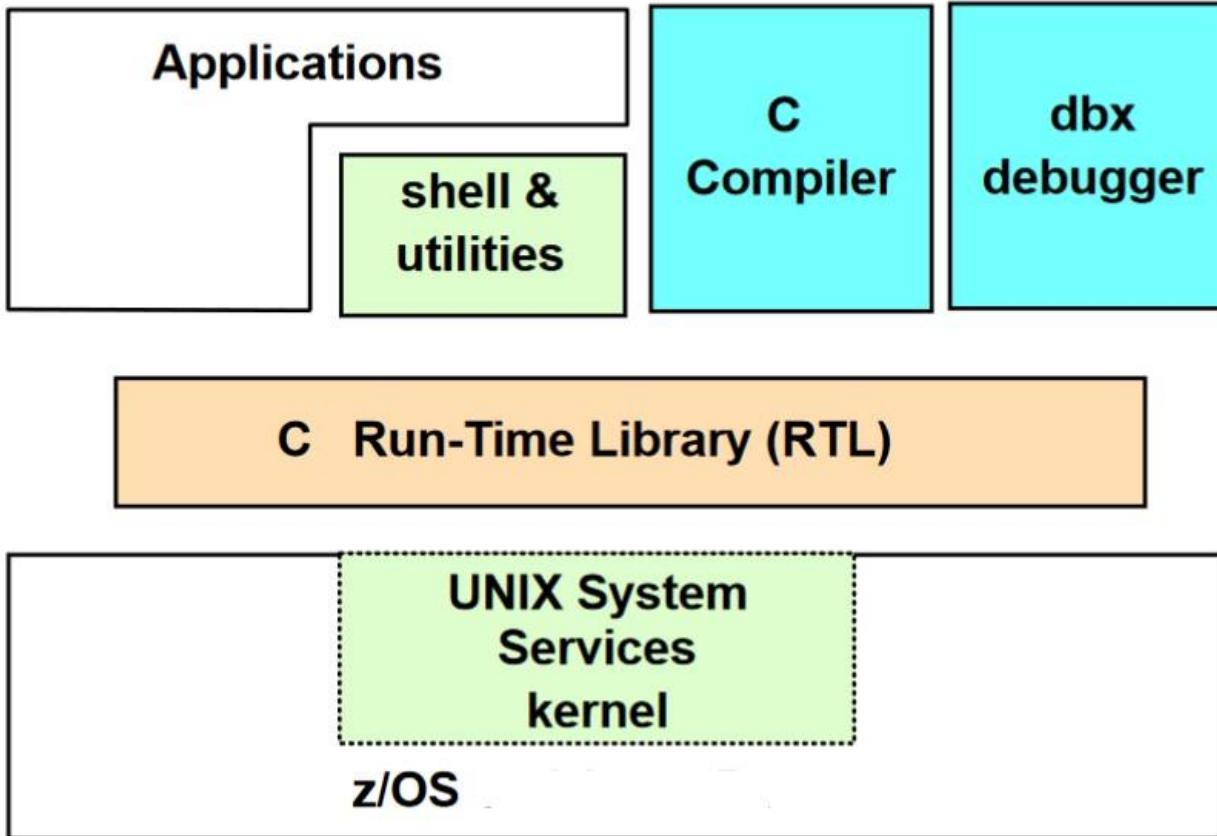


Use **ISHELL** command
to enter USS in
TSO/ISPF like mode

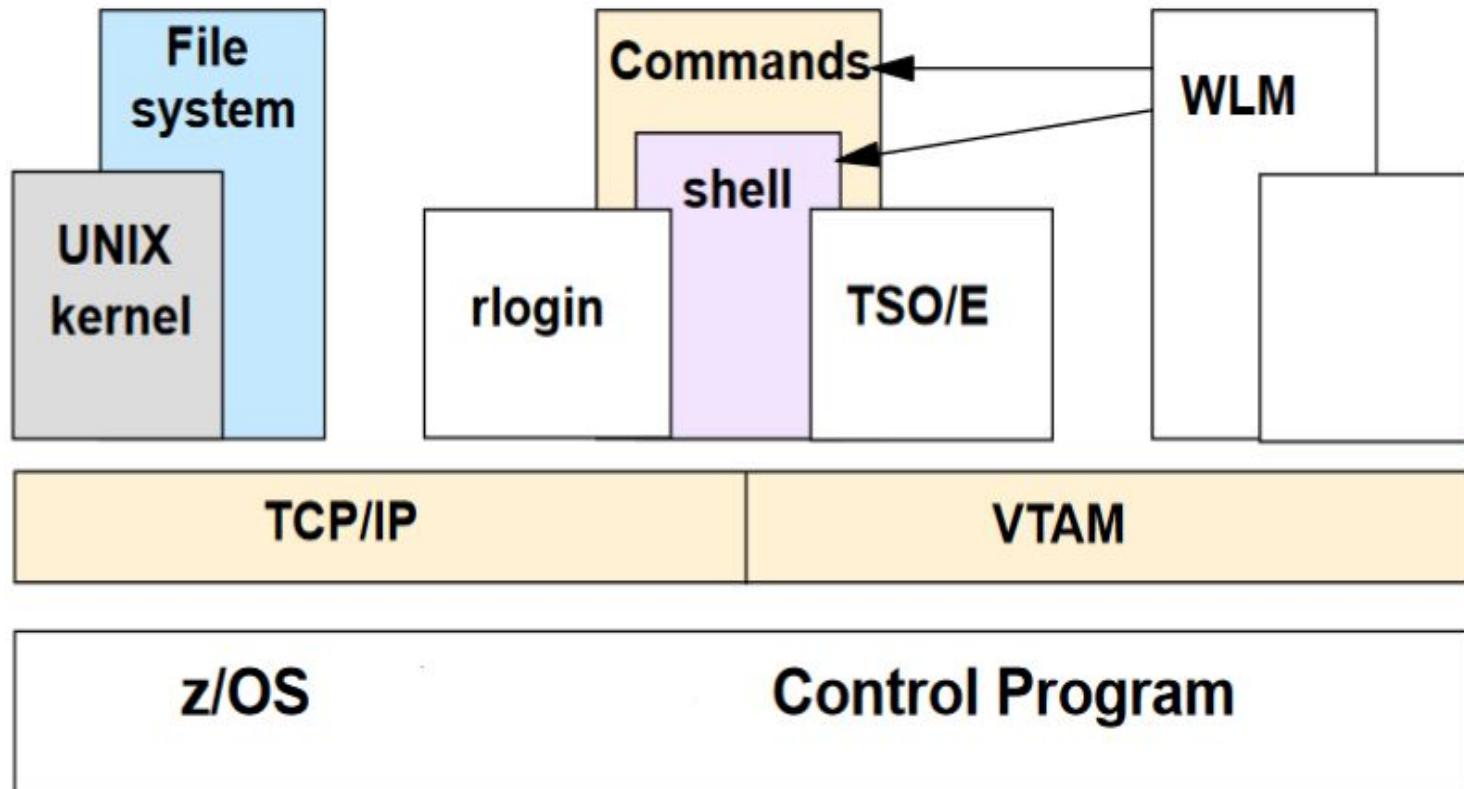
Use **OVMS** command
to enter USS in command
line mode



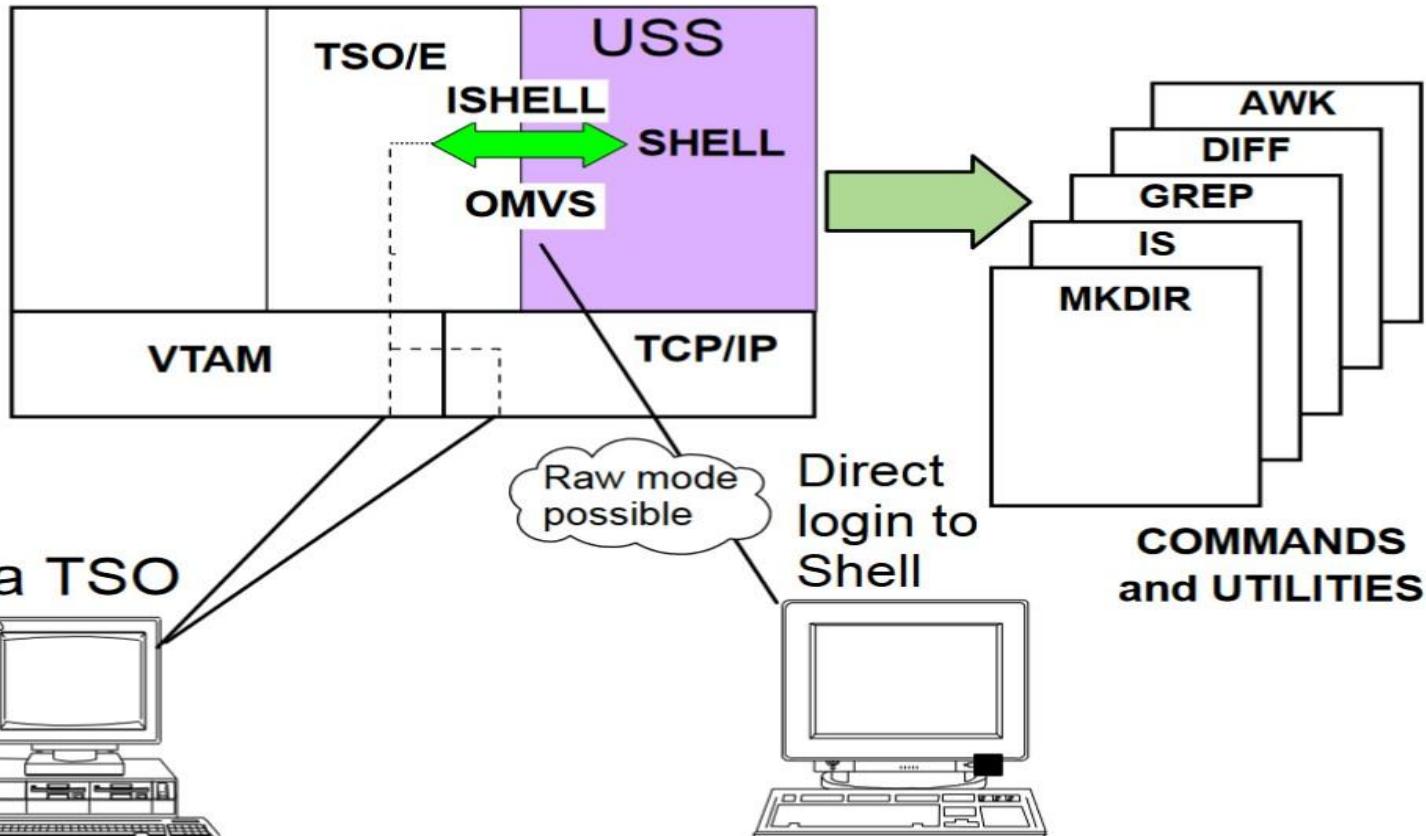
Componentes del Sistema de Servicios UNIX



Estructura del Sistema

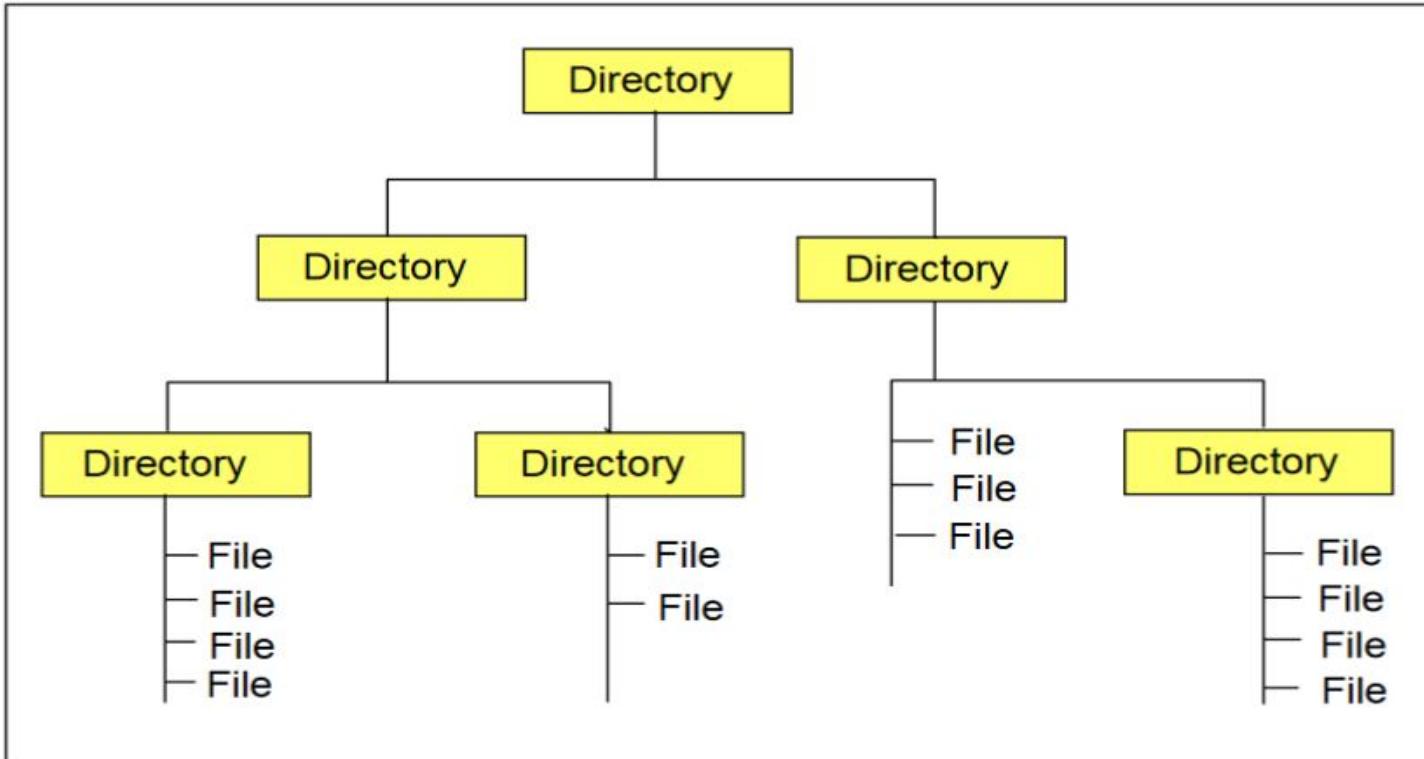


Interfaces con el USS

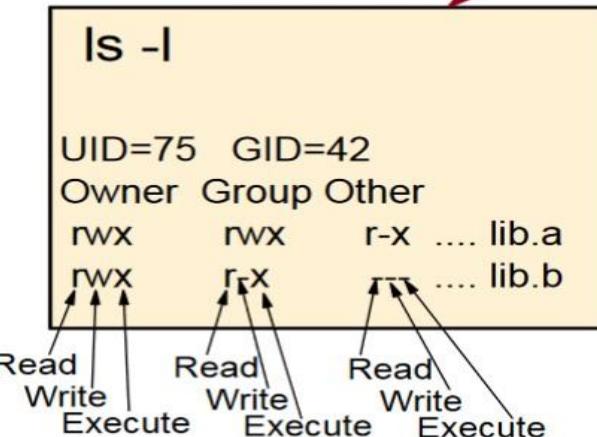
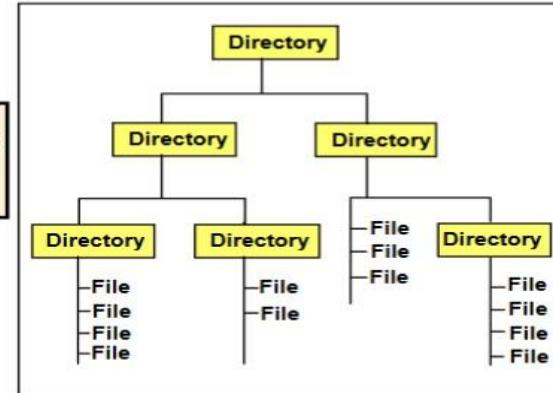
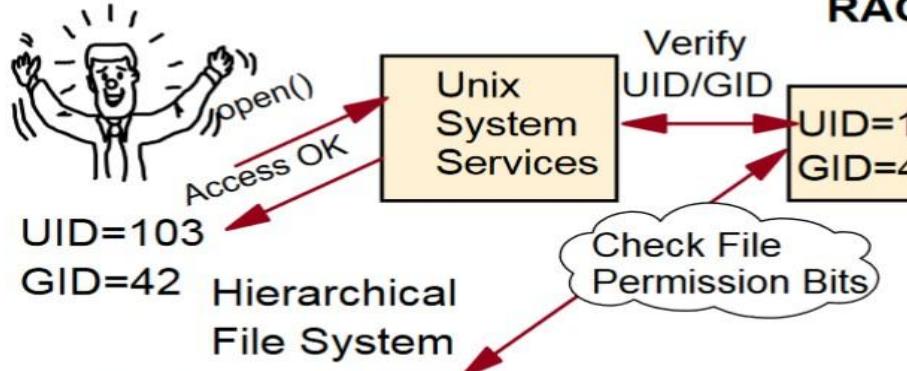


Hierarchical File System

HFS data set

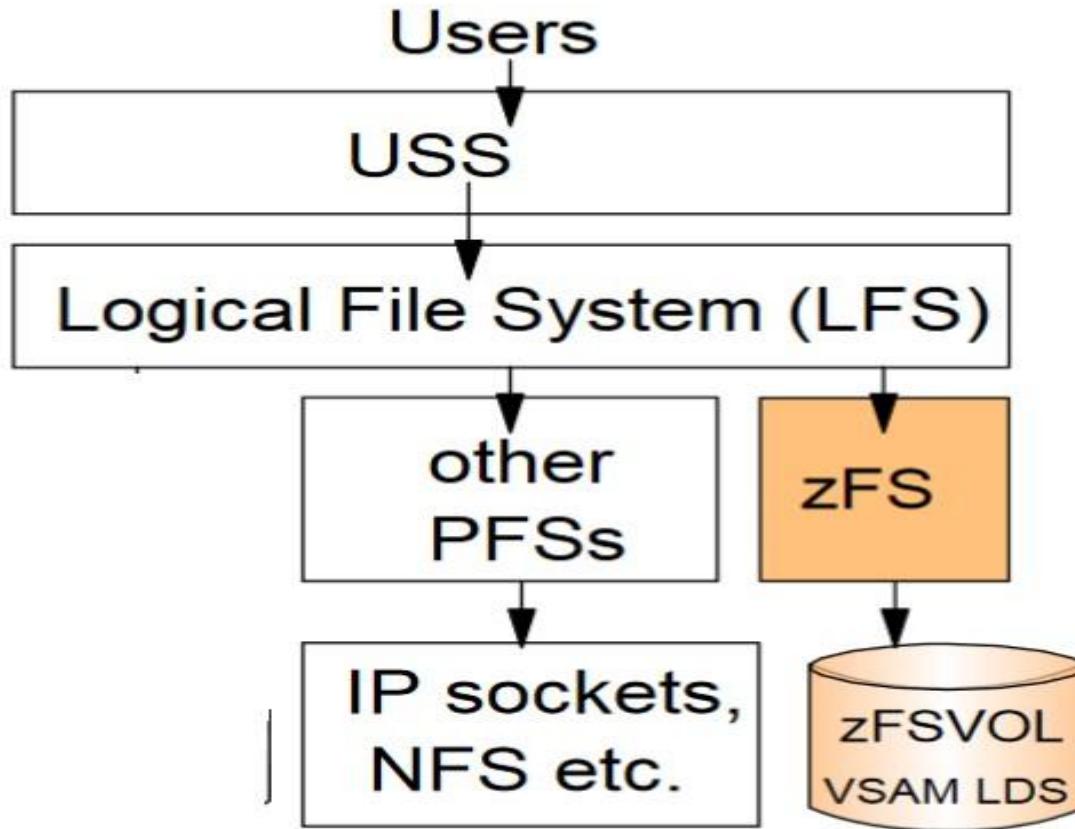


Ficheros y seguridad en USS

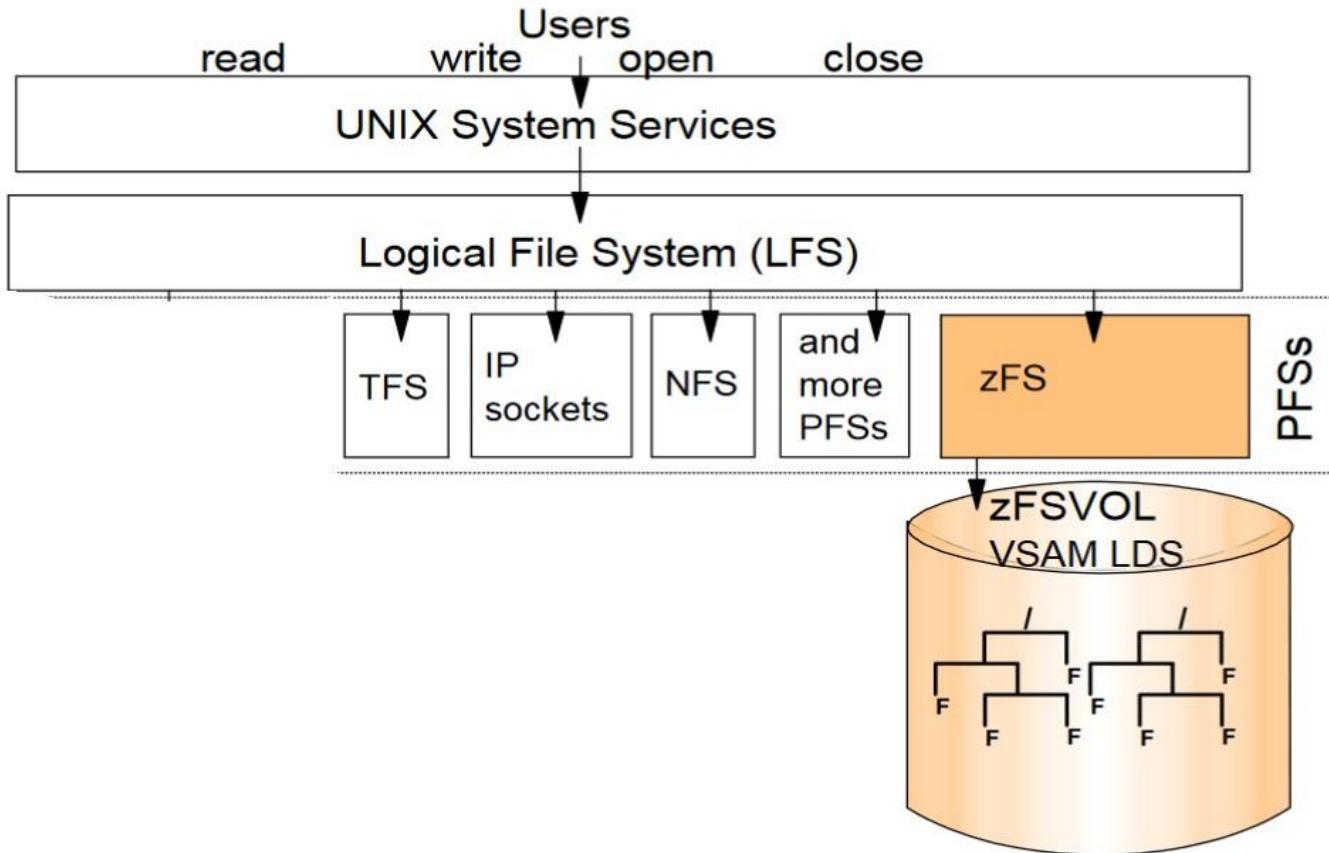


	Files	Directories
Read	read or print contents of file	read (not search) directory
Write	change/add/delete contents of file	change/add/delete entries in directory
Execute	execute file	search a directory

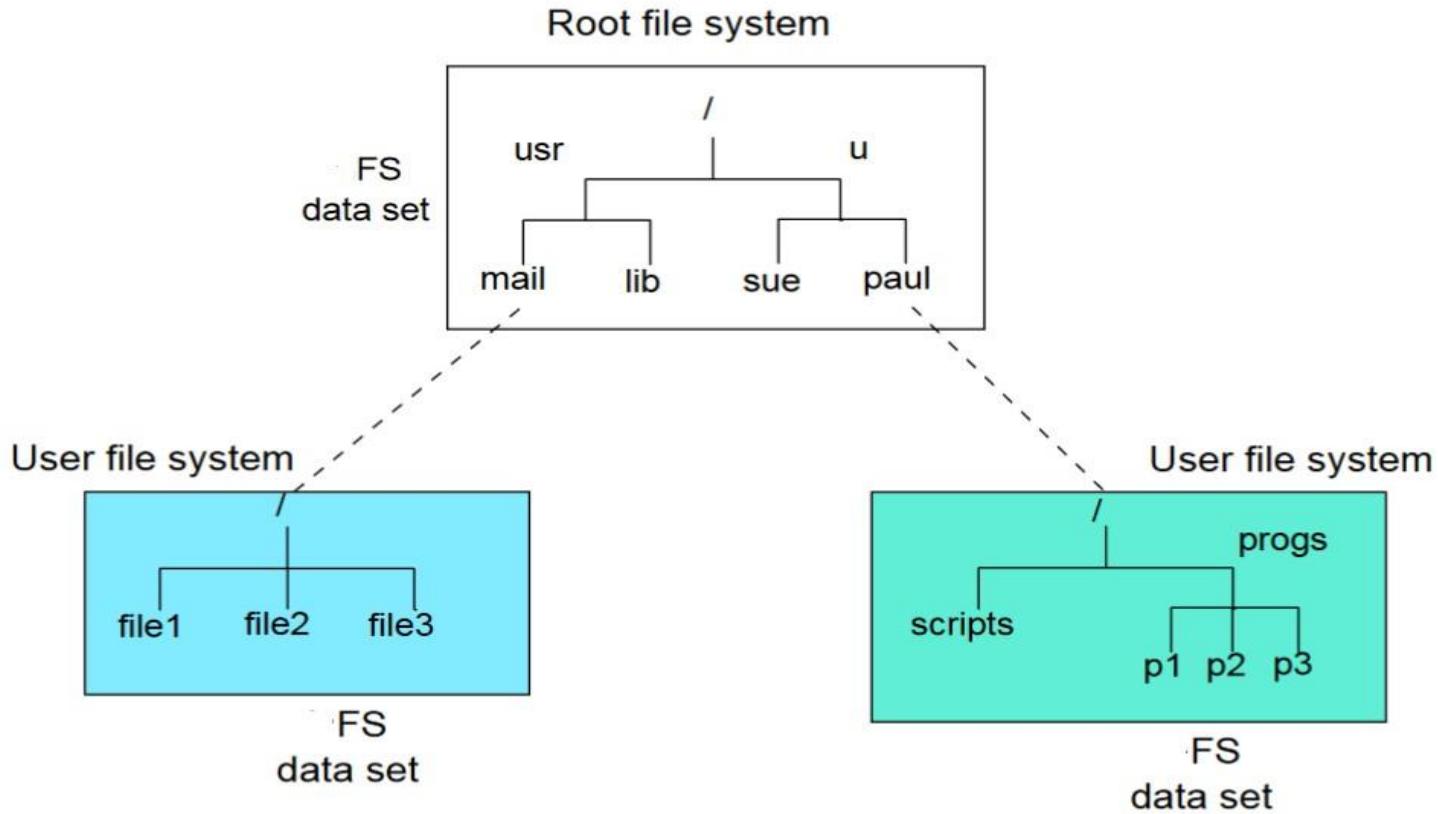
DFS zSeries File System (zFS)



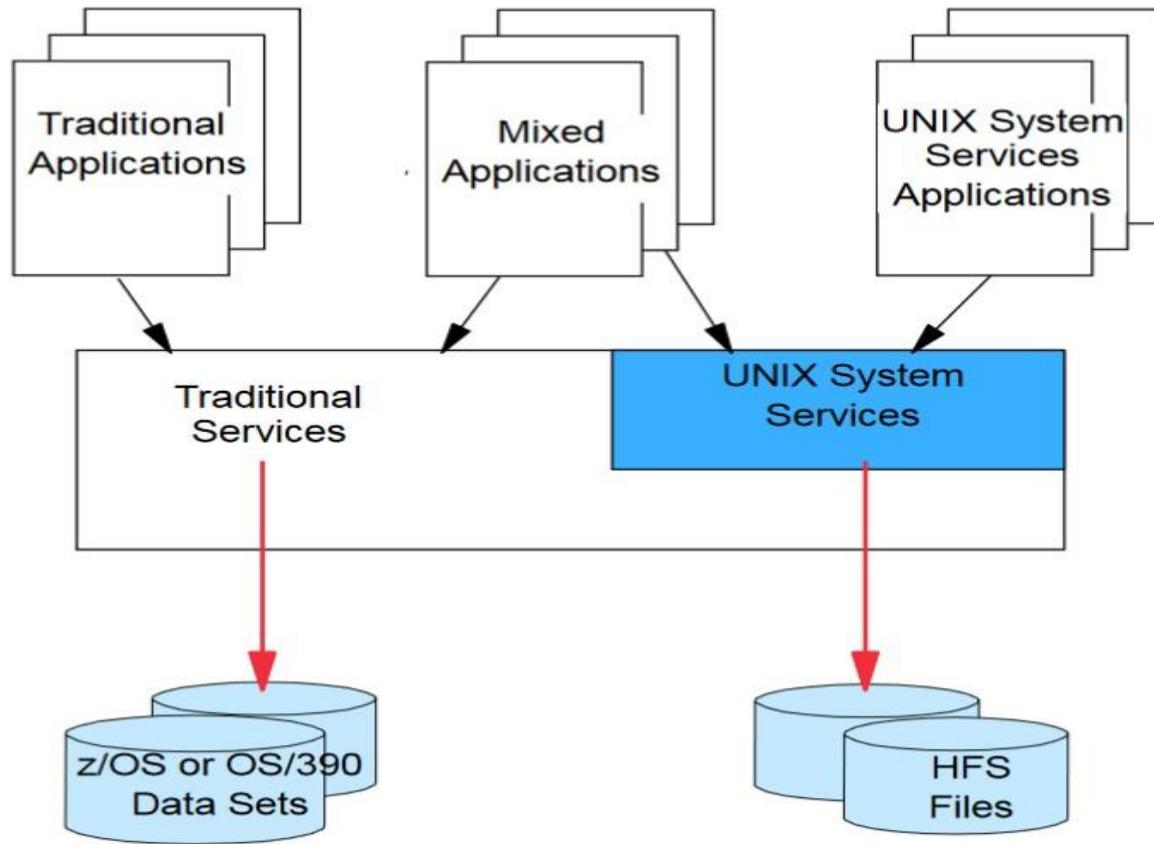
Physical File Systems (PFSs)



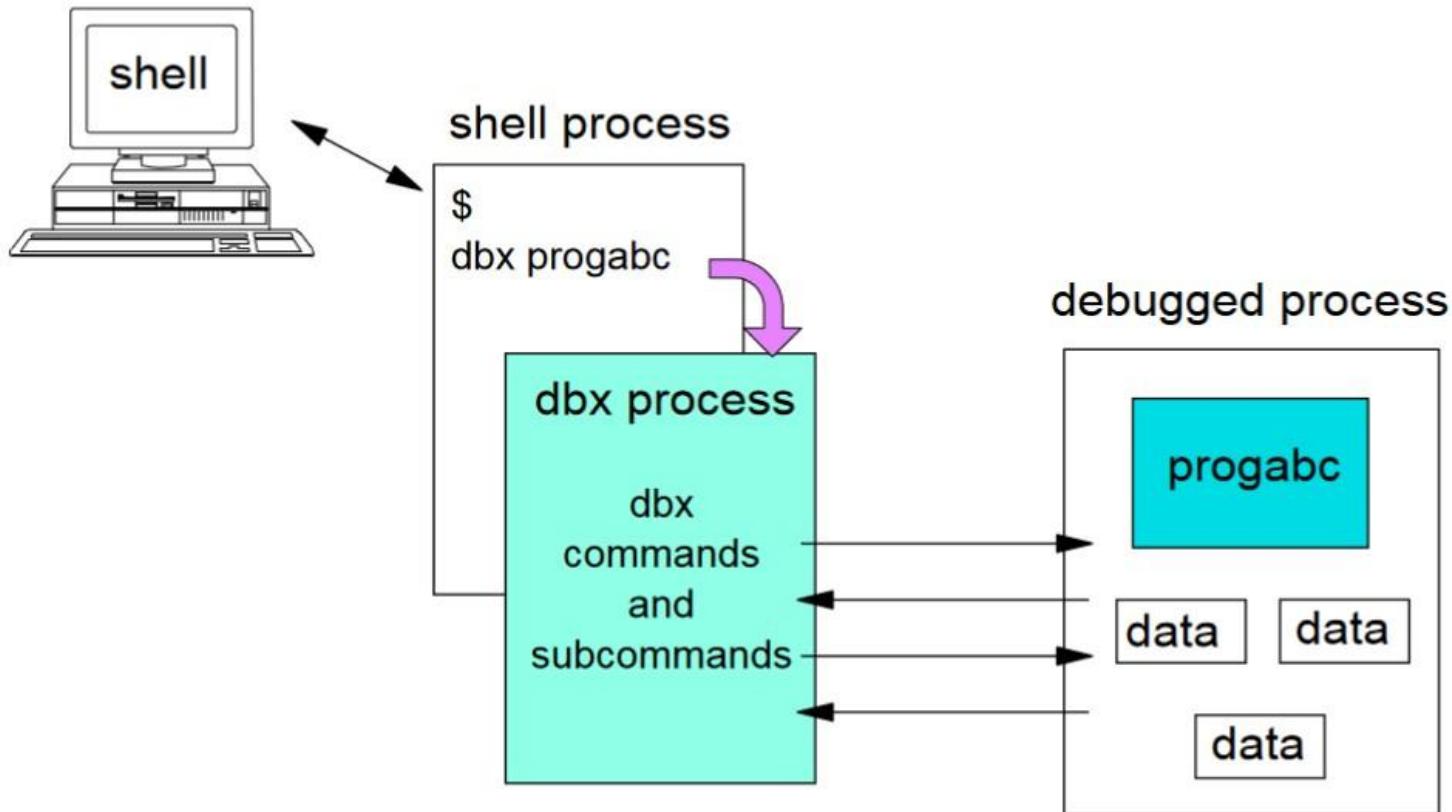
File System Structure



Application Development



El dbx Debugger



Services of the Supervisor - Resource Balanc.

Program Linkage

Dispatching

Interrupt Handling

Supervisor Calls

Inter-address Space Communication

Serialization

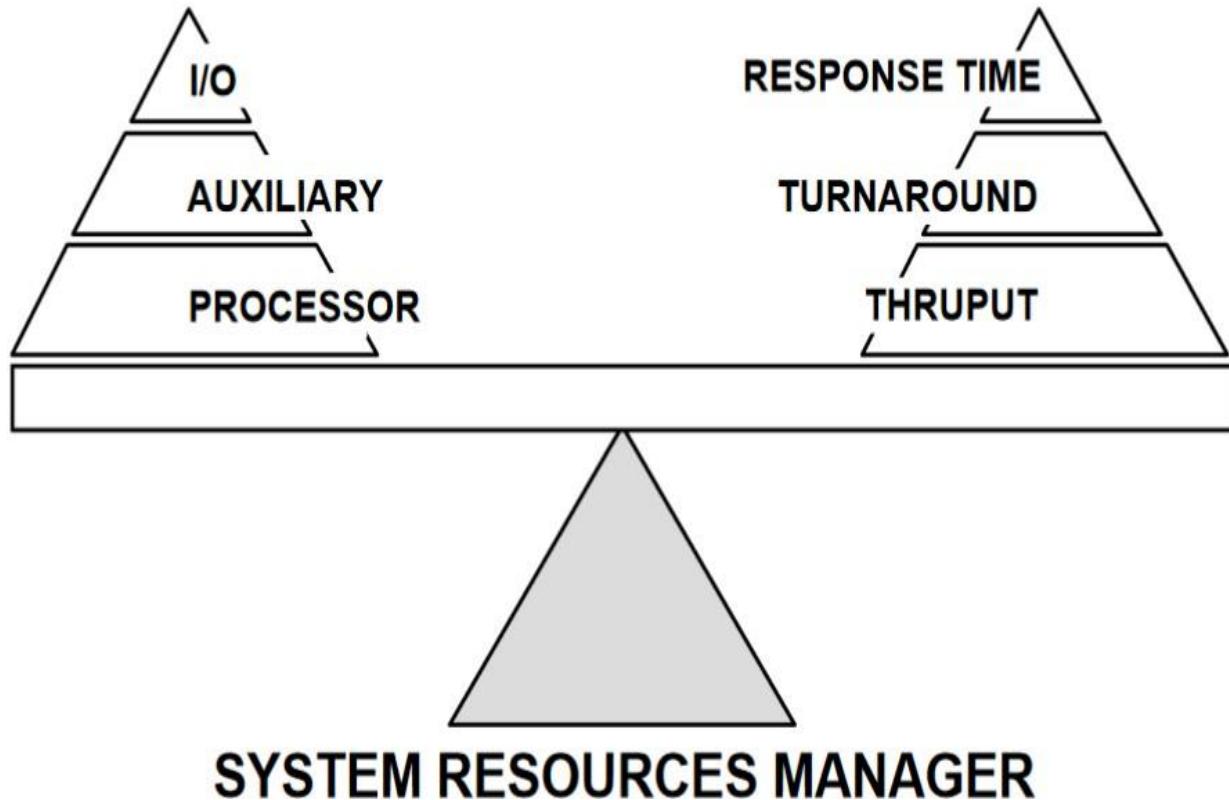
UNIX System Services



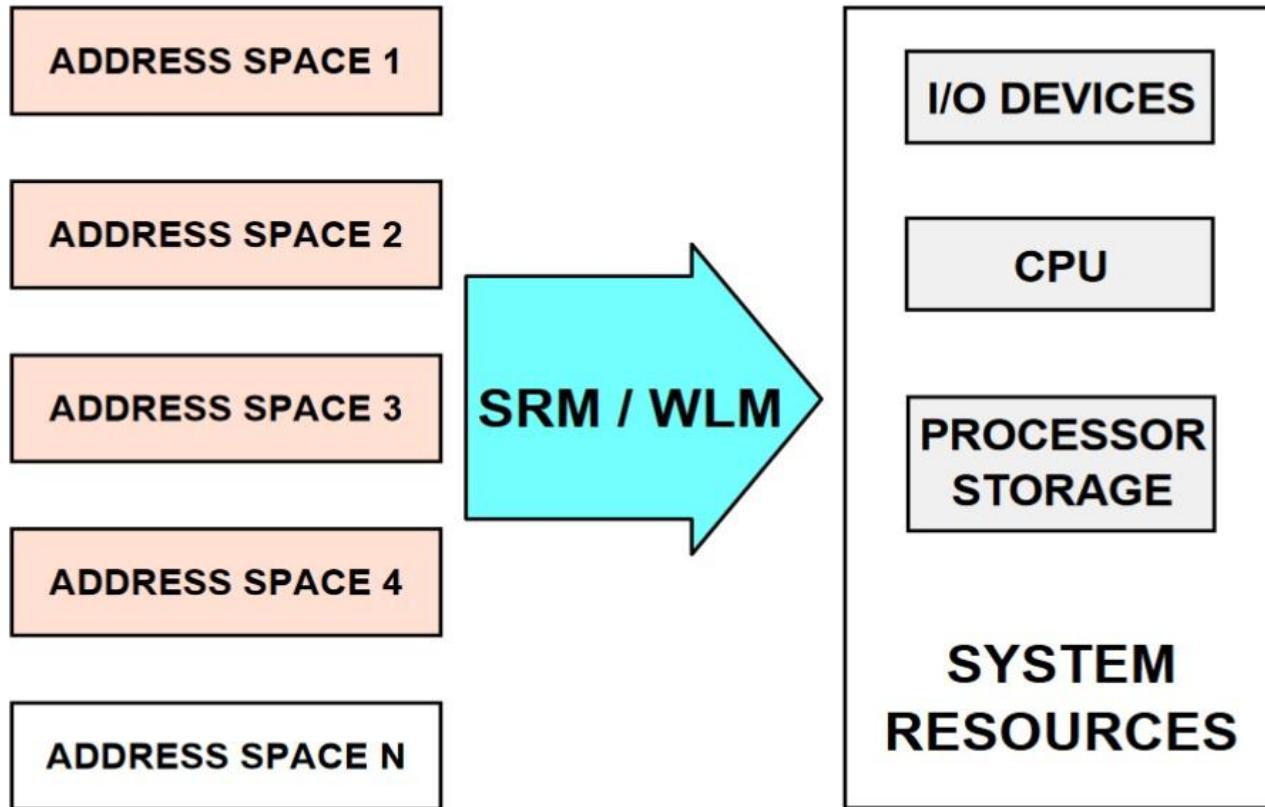
Resource Balancing

Measurement and Reporting

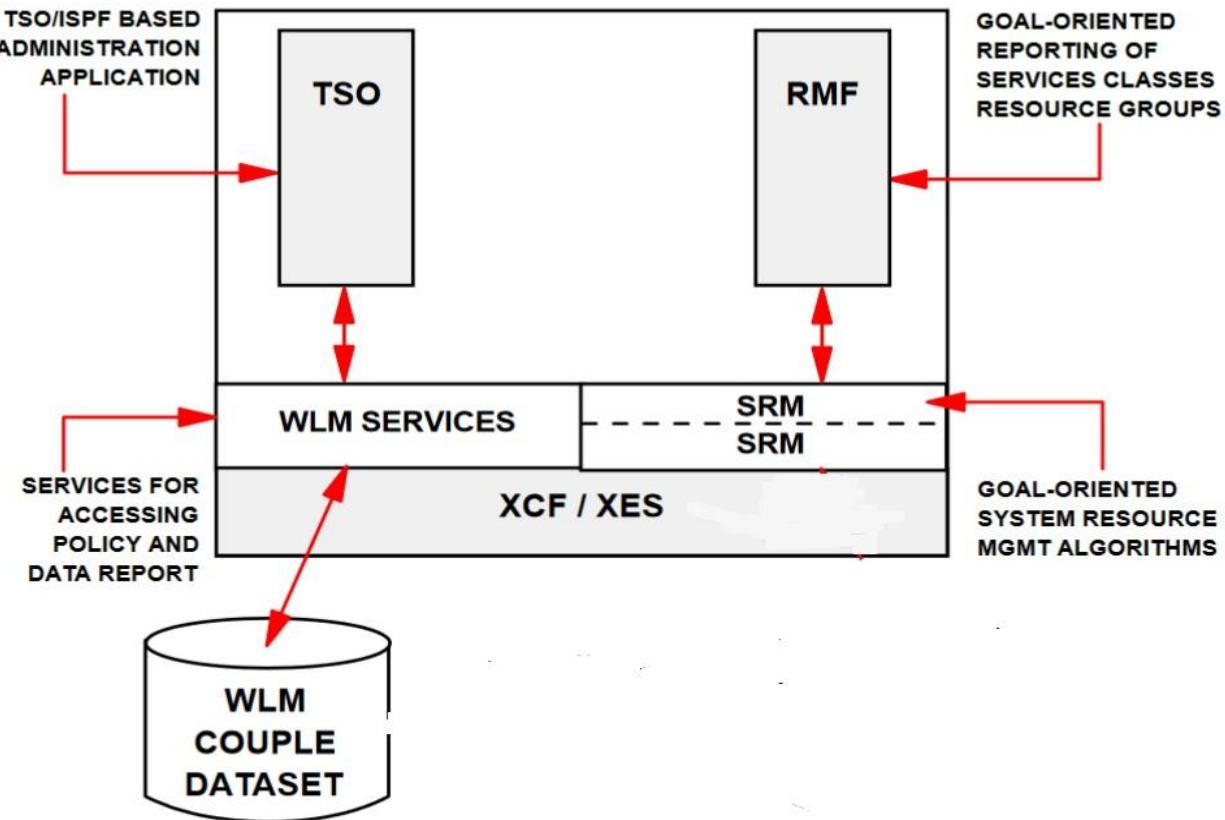
Balancing System Resources



SRM/WLM



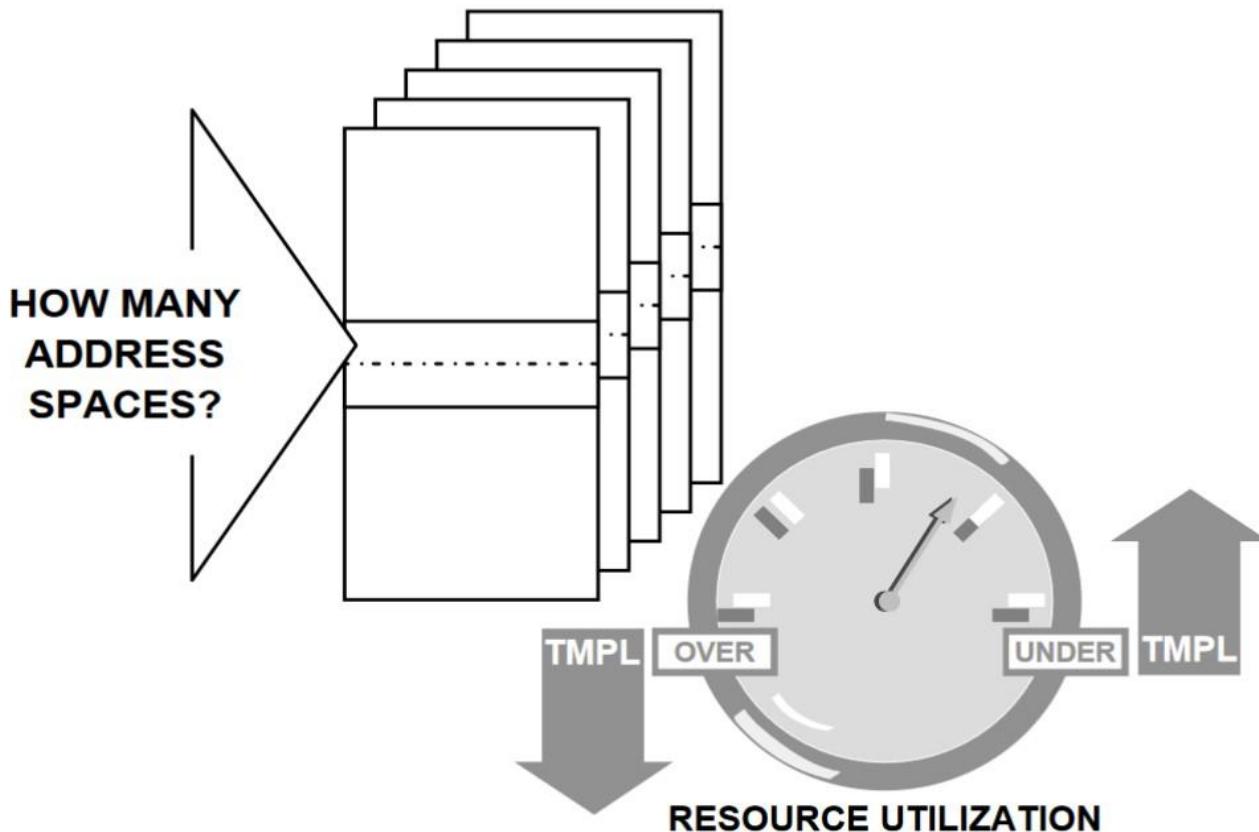
El Entorno WLM



Técnicas SRM

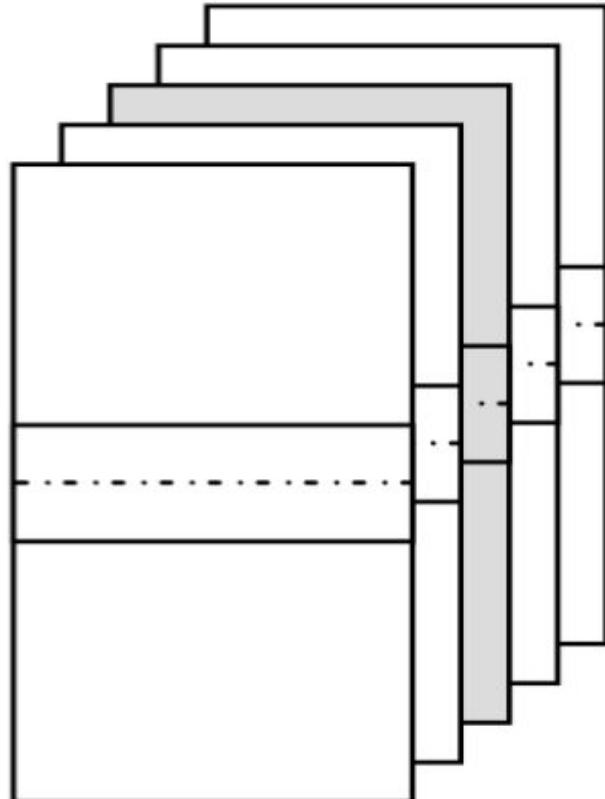
- Ajustar el nivel de la programación multinivel
- Swapping
- Steal page frames
- Restricción en la creación del address space
- Ajustar la prioridad de envío

Multiprogramming Level (MPL)



Swapping

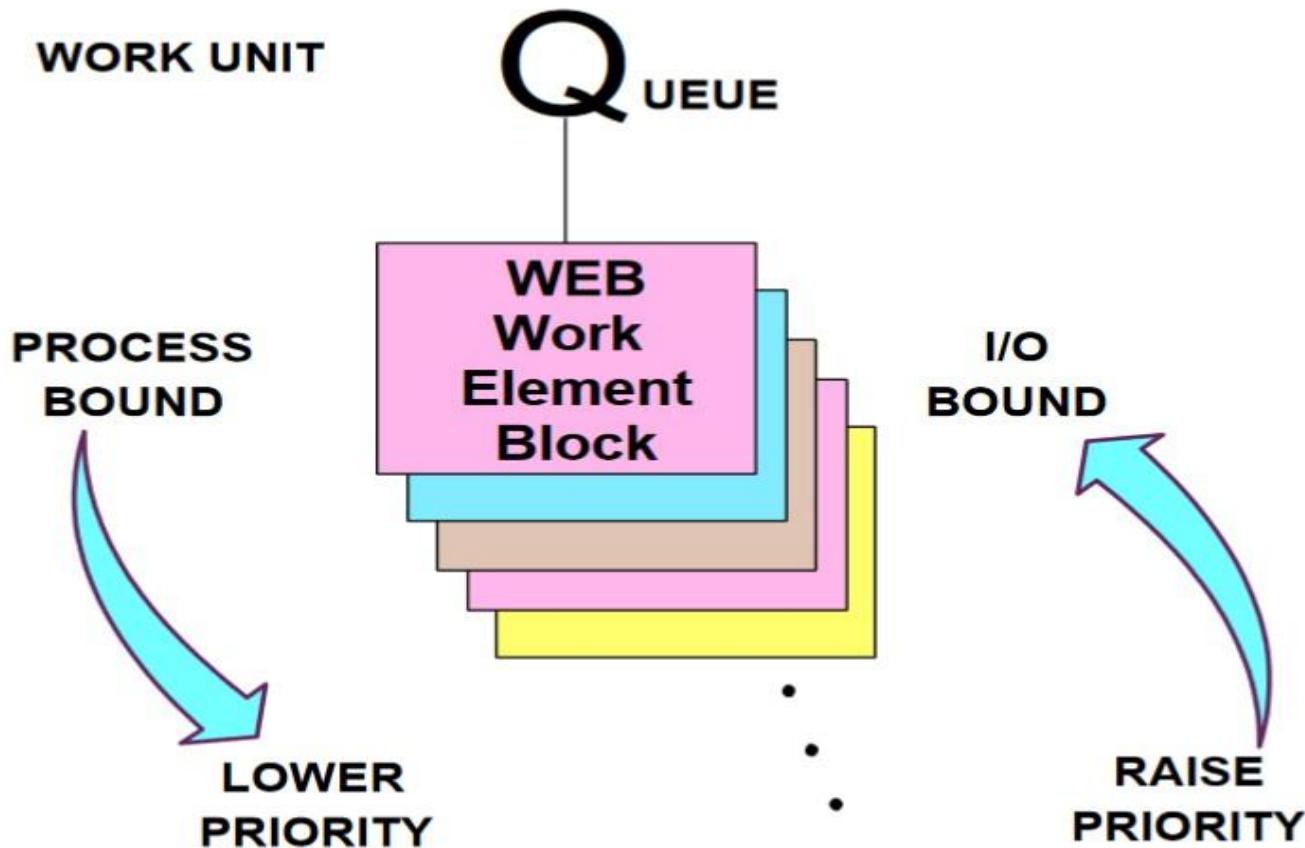
SERVICE
PROCESSOR
STORAGE
I/O



Restricción en la creación del address space



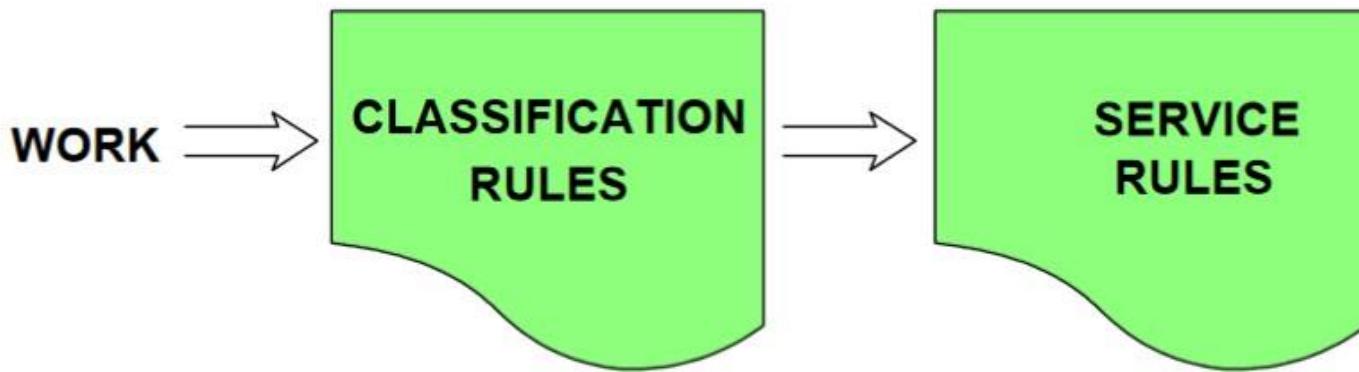
Ajuste de las prioridades de envío



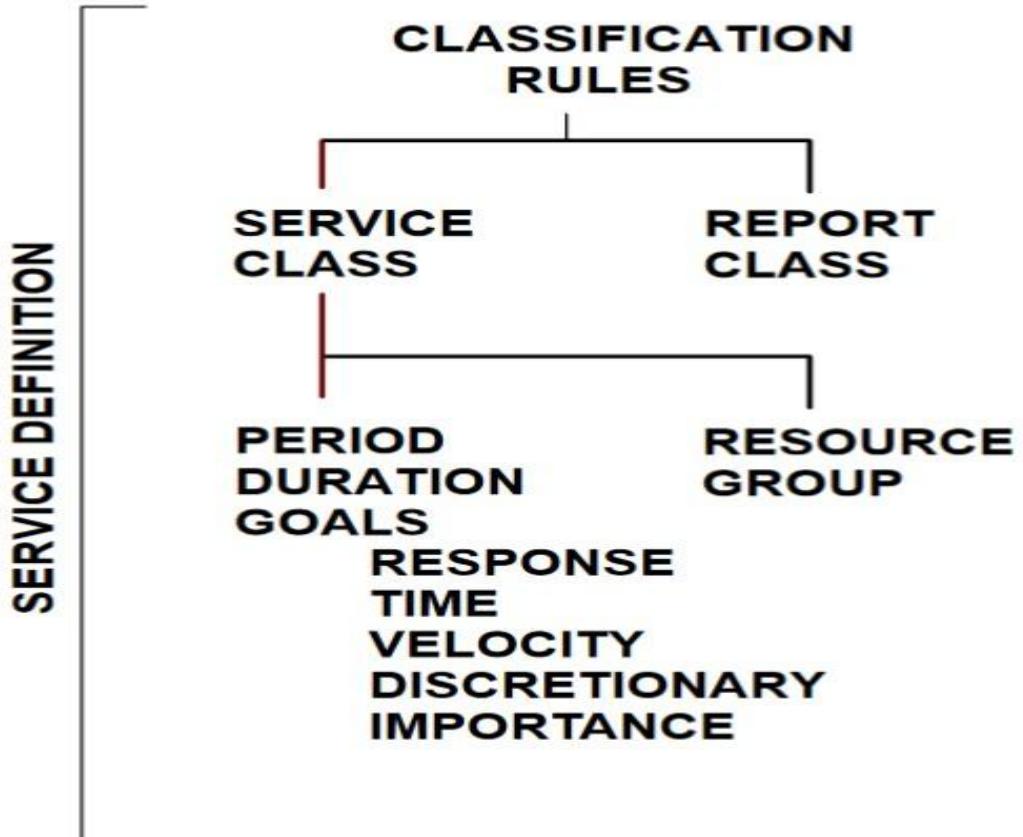
El rol de WLM



Clasificación: WLM - SRM



Controles de WLM



Services of the Supervisor - Msrmt & Rptng

Program Linkage

Dispatching

Interrupt Handling

Supervisor Calls

Inter-address Space Communication

Serialization

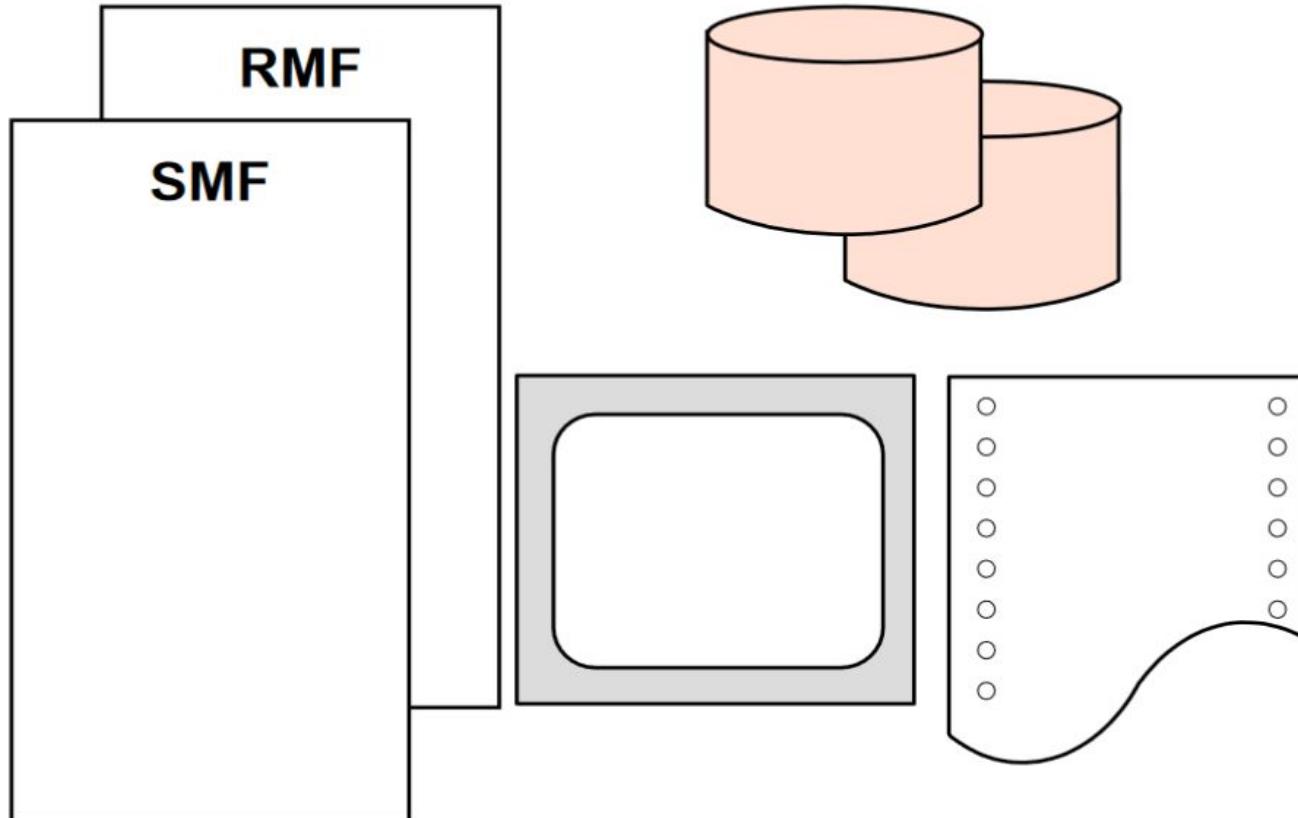
UNIX System Services

Resource Balancing

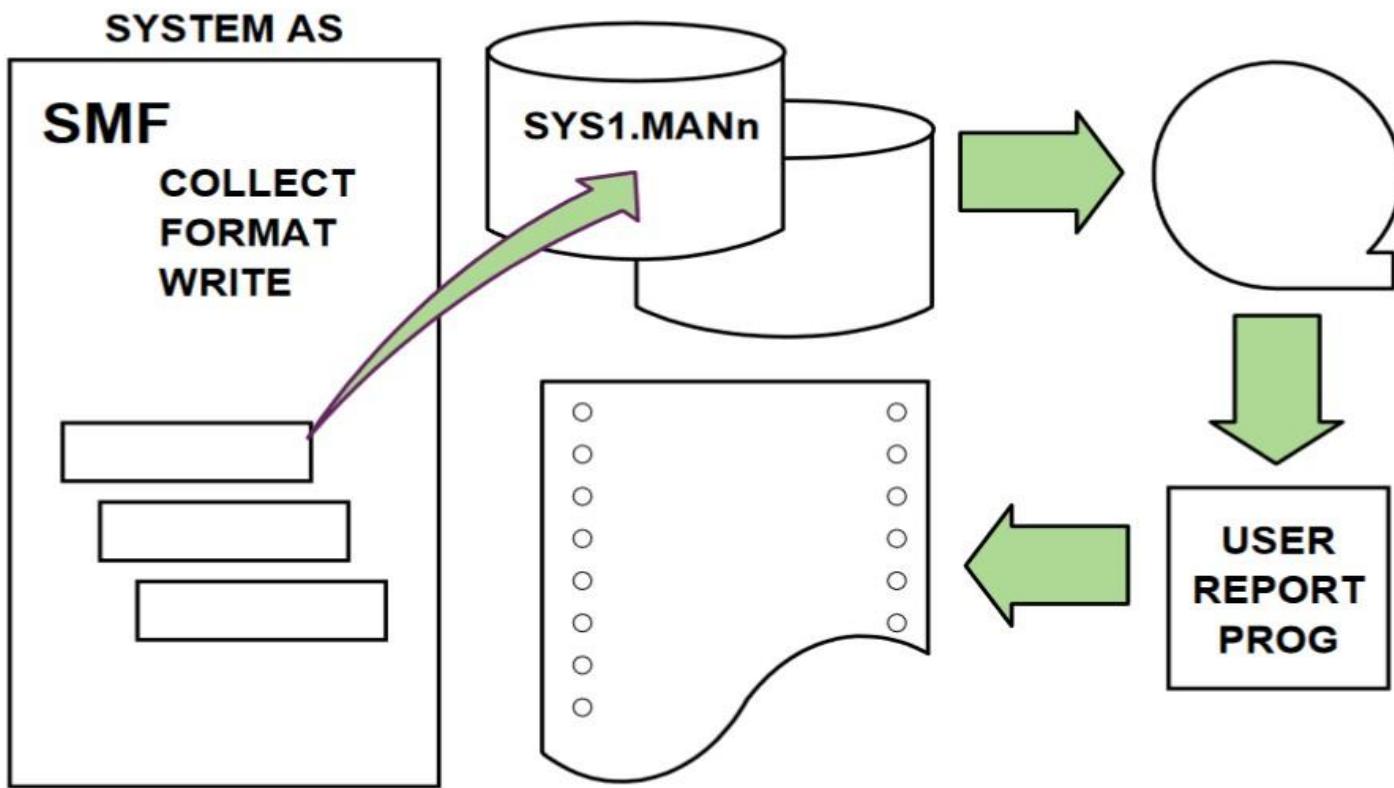
Measurement and Reporting



Medición y Elaboración de informes



SMF



Performance Reporting: RMF

RMF - Performance Management

Selection ===>

Enter selection number or command on selection line.

1 Postprocessor Postprocessor reports for Monitor I, II, and III (PP)

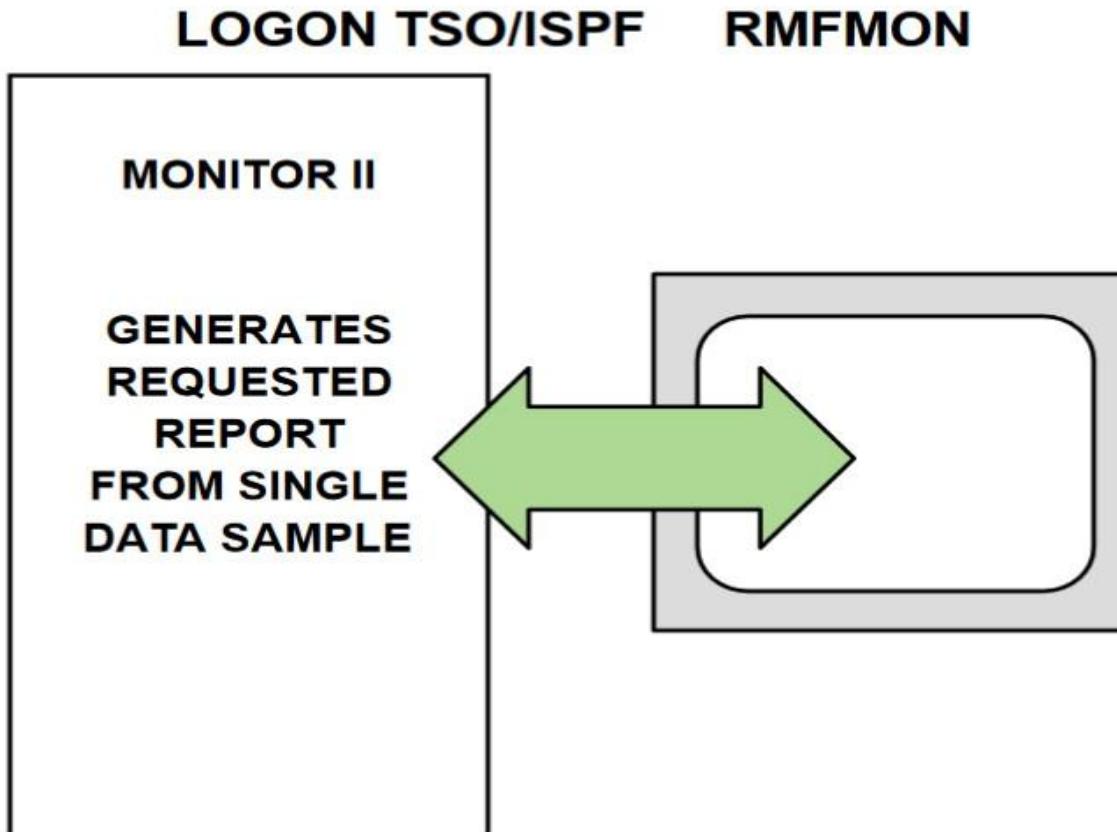
2 Monitor II Snapshot reporting with Monitor II (M2)

3 Monitor III Interactive performance analysis with Monitor III (M3)

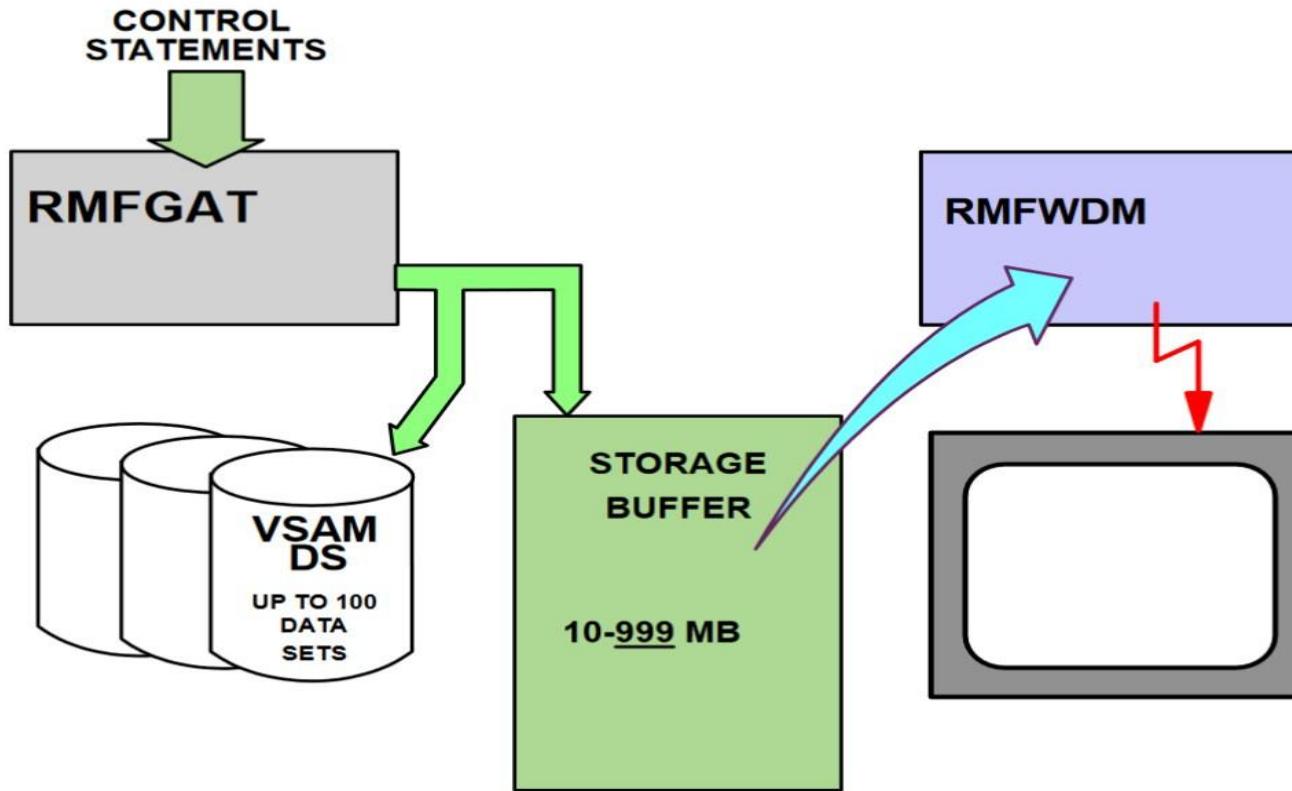
U USER User-written applications (add your own ...) (US)

T TUTORIAL X EXIT

RMF Monitor II



RMF Monitor III





MUCHAS GRACIAS

